ABSTRACT

BLIND SEPARATION FOR INTERMITTENT SOURCES VIA SPARSE DICTIONARY LEARNING

by
Annan Dong

Radio frequency sources are observed at a fusion center via sensor measurements made over slow flat-fading channels. The number of sources may be larger than the number of sensors, but their activity is sparse and intermittent with bursty transmission patterns. To account for this, sources are modeled as hidden Markov models with known or unknown parameters. The problem of blind source estimation in the absence of channel state information is tackled via a novel algorithm, consisting of a dictionary learning (DL) stage and a per-source stochastic filtering (PSF) stage. The two stages work in tandem, with the latter operating on the output produced by the former. Both stages are designed so as to account for the sparsity and memory of the sources. To this end, smooth LASSO is integrated with DL, while the forward-backward algorithm and Expectation Maximization (EM) algorithm are leveraged for PSF. It is shown that the proposed algorithm can enhance the detection and the estimation performance of the sources, and that it is robust to the sparsity level and average duration of transmission of the source signals.

BLIND SEPARATION FOR INTERMITTENT SOURCES VIA SPARSE DICTIONARY LEARNING

by
Annan Dong

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering

Helen and John C. Hartmann Department of Electrical and Computer Engineering

May 2019

APPROVAL PAGE

BLIND SEPARATION FOR INTERMITTENT SOURCES VIA SPARSE DICTIONARY LEARNING

Annan Dong

Dr. Alexandar M. Haimovich, Dissertation Advisor — Date
Distinguished Professor of Electrical and Computer Engineering, NJIT

Dr. Osvaldo Simeone, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Ali Abdi, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Jörg Kliewer, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Octavia Dobre, Committee Member — Date
Professor of Electrical and Computer Engineering, Memorial University of Newfoundland, Canada

BIOGRAPHICAL SKETCH

Author: Annan Dong

Degree: Doctor of Philosophy

Date: May 2019

Undergraduate and Graduate Education:

- Doctor of Philosophy in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2019
- Master of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2014
- Bachelor of Science in Electrical Engineering,
  Fudan University, Shanghai, P. R. China, 2011

Major: Electrical Engineering

Presentations and Publications:

A. Dong, O. Simeone, A. Haimovich, J. Dabin, "Sparse Dictionary Learning and Per-source Filtering for Blind Radio Source Separation," *53rd Annual Conference on Information Sciences and Systems (CISS)*, Accepted, 2019.

A. Dong, O. Simeone, A. Haimovich, J. Dabin, "Blind Sparse Estimation of Intermittent Sources over Unknown Fading Channels," *IEEE Transactions on Vehicular Technology*, Submitted, 2018.

You only live once.

Don't wait until you lose something to appreciate it.

ACKNOWLEDGMENT

First, I would like to thank my advisor Dr. Alexander M. Haimovich. It is an honor to be his Ph.D. student. I always find his educational methods inspirational and motivational. I appreciate all his contributions of time, ideas, and countless intuition, insight to teach me how to solve my problems by grasping their essences. