Blind Deconvolution of Multi-Input Single-Output Systems with Binary Sources

Konstantinos I. Diamantaras, Member, IEEE and Theophilos Papadimitriou, Member, IEEE *†‡

IEEE Transactions on Signal Processing, vol. 54, no. 10, pp. 3720-3731, October 2006

Abstract

The problem of Blind Source Separation for Multi-Input Single-Output (MISO) Systems with binary inputs is treated in this paper. Our approach exploits the constellation properties of the successor values for each output sample. In the absence of noise, the successors of each output value form a characteristic finite set of clusters (successor constellation). The shape of this constellation is invariant of the predecessor value and it only depends on the last filter tap. Consequently, the localization of the successors constellation can lead to the removal of the last filter tap, thus reducing the length of the filter – a process we call *channel deflation*. Based on the Successor Observation Clustering (SOC) we develop two algorithms for Blind Source Separation: SOC-1 and SOC-2 differing mainly on the required size of the data set.

Furthermore, the treatment of the system in the presence of noise is described using data clustering and data correction. The problem of noise is attacked using a statistical mode based method. Moreover, we correct the problem of missclassified observations using an iterative scheme based on the Viterbi algorithm for the decoding of a Hidden Markov Model.

1 Introduction

Consider the general case of a set of signals transmitted in a wireless system and received by a set of captors. This situation may occur under communication context (analog or digital signals emitted from transmitters and received from an antenna array) or under signal processing context (voices and sounds received by a set of microphones). The received signals are mixtures of the source signals. The study of the mixtures is not trivial, due to the complexity of the system. In systems where we can assume that the delay of the sources is negligible we can adopt the instantaneous mixing model. This occurs, for example, in a recording studio, where the receivers (microphones) are near the sources (musical instruments), the walls do not rebound the sound waves and the outside noises are minimized. However, this is the case of a special "sterilized" environment. In real life, the sources arrive in the receivers with various time delays, they suffer intersymbol interference and the mixtures are always distorted from the effect of noise.

*K.I. Diamantaras is with the Department of Informatics, Technological Education Institute of Thessaloniki, GR-54101, Sindos, Thessaloniki, Greece

†Th. Papadimitriou is with the Department of Int. Economic Relations and Development, Democritus University of Thrace, GR-69100, Komotini, Greece

‡This work has been supported by the "EPEAEK Archimedes-II" Programme funded in part by the European Union (75%) and in part by the Greek Ministry of National Education and Religious Affairs (25%).

As a consequence, the communication and signal processing scenarios, we described, are better modelled using the convolutive *Multi-Input Multi-Output* (MIMO) model.

The recovery of the source signals using only their mixtures, making as little as possible *a priori* assumptions on the mixing parameters is referred as *Blind Source Separation*

(BSS). The problem in its general form is unsolvable. As a consequence, the proposed BSS techniques are build on special cases and restricted to datatypes with specific properties. There is a vast collection of methods that assume the non-gaussianity of the source signals. Their separation is based on the statistical independence of the extracted sources and *Higher-Order Statistics* (HOS) [1, 2, 3, 4, 5, 6]. Another group of techniques is based on the assumption that the source signals are white (or that they are whitened). These mixture signals can be separated using *Second-Order Statistics* (SOS) [7, 8, 9, 10, 11, 12]. Alternatively, a group of methods is based on the assumption that the sources have Finite Alphabet or Constant Modulus [13, 14, 15, 16].

In general, sufficient number of receivers (more receivers than transmitters) guarantee the deconvolution of the signals using various techniques [16, 17]. In many real-world applications, however, the number of the involved source signals is larger than the number of the received signals. Similarly, in other applications the number of the sources is *a priori* unknown. The sources estimation of the described problem is often encountered as under-determined or over-complete Blind Source Separation [18, 19, 20]. The most extreme case of under-determined Blind Source Separation is when we receive only one mixture of several source signals and it is better described by the *Multi-Input Single-Output* (MISO) model. The blind deconvolution of a MISO system with finite alphabet signals is discussed in [21]. The approach is algebraic and converts the nonlinear system into a linear one using the finite alphabet property. The application of a MIMO technique to a MISO system was proposed in [22]. The method behaves almost as good as in the MIMO case. In [23], a semi-blind technique is presented, in which training sequences of the system are available and they can be used to acquire channel estimates. Aldana *et al.* in [24] treats the input symbols as discrete random variables in a stochastic likelihood criterion. The system is solved by applying the Expectation-Maximization algorithm in the frequency domain.

The proposed method is restricted to binary source signals. In [25] we addressed the problem of *Single-Input Single-Output* (SISO) system when the sources are integer binaries or complex numbers with binary components. The method recursively eliminates the channel parameters by studying at each step the succession patterns of the observation signal. The only assumption of the method is that every possible succession couple exists in the output dataset. The resulting system is memoryless and the source estimation is straightforward. Recently, we extended this technique to MISO systems with binary sources [26].

In this paper we will re-visit the blind deconvolution MISO problem with binary sources. Our model is described by the following equation:

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) + e(k), \quad k = 1, ..., K \tag{1}$$

where $\mathbf{h}_i$ for $i = 0, ..., L-1$ are a set of unknown real n-dimensional mixing vectors, $\mathbf{s}(k) = [s_1(k) \cdots, s_n(k)]^T$ is the n-dimensional source signal, and $e(k)$ corresponds to additive gaussian noise. The n independent binary sources take discrete values from the set $\{-1, +1\}$. The observations of the mixtures are real-valued scalars. The proposed approach is not related with the statistical properties of the signals (higher order or second order). Instead, we exploit the

geometrical properties of the observations succession. For any observation prototype a limited set of potential successor observations exists. We will show that if a complete set of successors for an observation is located and collected, then it can be used to create a simpler system i.e a MISO system with one filter tap less. Two algorithms based on the SOC will be developed, we call them SOC-1 and SOC-2 and treat large and small datasets respectively. In order to explore the presented methods in realistic framework, we insert an additive gaussian noise source in the MISO model. The elimination of noise is attained using a data clustering technique based on the statistical mode of the observation classes. This technique seems to drastically decrease the effect of noise, even though it introduces a new error source in the system, through mis-clustered observations. This happens whenever the noise component of an observation is larger than the corresponding class variance. We approach the problem using the Hidden Markov Model and we treat it successfully, using a data correction step based on the Viterbi algorithm.

This paper is organized as follows. In Section 2 the basic concepts of the presented methods in a simple noiseless system is presented. The section 3 is devoted in the study of the assumptions validity. The treatment of noise in the model will be described in section 4. First we describe how small amounts of noise can be eliminated from the dataset using a simple clustering technique. Then, we will introduce a method for the treatment of datasets heavily infected from noise. Results on several scenarios are presented in section 5. The paper concludes in section 6.

2 The Noiseless case

At first the noise component will be ignored, in order to introduce the concept of the successors constellation and describe the methods, SOC-1 and SOC-2, which are built on this idea. The treatment of the realistic model where the noise is present will be discussed in section 4. The basic ideas, that motivated us, are described in subsection 2.1, and they build the first method. An improvement of this method, which can deal with smaller datasets, will be introduced in subsection 2.2.

2.1 Successor Observation Clustering

Let us consider a noiseless MISO model described by the following equation:

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) \tag{2}$$

where $\mathbf{h}_i$ for $i = 0, ..., L-1$, are a set of unknown real n-dimensional mixing vectors called *filter taps*. The source vector signal $\mathbf{s}(k) = [s_1(k) \cdots, s_n(k)]^T$ is composed of n independent binary antipodal signals: $s_i(k) \in \{-1, +1\}$. The observations of the mixtures are real-valued scalars. For each k, the vector $\mathbf{s}(k)$ can take one of 2^n values denoted by $\mathbf{b}_i^{(n)}$, $i = 1, \cdots, 2^n$. The vector $\mathbf{b}_i^{(n)^T}$ is the i-th row of the matrix $\mathbf{B}^{(n)}$ defined in (3).

$$\mathbf{B}^{(n)} \stackrel{\triangle}{=} \begin{bmatrix} -1 & -1 & \cdots & -1 & -1 \\ -1 & -1 & \cdots & -1 & 1 \\ -1 & -1 & \cdots & 1 & -1 \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & 1 & \cdots & -1 & 1 \\ 1 & 1 & \cdots & 1 & -1 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \tag{3}$$

In [21] Yellin *et al.* introduced the notion of observation equivalence for SISO systems:

$$x(k) = \sum_{i=0}^{L-1} h_i s(k-i) \tag{4}$$

where $x(.), h_i,$ and $s(.)$ are scalars.

Definition 1 *Two observations $x(k)$ and $x(l)$ are said to be equivalent if the input values that produce them according to Eq.* (4) *are identical: $s(k-i) = s(l-i)$, for all $i = 0, \cdots, L-1$.*

Note that two equivalent observations are necessarily equal, but the converse may not be true. It is possible that two equal observations $x(k) = x(l)$, are produced by two different strings of input symbols $[s(k), \cdots, s(k-L+1)] \neq [s(l), \cdots, s(l-L+1)]$.

Let us extend the concept of observation equivalence to MISO systems by simply replacing the scalar inputs with vector inputs. Each observation $x(k)$ is generated by the linear combination of L n-dimensional source vectors, therefore, the observation space $\mathcal{X} \ni x(k)$ is a discrete set consisting of, at most, M elements, $M = 2^{nL}$. The cardinality $|\mathcal{X}|$ will be less than M if and only if there exist two different L-tuples $\{\mathbf{b}_{j_0}^{(n)}, \cdots, \mathbf{b}_{j_{L-1}}^{(n)}\}$ and $\{\mathbf{b}_{l_0}^{(n)}, \cdots, \mathbf{b}_{l_{L-1}}^{(n)}\}$, of binary vectors such that $\sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{b}_{j_i}^{(n)} = \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{b}_{l_i}^{(n)}$. The following avoids this situation:

Assumption 1 *Two observations $x(k)$, $x(l)$, are equivalent if and only if they are equal.* ■

Hence, $|\mathcal{X}| = M$ and so the observation sequence has length at least M. In other words, to every observation value $r \in \mathcal{X}$ corresponds a unique L-tuple $\{\bar{\mathbf{b}}_0(r), \cdots, \bar{\mathbf{b}}_{L-1}(r)\}$ of consecutive source vectors that generates this observation. No other observation value $r' \in \mathcal{X}$ corresponds to the same L-tuple of binary vectors. For any $x(k) = r$, we have

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{5}$$

since, by definition,

$$\bar{\mathbf{b}}_i(r) = \mathbf{s}(k-i), \qquad \text{for } i = 0, \cdots, L-1.$$

Now the **successor observation**, $x(k+1)$, can be written as:

$$\begin{aligned} x(k+1) &= \mathbf{h}_0^T \mathbf{s}(k+1) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-(i-1)) \\ &= \mathbf{h}_0^T \mathbf{s}(k+1) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \end{aligned} \tag{6}$$

Since $\mathbf{s}(k+1)$ is an n-dimensional binary antipodal vector, $x(k+1)$ can take one of the following 2^n possible values:

$$y_p(r) = \mathbf{h}_0^T \mathbf{b}_p^{(n)} + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r), \quad p = 1, \cdots, 2^n \tag{7}$$

Note that the successor values $y_p(r)$ do not depend on the specific time index k but only on the observation value r. Therefore, each observation value r creates a class of successors $\mathcal{Y}(r)$ with cardinality $|\mathcal{Y}(r)| = 2^n$. Furthermore, we have $\sum_{p=1}^{2^n} \mathbf{b}_p^{(n)} = 0$, so the mean $\bar{y}(r)$ of the members of $\mathcal{Y}(r)$ is:

$$\begin{aligned} \bar{y}(r) &= \frac{1}{2^n} \sum_{p=1}^{2^n} y_p(r) \\ &= \frac{1}{2^n} \left(\mathbf{h}_0^T \sum_{p=1}^{2^n} \mathbf{b}_p^{(n)} + 2^n \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \right) \\ &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r). \end{aligned} \tag{8}$$

Now, let us replace every $x(k) = r$ with the mean $\bar{y}(r)$ to obtain a new sequence

$$\begin{aligned} x^{(2)}(k) &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \\ x^{(2)}(k) &= \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i+1) \end{aligned} \tag{9}$$

The new MISO system (9) has the same taps as the original system (2) except that it is shorter since $\mathbf{h}_0$ is missing. We will say that the system has been deflated. An additional but trivial difference is that the source sequence is time-shifted. Based on the discussion above, the whole **filter-** or **system-deflation** method, is summarized as follows:

Algorithm 1 (SOC-1.)

Step 1. *For every $r \in \mathcal{X}$ locate the set of time instances $\mathcal{K}(r) = \{k : x(k) = r\}$.*

Step 2. *Find the successor set $\mathcal{Y}(r) = \{x(k+1) : k \in \mathcal{K}(r)\}$. This set must contain 2^n distinct values $y_1(r)$, ..., $y_{2^n}(r)$.*

Step 3. *Compute the mean $\bar{y}(r) = 1/2^n \sum_{i=1}^{2^n} y_i(r)$.*

Step 4. *Replace $x(k)$ by $\bar{y}(r)$, for all $k \in \mathcal{K}(r)$.* ■

Clearly, for this method it is essential that all observation/successor pairs $[r, y_i(r)]$, $i = 1, \cdots, 2^n$, will appear, at least once, in the output sequence x. Applying the deflation method $L-1$ times, the system will be eventually reduced to a multi-input single-output instantaneous problem:

$$x^{(L)}(k) = \mathbf{h}_{L-1}^T \mathbf{s}(k-L+1) \tag{10}$$

The BSS problem of the type (10) has been treated in [27].

2.2 Alternative Successor Observation Clustering

The main disadvantage of the section 2.1 method stems from the assumption that the dataset must contain every possible observation/successor pair. As the size of the MISO system increases this assumption requires exponentially larger observation datasets. This section is devoted to an alternative method for the treatment of systems where larger observation datasets are not available. First, observe that for any $r \in \mathcal{X}$ the *centered successors*:

$$c_i = y_i(r) - \bar{y}(r) = \mathbf{h}_0^T \mathbf{b}_i^{(n)} \qquad i = 1, \cdots, 2^n \tag{11}$$

are independent of r. Thus every observation has the same set of centered successors. We shall refer to the set $C = \{c_i;\ i = 1, \cdots, 2^n\}$ as the **centered successor constellation set** of system (2). C can be easily computed by first obtaining $\mathcal{Y}(r)$, for any r, and then subtracting the mean $\bar{y}(r)$ from each element $y_i(r) \in \mathcal{Y}(r)$. Note that C is symmetric in the sense that $c \in C \Leftrightarrow -c \in C$.

Now, for every observation value $r = x(k) \in \mathcal{X}$ we have

$$x(k) = \mathbf{h}_0^T \mathbf{s}(k) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) \tag{12}$$

$$r = \mathbf{h}_0^T \mathbf{b}_j^{(n)} + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r), \qquad \text{some } j \tag{13}$$

$$= c_j + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r), \qquad \text{some } j \tag{14}$$

Furthermore, due to the symmetry of the constellation set, there exists a "dual" observation value $r^d \in \mathcal{X}$ (Fig. 1) such that

$$r^d = -c_j + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{15}$$

$$r^d = r - 2c_j \tag{16}$$

For every observation $r \in \mathcal{X}$, there exists a unique index $j \in \{1, \cdots, 2^n\}$ such that $r - 2c_j \in \mathcal{X}$. Indeed, assume that there existed a different index $\zeta \in \{1, \cdots, 2^n\}$, $\zeta \neq j$, such that $r' = r - 2c_\zeta \in \mathcal{X}$, so

$$r' = c_j - 2c_\zeta + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r)$$

$$= \mathbf{h}_0^T (\mathbf{b}_j^{(n)} - 2\mathbf{b}_\zeta^{(n)}) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r)$$

Now, by Assumption 1, there is a unique binary L-tuple $\{\bar{\mathbf{b}}_0(r'), \cdots, \bar{\mathbf{b}}_{L-1}(r')\}$ which generates the observation r' according to (5), so $(\mathbf{b}_j^{(n)} - 2\mathbf{b}_\zeta^{(n)}) = \bar{\mathbf{b}}_0(r)$, is a binary vector. However, this is impossible, unless $\zeta = j$, thus the proof.

Then the dual value r^d can be identified by testing all $r-2c_j$, $j=1,\cdots,2^n$, for membership in the observation space $\mathcal{X}$. Let us now replace $x(k)$ by the average of r, r^d, to obtain

$$\tilde{x}^{(2)}(k) = (r+r^d)/2 = \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{17}$$

Note that $\mathbf{b}_i(r) = \mathbf{s}(k-i)$, so

$$\tilde{x}^{(2)}(k) = \sum_{i=1}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) \tag{18}$$

Eq. (18) describes a new, shortened MISO system,

Assumption 2 *For* **only one** $r_0 \in \mathcal{X}$*, there exist at least* 2^n k_i, $i=1,\cdots,2^n \in \{1,2,\cdots,K\}$ *such that* $x(k_i)=r_0$, $x(k_i+1)=y_i(r_0)$, $i=1,\cdots,2^n$. *In addition to that, every possible value of* $\mathcal{X}$ *exists at least once in the dataset.* ■

Using Assumption 2 it is obvious that we can compute the centered successor constellation C from the set $\mathcal{Y}(r_0)$. We shall show that there exists, as well, a complete constellation set for the deflated system (18) for some observation $r_0^{(2)}$. Indeed, for all k_i in Assumption 2, $\mathbf{s}(k_i-j)=\bar{\mathbf{b}}_j(r)$, $j=1,\cdots,L-1$, and $\mathbf{s}(k_i)$ takes all 2^n possible values, for $i=1,\cdots,2^n$. So the set

$$\tilde{x}^{(2)}(k_i+1) = \sum_{j=1}^{L-1} \mathbf{h}_j^T \mathbf{s}(k_i-j+1) = \mathbf{h}_1^T \mathbf{s}(k_i) + \sum_{j=1}^{L-2} \mathbf{h}_{j+1}^T \bar{\mathbf{b}}_j(r)$$

contains all 2^n the successors of

$$\tilde{x}^{(2)}(k_i) = \sum_{j=1}^{L-1} \mathbf{h}_j^T \mathbf{s}(k_i-j) = \sum_{j=1}^{L-1} \mathbf{h}_j^T \bar{\mathbf{b}}_j(r) = r_0^{(2)}$$

By induction, it is straightforward to show that a complete set of successors (a complete constellation) exists in all deflated systems.

Summarizing the above results, our second method for obtaining the deflated system (18) is described below:

Algorithm 2 (SOC-2.)

Step 1. *Locate an observation value* r_0 *for which* 2^n *distinct successors* $y_i(r_0)$, $i=1,\cdots,2^n$, *exist in the dataset*

Step 2. *Compute the successor constellation set* C *according to* (11)

Step 3. *For every observation* $r=x(k)$ *find the (unique) value* j *for which* $r-2c_j \in \mathcal{X}$. *Call* $r^d = r-2c_j$.

Step 4. *Replace* $x(k)$ *by* $(r+r^d)/2$

Again, the $L-1$ times repetition of this algorithm will reduce the system into a memoryless one

$$\tilde{x}^{(L)}(k) = \mathbf{a}_{L-1}^T \mathbf{s}(k) \tag{19}$$

2.3 Comparing the two methods

The main advantage of SOC-2 over SOC-1 is that it needs quite smaller datasets. In order to estimate the data size reduction, we created large datasets and we examined the minimum dataset that verifies the assumptions for both methods. Our tests were performed on a variety of MISO systems with different numbers of sources and filter lengths. We tested 100 datasets per system and we kept the worst case, i.e., the maximum dataset length. The results are shown in Table 1, where we define:

$$Gain = 100 \times \frac{K_{SOC-1} - K_{SOC-2}}{K_{SOC-1}} \tag{20}$$

with K_{SOC-1}, K_{SOC-2} being the required dataset size for SOC-1 and SOC-2 respectively. Clearly, as the system becomes more complex, the gain increases. For SISO systems the gain is around 50%, while for more than 2 sources the gain can go up as high as 90%. We note that the experiments were designed using randomly generated binary independent sources.

3 Solvability

In [28] Li *et al.* introduced the notion of bi-independence in binary sense investigating the solvability of an instantaneous BSS problem with binary sources.

Definition 2 *The vectors $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n$ are called bi-independent if there does not exist a set of numbers $c_i \in \{-1, 0, 1\}$ not all zeros, such that:*

$$\sum_{i=1}^{n} c_i \mathbf{a}_i = \mathbf{0} \tag{21}$$

Bi-independence is closely related to the notion of well-posedness of system (2) introduced by the same authors in [29]. We will use the notion of bi-independence in order to investigate the validity of the equivalence assumption.

Any MISO system can be transformed to an equal SISO as follows:

$$\begin{aligned} x(k) &= \sum_{i=0}^{L-1} \mathbf{h}_i^T \mathbf{s}(k-i) \\ &= \sum_{j=1}^{n} \sum_{i=0}^{L-1} h_{i,j} s_j(k-i) \end{aligned} \tag{22}$$

where $h_{i,j} \in \mathbb{R}$ is the j-the element of the i-th filter.

Now consider an observation value r such that the equivalence assumption is not valid:

$$\begin{aligned} r &= x(k) = \sum_{j=1}^{n} \sum_{i=0}^{L-1} h_{i,j} s_j(k-i) \\ &= x(l) = \sum_{j=1}^{n} \sum_{i=0}^{L-1} h_{i,j} s_j(l-i) \end{aligned} \tag{23}$$

and $[s_1(k), ..., s_1(k+1-L), s_2(k), ..., s_n(k+1-L)] \neq [s_1(l), ..., s_1(l+1-L), s_2(l), ..., s_n(l+1-L)]$

$$\begin{aligned} 0 &= x(k) - x(l) \\ &= \sum_{j=1}^{n} \sum_{i=0}^{L-1} h_{i,j}(s_j(k-i) - s_j(l-i)) \\ &= 2 \sum_{j=1}^{n} \sum_{i=0}^{L-1} h_{i,j} \delta_{s(k,l)}(i) \end{aligned} \tag{24}$$

where $\delta_{s(k,l)}(i) = \frac{1}{2}\{s_j(k-i) - s_j(l-i)\}$ $i = 0, ..., L-1$. It is straightforward to confirm that $\delta_{s(k,l)}(i) \in \{-1, 0, +1\}$ $i = 0, ..., L-1$ and $\exists\, i \in \{0, ..., L-1\} : \delta_{s(k,l)}(i) \neq 0$. This means that the mixing parameters are bi-dependent scalars.

Proposition 1 *The equivalence assumption is valid if and only if the mixing parameters $h_{i,j}$, $j = 1, \cdots, n$, $i = 0, \cdots, L-1$, of the MISO system are bi-independent.*

Proposition 1 is true with probability 1, when the mixing parameters h_{ij} are randomly generated from a continuous set of values, since the cases where these parameters are bi-dependent are discrete. However, there are some simple cases where we bi-dependence occurs, such as, for example, when one or more parameters h_{ij} are zero, or when $h_{i_1,j_1} = \pm h_{i_2,j_2}$, for some i_1, i_2, j_1, j_2.

4 Blind Separation of Noisy Binary MISO Systems

It is easy to identify the presence of noise since it generates a continuous observation space as opposed to the discrete space of the noiseless case. The noisy observation at time k is

$$x(k) = r + e(k) \tag{25}$$

where r is a prototype in $\mathcal{X}$ and $e(k)$ is the noise component.

For any $r \in \mathcal{X}$ let us define the following sets:

- the set $\mathcal{T}(r) = \{k : x(k) = r + e(k)\}$ which includes all the noisy observations that are distorted versions of r, and
- the set $\mathcal{O}(r) = \{k : r = \arg\min_{\bar{r} \in \mathcal{X}} |x(k) - \bar{r}|\}$ which includes the noisy observations closest to r.

There are two cases:

1. When $\mathcal{T}(r) = \mathcal{O}(r)$ $\forall\, r \in \mathcal{X}$ then the noise power is small and the problem can be treated as a noiseless one after the application of clustering techniques (see Fig. 2 (a)).
2. If $\exists\, r \in \mathcal{X} : \mathcal{T}(r) \neq \mathcal{O}(r)$, then the noise power is significant and the problem requires more complex treatment than simple clustering. In this case for some k, the sample $x(k)$ generated from prototype r_1 is closer to prototype r_2, ie. $k \in \mathcal{T}(r_1)$, while $k \in \mathcal{O}(r_2)$.

Unfortunately, $\mathcal{T}(r)$ is unknown since it depends on the noise component which is unobservable. As a consequence, it is not trivial to find which of the above cases describes more accurately the observed dataset. In the next section we will describe an efficient clustering procedure for the approximation of the true r with an estimated $\hat{r}$, using the distorted observations $x(k)$. Then we will show, that the scanning of the estimated noiseless observations sequence can reveal if further treatment with a data correction step is needed.

4.1 Data Clustering

Whenever we treat datasets distorted with low levels of noise conventional data clustering methods are sufficient for the estimation of the observation prototypes and the reconstruction of the distorted dataset. In cases of heavily distorted datasets the data clustering step consists of the first step in a complex correction method as the one we will describe in the next section. The two steps in the proposed scheme are: a) Determine the number of classes in the observation dataset, and b) Estimate a noiseless observation for every class.

In general, the number of classes can be determined by inspecting the histogram of the output and counting the number of peaks. This is valid for large datasets where the shape of every gaussian bell can be traced correctly. However in many experimental scenarios and true applications this is not the case. In order to find the number of classes (the cardinality of the observations) we created a robust technique based on the equiprobability of appearance of every observation, In general, the first and the last gaussian "bell" in the histogram are best distinguished. Consequently, we locate the peak as the first local maximum in the histogram. We estimate the area of the bell as equal to twice the area of the histogram from the start and up to the peak. Since all bells have equal areas, the number of classes can be approximated by dividing the total area of the histogram by the area of the first bell. The exact number of classes is the closest power of 2 to our approximation (the cardinality of the observation set is 2^{nL}). In order to ensure that small perturbations in the histogram will not be erroneously considered as the first peak, the histogram is firstly smoothed. In figure 3 we show the histogram of three heavily contaminated datasets.

The presented observation sets are created on a MISO system with three source signals mixed with a filter of length two ($n = 3, L = 1$). The correct observation classes are 64. Unfortunately, most of them are grouped in large bells. The counting of the gaussian bells would yield erroneous results. We can count 16, 18, and 11 peaks in (a), (b), and (c) histogram respectively. However, the proposed technique succeeds in estimating the correct number of classes in each of the selected examples.

Once we estimated the correct number of classes, if the observation prototypes are equiprobable then we can assume that every cluster should contain approximately the same number of noisy observations. If the noisy observation set has K observations, then each class should contain $l = \frac{K}{M}$ observations (remember $M = |\mathcal{X}|$). If the observations are sorted, then the first class should contain the first l observations, the second one should contain the next l observations, etc. This procedure will form M classes i.e. one for each r.

If the classes are accurately defined then a method based on the mean observation value of each class could be a good choice for the estimation of $\hat{r}$. However usually this is not the case. Outliers may lie in the limits of each class, yielding large errors in any mean based method. A better choice for the estimation of the center of every class as follows:

(a) sort the observed values (b) cut the sorted list in M parts of l samples each (c) let the

center $\hat{r}_i$ of part i to be the median value of this part. For the rest of the paper the estimated observation space will be noted $\hat{\mathcal{X}}$.

Using the estimated prototypes $\hat{r}_i$, we may attempt to "denoise" the observation sequence by replacing every noisy observation $x(k)$ with the closest noiseless prototype: $\forall k \in \mathcal{O}(\hat{r}) : x(k) \leftarrow \hat{r}$. Since we use a nearest neighbor rule, the boundaries lo_i, hi_i of class i are in the middle between class centers:

$$[lo_i, hi_i] = \begin{cases} [-\infty, (\hat{r}_i + \hat{r}_{i+1})/2] & \text{if } i = 1 \\ [(\hat{r}_{i-1} + \hat{r}_i)/2, (\hat{r}_i + \hat{r}_{i+1})/2] & \text{if } 1 < i < |\mathcal{X}| \\ [(\hat{r}_{i-1} + \hat{r}_i)/2, \infty] & \text{if } i = |\mathcal{X}| \end{cases} \tag{26}$$

Every observation $x(k)$ between lo_i and hi_i will be replaced by the estimated $\hat{r}_i$. The noise component may move the observation outside the boundaries of the correct class. Even if all prototypes are perfectly estimated, i.e. $\hat{r}_i = r_i$, the sample $x(k)$ may still be misclassified. However if the clustering step concludes with the correct dataset, then the data can be separated using SOC-2, since assumption 2 is fulfilled.

A simple test that can reveal if there are misplaced observations consists of creating the successor set $\mathcal{Y}(\hat{r})$ for every $\hat{r}$. The cardinality of this set must be equal to 2^n. If $\exists\, \hat{r}_1 \in \hat{\mathcal{X}} : |\mathcal{Y}(\hat{r}_1)| > 2^n$, then this means that the successor set of $\hat{r}_1$ contains more than expected observations. So at least one inaccurately clustered observation exists in the dataset. In that case, an additional step for the correction of the clustered dataset is needed. In the next section we will develop a technique for the correction of such cases. On the contrary, if every class includes the correct data, the dataset can be directly processed using the algorithm presented in section 2.

4.2 Data Correction

In the previous section we showed that the reconstructed sequence may contain mis-clustered observations. In this section we will attack the problem using a set of novel methods for data correction. First, we will consider the case of isolated erroneous observations in the dataset and then we will describe a method for the treatment of severely contaminated datasets. The method is robust to single errors. A single error in the observation set will generate an isolated outlier which can be easily located and corrected. We locate the presence of an outlier $x(k+1)$ because it lies outside the successor set $\mathcal{Y}(r)$ of the value $r = x(k)$. Then suppose that $x(k+1)$ is preceded and followed by two correct values $x(k)$ and $x(k+2)$, respectively. We claim that there is a unique correct value for $x(k+1)$. Indeed, we have $x(k) = \mathbf{h}_0^T \mathbf{s}(k) + ... + \mathbf{h}_{L-1}^T \mathbf{s}(k-L+1)$, and $x(k+2) = \mathbf{h}_0^T \mathbf{s}(k+2) + ... + \mathbf{h}_{L-1}^T \mathbf{s}(k-L+3)$. According to Assumption 1 any L-tuple of the values $\mathbf{s}(k+2), \mathbf{s}(k+1), \mathbf{s}(k), ..., \mathbf{s}(k-L+1)$ forms a unique observation, and since the value $x(k+1) = \mathbf{h}_0^T \mathbf{s}(k+1) + ... + \mathbf{h}_{L-1}^T \mathbf{s}(k-L+2)$ is composed of all-except-two of the above values it is also unique. In fact this argument can be extended to show that we can tolerate a sequence of at most L-1 consecutive outliers. In practice, we can easily construct the dendrogram from $x(k)$ to all possible values of $x(k+2)$ and select the unique path that terminates at the actual $x(k+2)$. This path passes from the correct value of $x(k+1)$.

In case that the processed dataset is distorted so heavily that the above scheme is insufficient, then we can use a data correction techniques based on the Viterbi algorithm. It is important to distinguish the processed signal from the error-free signal. Although both signals are composed of the estimated prototypes $|\hat{\mathcal{X}}|$, the latter one is the correct observation sequence, while the

former signal contains some errors in the sequence. In the following the error-free sequence and the erroneous one will be noted x_f and x_e respectively. If we consider x_f as a hidden signal which emits at every instant the visible signal x_e, then we can model the system as a first order Hidden Markov Model (HMM). The problem of correcting the data is transformed into the problem of decoding the HMM i.e. recovering the hidden signal using only the visible one. A well known algorithm for decoding a HMM is the Viterbi algorithm ([30] chapter 3). The Viterbi algorithm determines the optimal path in a trellis diagram, i.e. a sequence of discrete states, where at every stage of the diagram M potential states are considered. In our application, the k-th stage corresponds to observation $x_f(k)$ and each one of the states in this stage corresponds to the $M = 2^{nL}$ possible values of $x_f(k)$. The method uses two types of probabilities: (i) the transition probability from state $\hat{r}_i$ at instant $t-1$ to state $\hat{r}_j$ at the next instant t, and (ii) the emission probability that the visible value $x_e(t)$ is emitted while the system is at the hidden state $\hat{r}_i$ at time t. The Viterbi algorithm calculates the most probable path of hidden states based on these probabilities given the visible sequence x_e. In the following we will construct an iterative scheme based on the Viterbi algorithm for the dataset correction.

First, we must estimate the probability of succession between every observation prototype. Remember that in noisy datasets the cardinality of the successor set of $\hat{r}$ will be bigger than expected: $|\mathcal{Y}(\hat{r})| \geq 2^n$ i.e. it contains the correct successors plus a number of incorrectly clustered observations. Normally, the frequency of appearance of the correct successors is much higher than the one of the wrong successors. Consequently, if we collect the 2^n successors with the highest frequency of appearance in the successors set $\mathcal{Y}(\hat{r})$, for every observation $\hat{r} \in \hat{\mathcal{X}}$, then we can form the correct successors set $\mathcal{Y}_{2^n}(\hat{r})$. It is straightforward that $\mathcal{Y}_{2^n}(\hat{r}) \subseteq \mathcal{Y}(\hat{r})$. Conversely, if the frequency of appearance of an erroneous successor is higher that one of the correct successors, then the problem is unsolvable.

In the scenarios that we tested, we assumed the equiprobability of appearance for every potential successor:

$$p_{ij}^{(t)} = p(x_f(k) = \hat{r}_i | x_f(k-1) = \hat{r}_j) = \begin{cases} \frac{1}{2^n} & \text{if } \hat{r}_i \in \mathcal{Y}_{2^n}(\hat{r}_j) \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

The transition probabilities $p_{ij}^{(t)}$ form a $2^M \times 2^M$ matrix $\mathbf{P}^{trans}$. If the observation prototypes that correspond to every row and column are sorted then the transition matrix has some interesting properties to investigate (for the proofs see the appendix).

Property 1. Every row of $\mathbf{P}^{trans}$ has exactly 2^n non-zero elements;

Property 2. Every column of $\mathbf{P}^{trans}$ has exactly 2^n non-zero elements;

Property 3. The rows of $\mathbf{P}^{trans}$ corresponding to opposite prototypes (the same prototype with opposite signs) are " *mirror*" vectors[1].

These properties will be used in the data correction scheme.

Secondly, we must calculate the probability that a hidden observation $x_f(k)$ may emit the visible observation $x_e(k)$. The distribution of the observation within a class is gaussian, due to the noise component, thus the emission probability is:

[1]We call $a = [a_1, \cdots, a_n]^T$ and $\bar{a} = [a_n, \cdots, a_1]^T$ "mirror" vectors since the one is the mirroring image of the other

$$
\begin{aligned}
p_{ij}^{(e)} &= p(x_e(k) = \hat{r}_i | x_f(k) = \hat{r}_j) \\
&= \int_{lo_i}^{hi_i} \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(\frac{(t-\mu_j)^2}{2\sigma_j^2}\right) dt
\end{aligned} \tag{28}
$$

i.e. it equals the probability that the realization of a random process following $N(\mu_j, \sigma_j)$, can be found between the bounds of another class ($[lo_i, hi_i]$). The mean and the standard deviation values are calculated using the $\mathcal{O}(r)$ sets, since the correct $\mathcal{T}(r)$ are unobservable:

$$
\mu_j \simeq E(x_e(k)), \qquad k \in \mathcal{O}(r_j) \tag{29}
$$

$$
\sigma_j \simeq E(x_e(k) - \mu_j)^2, \qquad k \in \mathcal{O}(r_j) \tag{30}
$$

Equipped with probabilities (27) and (28) the Viterbi algorithm can decode the HMM. We constructed a two step iterative scheme for the correction of the observation sequence: (a) the Viterbi step, and (b) the transition matrix step. The former step consists of a repetition of the Viterbi algorithm over a number of times. The transition probability is updated in every iteration with the corrected sequence. The latter step consists of ensuring that the transition matrix Properties 1-3 are restored after the Viterbi step. This is achieved as follows: we, first, locate the column with the highest 2^n values. We create the ideal transition vector: a) we replace the 2^n highest values with $1/2^n$ (the correct transition probability); b) the rest of the vector is set to zero. This vector should appear in the transition matrix 2^n times. Consequently, we search and locate the $2^n - 1$ vectors in the matrix that are the most similar to the ideal one in the least squares sense, and we replace them with it. Finally, we create the ideal "mirror" vector and proceed similarly. The same scheme is applied until all the columns of the transition matrix are replaced ($M/2^{n+1}$ times).

The scheme concludes with the final decoding using the Viterbi algorithm. The separation of the signal is performed using the first algorithm.

We can summarize the proposed algorithms in the following general scheme:

General Algorithm

1. Investigate the noise presence in the system.
2. In the noiseless case follow SOC-2 and exit
3. In the noisy case proceed to the Data Clustering Method
   (a) Estimate the number of classes, by counting the peaks of the gaussian bells.
   (b) Calculate $\hat{r}_i$ for each class using the statistical median of the sorted groups
   (c) Substitute every observation in class $\mathcal{O}(\hat{r}_i)$ with $\hat{r}_i$
4. Create the Successors Set $\mathcal{Y}(\hat{r}_i)$ for each $\hat{r}_i$
5. If the cardinality of $\mathcal{Y}(\hat{r}_i)$ is equal for every $\hat{r}_i$ then follow SOC-1 and exit
6. Else proceed to the Data Correction Method
   (a) Estimate the transition probabilities $p(x_e(k) = \hat{r}_2 | x_e(k-1) = \hat{r}_1)$ using Eq.(27)
   (b) Calculate the probability $p(x_f(k) = \hat{r}_j | x_e(k) = \hat{r}_i)$ using Eq.(28)
   (c) Find the most probable sequence of hidden states using the visual states [30]
   (d) Correct the observation dataset.
7. Follow SOC-1 and exit

5 Results

In the experiments that follow we measure two types of errors: (a) misclassified observations and (b) final Bit Error Rate (BER). In order to account for the first type of errors, we introduce the Observation Error Rate (OER) which is defined as the ratio of the misclassified samples over

the total observations in the dataset.

We tested the proposed methods in three experiment sets. In the first set, the same filter was used in every experiment. In the second set, each experiment was conducted using a different filter selected randomly from a family of filters. In both cases we performed 1000 Monte Carlo experiments using 10000 observation samples in each experiment. The third experiment consist of the mixing of three binary images in a MISO framework.

As discussed before, a system with sparse prototypes can tolerate higher noise levels, than one with dense prototypes. In order to estimate the significance of the noise with respect to the distance between the clusters, we introduce the *Minimum Normalized Cluster Distance* (MNCD) defined as the ratio of the smallest distance between the observation prototypes and the noise standard deviation:

$$MNCD = \frac{\min_{i \neq j} |\hat{r}_i - \hat{r}_j|}{\sigma_e} \tag{31}$$

Clearly $MNCD \geq 0$. When $MNCD \to 0$ the system is highly influenced of the noise component, while for $MNCD \to \infty$ the noise is negligible. MNCD is proportional to the SNR, since $SNR = \sigma_s/\sigma_e$, and so

$$MNCD = \frac{\min_{i \neq j} |\hat{r}_i - \hat{r}_j|}{\sigma_s} SNR \tag{32}$$

5.1 Experiment Set 1

In the first set of experiments two sources were mixed with the following filter: $\mathbf{h}_1 = [0.5360, 0.9212]^T$ and $\mathbf{h}_2 = [0.7647, 0.7455]^T$. We created 1000 datasets and injected various levels of noise. The noisy observation datasets were clustered and corrected, prior to applying the source estimation method. In figure 4 we show the Observation Error Rate (OER) before and after the data correction step.

The data correction step decreases the OER for every noise level that we tested. We must note that for higher than 25 dB noise level the observation error is practically eliminated by this process. Even in the case of datasets contaminated by 20 dB of noise, the OER is around 5%. The dashed line in the figure marks the point where $MNCD = 1$, i.e the point where the noise variance is equal to the minimum distance between prototypes. The MNCD increases with the SNR (from left to right).

In figure 5 the result of the signal estimation is presented. The BER behavior is similar to the OER behavior in Figure 4. Above 20 dB the estimation is very accurate, while above 25 dB the estimation is perfect.

5.2 Experiment Set 2

In the MISO model used in the second set of experiments, two binary inputs are mixed with a filter of length two ($n = 2$ and $L = 1$). Every one of the 1000 datasets was created by mixing the inputs using a different filter realization. The filter taps are given by the formula $h_{i,j} = (-1)^{\alpha} * \beta$, where α is a random binary process taking values in $\{0, 1\}$, and β follows the uniform distribution in the segment $[0.1, 1.1]$ ($\beta \sim U(0.1, 1.1)$). Various levels of noise were injected in the datasets.

In Figure 6 we show the observation error rate before and after the data correction step for various noise levels.

The Observation Error Rate is plotted against the MNCD in order to show the importance of this measure in the method performance. It can be noted that when the $MNCD > 0.5$ i.e the noise standard deviation is double than the minimum prototypes distance, the OER is less than 5%. As the $MNCD$ factor increases the OER decreases for any noise scenario. The usefulness of the $MNCD$ factor can be witnessed from the almost linear relation between the OER and $MNCD$ of the datasets before the data correction step. The SNR would not reveal that relation.

In Figure 7 we present the performance of the blind source estimation using the BER for various noise levels. The estimation is quite accurate for SNR higher than 25 dB. However, it must be outlined that the method success is equally due to the signal estimation method and the data correction method. The overall accuracy would not be that high, if the OER was left higher than 15%

5.3 Experiment Set 3

The third experiment involves three binary signals corresponding to the column-wise arrangement of an equal number of two-tone images: a thresholded image of "cameraman", a binary text image and a chequered image. The source signals (top row of Fig. 9) are mixed together into a single observation signal (Fig. 8). The reconstructed images (bottom row of Fig. 9) are identical to the true sources.

6 Conclusion

In this paper we presented a new technique for the blind separation of MISO systems when the source signals are binary. The approach is based on the geometrical patterns of pairs of consecutive observations. We show that the collection of all the potential successor observation follows a standard constellation pattern. The exploitation of the properties of this constellation pattern is the basic idea behind the two methods presented here. These methods perform filter deflation, i.e. they create MISO system with on tap less than the initial one. If this procedure is applied recursively, we end up with an instantaneous system which can be directly solved. The treatment of noise is also presented in detail. The method was successfully tested under various noise conditions in a Monte Carlo framework and with two-tone image mixtures.

Proof 1 *(Property 1). Equation* (27) *shows that every observation prototype (row) has* 2^n *successors (non-zero elements)*

Proof 2 *(Property 2). Every observation* $x(k)$ *is the successor of the observation* $x(k-1)$*. If*

$$x(k) = \sum_{i=0}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{33}$$

then the ***previous observation****,* $x(k-1)$ *is:*

$$x(k-1) \;=\; \mathbf{h}_{L-1}^T \mathbf{s}(k-1) + \sum_{i=0}^{L-2} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \tag{34}$$

Since $\mathbf{s}(k-1)$ *is an* n*-dimensional binary antipodal vector,* $x(k-1)$ *can take one of the following* 2^n *possible values:*

$$y_p(r) = \mathbf{h}_{L-1}^T \mathbf{b}_p^{(n)} + \sum_{i=0}^{L-2} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r), \quad p = 1, \cdots, 2^n \tag{35}$$

As a consequence every observation $x(k)$ *may be the successor of* 2^n *observations* $x(k-1)$*. Consequently every column* ($x(k-1)$) *of the transition matrix has* 2^n *non-zero elements* ($x(k)$)*.*

Proof 3 *(Property 3). This means that if* $c_r \in \mathcal{Y}(r)$ *(ie. is any of the successors of* r*), then* $-c_r \in \mathcal{Y}(-r)$*.*

First, check that for any observation prototype $r \in \mathcal{X}$*, the opposite prototype* $-r \in \mathcal{X}$*:*

$$r = \sum_{i=0}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_i(r) \tag{36}$$

$$\begin{aligned} -r &= \sum_{i=0}^{L-1} \mathbf{h}_i^T (-\bar{\mathbf{b}}_i(r)) \\ &= \sum_{i=0}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}'_i(r) \end{aligned} \tag{37}$$

where $\bar{\mathbf{b}}'_j(r) = -\bar{\mathbf{b}}_i(r)$*.*

Second, if $c_r \in \mathcal{Y}(r)$*, and* $c_{-r} \in \mathcal{Y}(-r)$ *then:*

$$c_r = \mathbf{h}_0^T \bar{\mathbf{b}}_r + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}_{i-1}(r) \tag{38}$$

and

$$c_{-r} = \mathbf{h}_0^T \bar{\mathbf{b}}_{-r} + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}'_{i-1}(r) \tag{39}$$

where $\bar{\mathbf{b}}_r$ *and* $\bar{\mathbf{b}}_{-r}$ *is any row of the matrix defined in Equation* (3)*.*

Now check that

$$-c_r = \mathbf{h}_0^T (-\bar{\mathbf{b}}_r) + \sum_{i=1}^{L-1} \mathbf{h}_i^T (-\bar{\mathbf{b}}_{i-1}(r)) \tag{40}$$

$$= \mathbf{h}_0^T (-\bar{\mathbf{b}}_r) + \sum_{i=1}^{L-1} \mathbf{h}_i^T \bar{\mathbf{b}}'_{i-1}(r) \tag{41}$$

where $-\bar{\mathbf{b}}_r$ *is a row of the matrix defined in Equation* (3)*. Consequently* $-c_r \in \mathcal{Y}(-r)$*.*

References

[1] J.-F. Cardoso, "Source Separation using higher order moments," in *Proc. IEEE ICASSP,*, vol. 4, Glasgow, U.K., 1989, pp. 2109–2112.

[2] P. Comon, "Independent component analysis, a new concept?" *Signal Processing*, vol. 36, pp. 287–314, 1994.

[3] C.-C. Feng and C.-Y. Chi, "Performance of Cumulant Based Inverse Filters for Blind Deconvolution," *IEEE Trans. Signal Process.*, vol. 47, no. 7, pp. 1922–1935, July 1999.

[4] L. Tong, Y. Inouye, and R.-W. Liu, "Waveform-Preserving Blind Estimation of Multiple Independent Sources," *IEEE Transactions on Signal Processing*, vol. 41, pp. 2461–2470, 1993.

[5] C. B. Papadias, "Globally Convergent Blind Source Separation Based on a Multiuser Kurtosis Maximization Criterion," *IEEE Trans. on Signal Processing*, vol. 48, no. 12, pp. 3508–3519, December 2000.

[6] D. Pham, "Blind Separation of Instantaneous mixture of sources via an independent component analysis," *IEEE Trans. Signal Processing*, vol. 44, pp. 2768–2779, November 1996.

[7] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A Blind Source Separation Technique Using Second-Order Statistics," *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 434–444, Febrary 1997.

[8] A. Chevreuil and P. Loubaton, "MIMO Blind Second-Order Equalization Method and Conjugate Cyclostationarity," *IEEE Trans. Signal Processing*, vol. 47, no. 2, pp. 572–578, February 1999.

[9] K. I. Diamantaras, A. P. Petropulu, and B. Chen, "Blind Two-Input Two-Output FIR Channel Identification based on Second-Order Statistics," *IEEE trans. Signal Processing*, vol. 48, no. 2, pp. 534–542, February 2000.

[10] T. P. Krauss and M. D. Zoltowski, "Bilinear Approach to Multiuser Second-Order Statistics-Based Blind Channel Estimation," *IEEE Trans. Signal Processing*, vol. 48, no. 9, pp. 2473–2486, September 2000.

[11] X. Li and H. Fan, "QR Factorization Based Blind Channel Identification and Equalization with Second-Order Statistics," *IEEE Trans. Signal Process.*, vol. 48, no. 1, pp. 60–69, January 2000.

[12] K. I. Diamantaras and T. Papadimitriou, "Oriented PCA and Blind Signal Separation," in *Proc. 4th Int. Symposium on Independent Component Analyis and Blind Signal Separation*, Nara, Japan, April 1-4 2003.

[13] K. Anand and V. Reddy, "Blind Separation of Multiple Cochannel BPSK Signals arriving at an Antenna Array," *IEEE Signal Processing Letters*, vol. 2, pp. 176–178, 1995.

[14] A.-J. van der Veen and A. Paulraj, "An Analytical Constant Modulus Algorithm," *IEEE Transactions on Signal Processing*, vol. 44, pp. 1136–1155, 1996.

[15] A.-J. van der Veen, "Analytical Method for Blind Binary Signal Separation," *IEEE Transactions on Signal Processing*, vol. 45, pp. 1078–1082, 1997.

[16] S. Talwar, M. Viberg, and A. Paulraj, "Blind Separation of Synchronous co-channel Digital Signals using an Antenna Array - Part I: Algorithms," *IEEE Trans. Signal Process.*, vol. 44, pp. 1184–1197, May 1996.

[17] K. Abed-Meraim, P. Loubaton, and E. Moulines, "A Subspace Algorithm for certain Blind Identification Problems," *IEEE Trans. Inform. Theory*, pp. 499–511, March 1997.

[18] P. Common and M. Rajih, "Blind Identification of Complex Under-Determined Mixtures," in *Proc. 5th Int. Symposium on Independent Component Analysis and Blind Signal Separation*, Granada, Spain, September 2004, pp. 105–112.

[19] E. Vincent and X. Rodet, "Underdetermined Source Separation with Structured Source Priors," in *Proc. 5th Int. Symposium on Independent Component Analysis and Blind Signal Separation*, Granada, Spain, September 2004, pp. 327–334.

[20] S. Winter, H. Siwada, S. Araki, and S. Makino, "Overcomplete BSS for Convolutive Mixtures based on Hierarchical Clustering," in *Proc. 5th Int. Symposium on Independent Component Analysis and Blind Signal Separation*, Granada, Spain, September 2004, pp. 652–660.

[21] D. Yellin and B. Porat, "Blind Identification of FIR systems excited by discrete alphabet inputs," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1331–1339, March 1993.

[22] N. Al-Dhahir and A. Sayed, "The finite-length Multi-Input Multi-Output MMSE-DFE," *IEEE Trans. Signal Process.*, vol. 48, pp. 2921–2936, October 2000.

[23] Y. Xie and C. Georghiades, "Two EM-type Channel Estimation Algorithms for OFDM with Transmitter Diversity," *IEEE Trans. on Commun.*, vol. 52, pp. 106–115, January 2003.

[24] C. Aldana, E. de Carvalho, and J. Cioffi, "Channel Estimation for Multicarrier Multiple Input Single Output Systems using the EM Algorithm," *IEEE Trans. Signal Process.*, vol. 51, pp. 3280–3292, December 2003.

[25] K. I. Diamantaras and T. Papadimitriou, "Blind Deconvolution of SISO Systems with Binary Source based on Recursive Channel Shortening," in *Proc. 5th Int. Symposium on Independent Component Analysis and Blind Signal Separation*, Granada, Spain, September 2004.

[26] ——, "Blind Deconvolution of Multi-Input Single-Output Systems with Binary Sources," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-05)*, March 2005.

[27] K. I. Diamantaras and E. Chassioti, "Blind separation of n binary sources from one observation: A deterministic approach," in *Proc. Second Int. Workshop on ICA and BSS*, Helsinki, Finland, June 2000, pp. 93–98.

[28] Y. Li, A. Cichocki, and L. Zhang, "Blind Separation and Extraction of Binary Sources," *IEICE Trans. Fundamentals*, vol. E86-A, pp. 580–589, March 2003.

[29] ——, "Blind Source Estimation of FIR Channels for Binary Sources: a Grouping Decision Approach," *Signal Processing*, vol. 84, pp. 2245–2263, 2004.

[30] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. John Wiley & Sons, Inc., 2001.

Table 1: The dataset length that satisfies the assumptions for both SOC-1 and SCO-2 in various profiles and the percentage gain

# Sources	# Taps	SOC-1	SOC-2	Gain (%)
1	2	64	33	48.44
1	3	133	72	45.86
1	4	396	176	55.56
1	5	787	310	60.61
2	1	98	39	60.20
2	2	658	108	83.59
2	3	3228	685	78.78
2	4	11801	2678	77.31
2	5	60443	13399	77.83
3	1	499	148	70.34
3	2	6888	657	90.46
3	3	49994	6349	87.30
4	1	2657	611	77.00
4	2	61605	6206	89.93

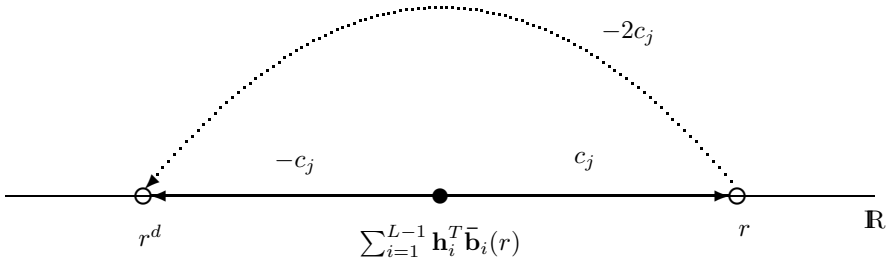


Figure 1: The geometric relation between r and r^d

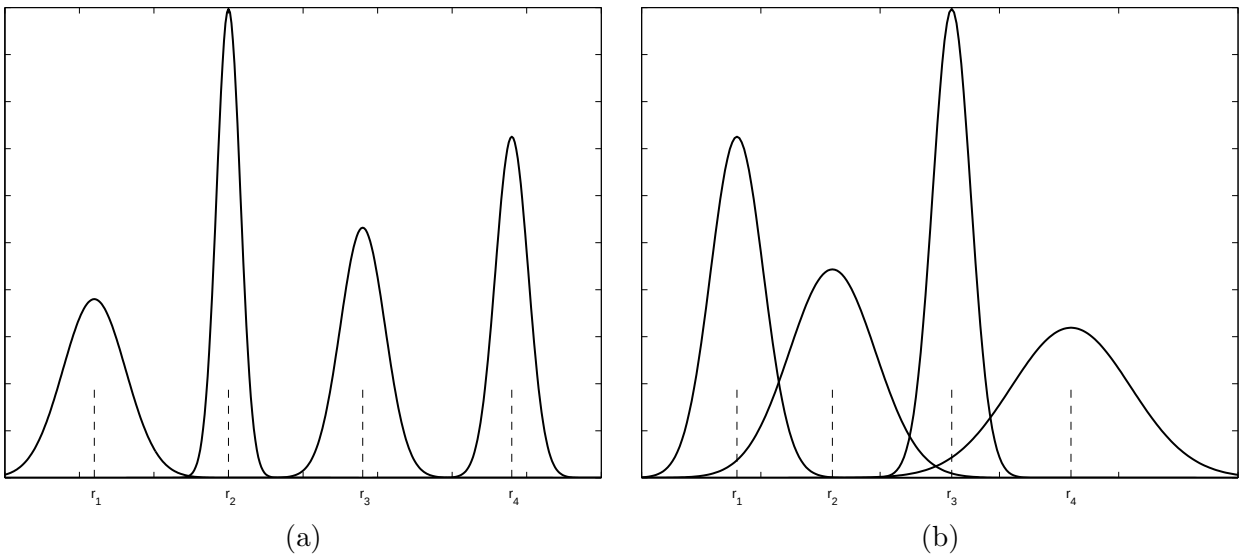
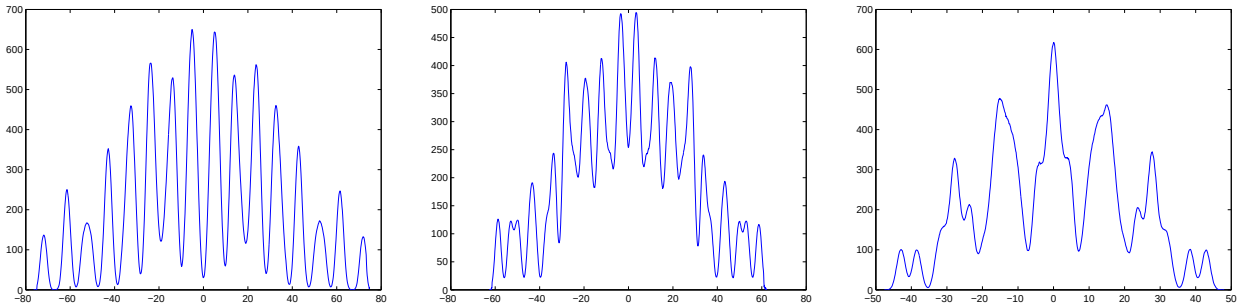


Figure 2: Two cases of MISO systems injected with noise. The noisy observations corresponding to some prototype r form a class with a certain distribution around r. Depending on the noise power (a) the classes do not intersect, or (b) the classes intersect with each other.



(a) (b) (c)

Figure 3: The histograms of three observation sets heavily contaminated by additive gaussian noise. They are the output of a MISO system mixing three binary signals with a two taps filter. A total of 64 gaussian bells are hidden in each of these histograms. The proposed method succeeds in estimating the exact number of classes in each case

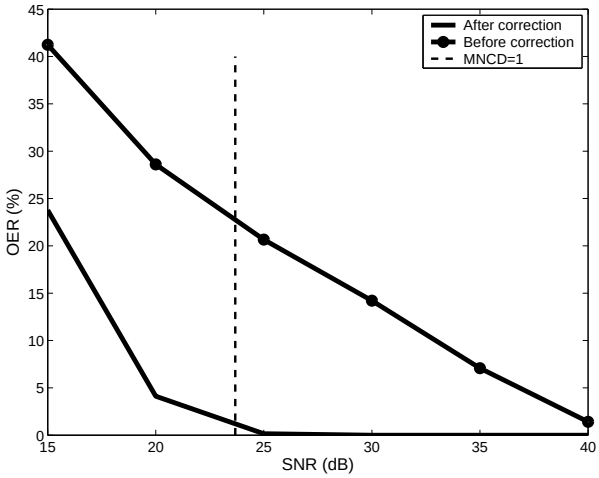


Figure 4: The performance of the data correction step, measured using the OER. The point were the $MNCD = 1$ is indicated with a vertical dashed line.

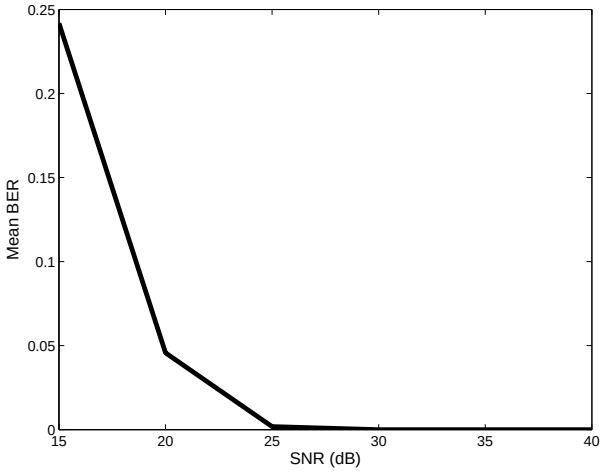


Figure 5: Mean BER for 1000 Monte Carlo experiments and for various SNR levels using the first experiment setup.

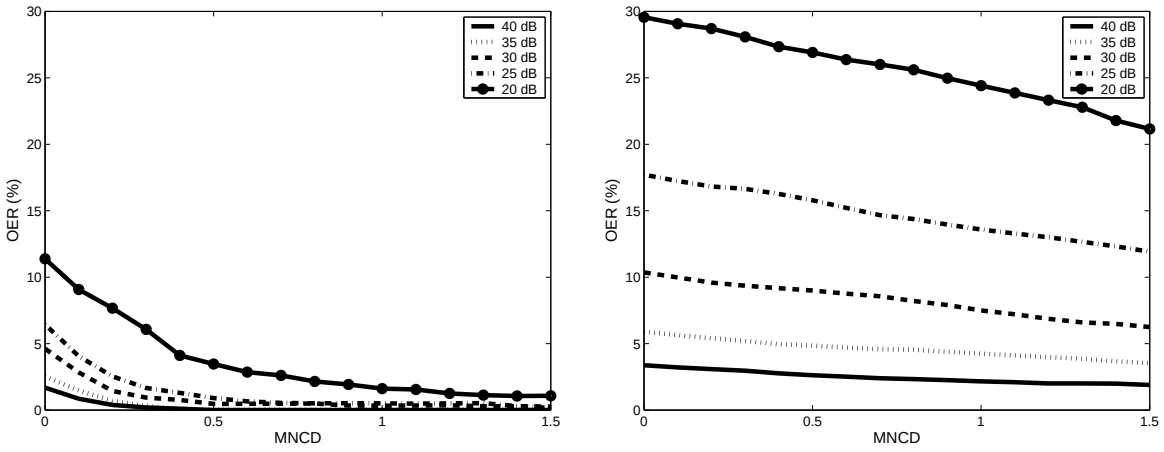


Figure 6: The correction of the dataset using the HMM approach, estimated using the observation error rate on 1000 datasets for noise levels varying from 20dB to 40 dB. Left is the dataset before the data correction step, while the right figure shows the corrected OER dataset

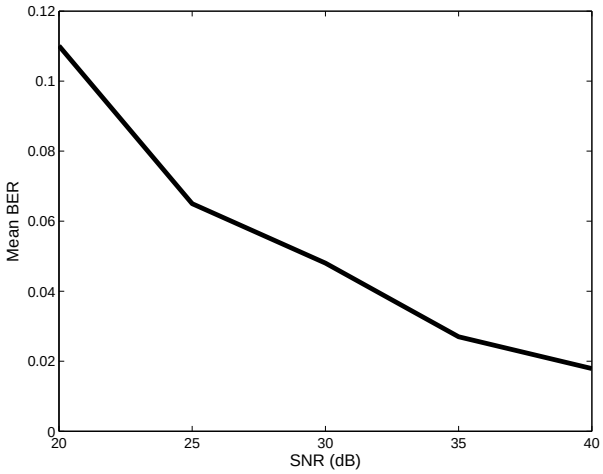


Figure 7: The estimation of the signal accuracy described by the BER for various noise levels

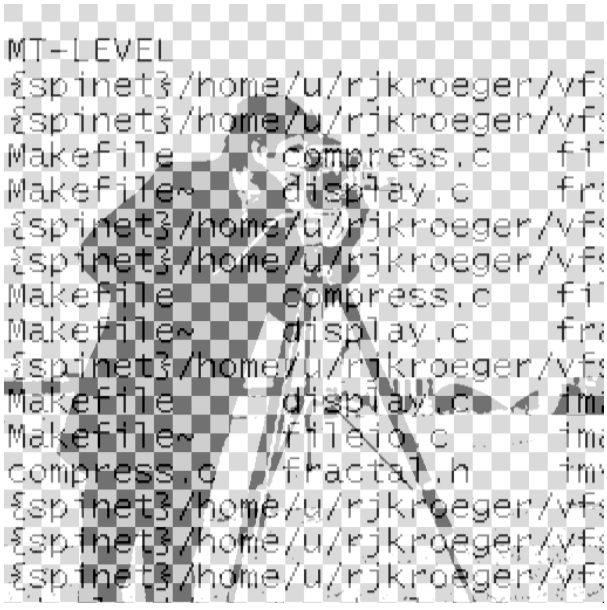


Figure 8: The mixture image

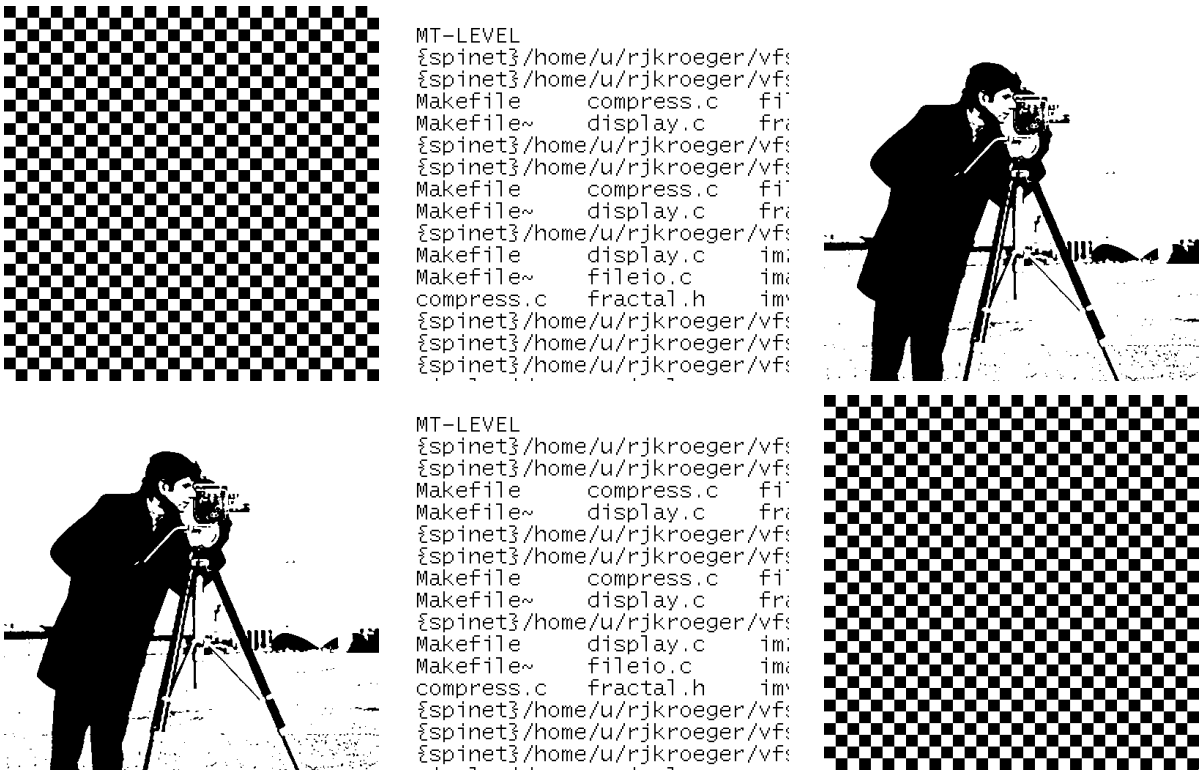


Figure 9: The true images are presented in the first row, while the reconstructed ones can be found in the second row. The reconstructed images are identical to the true images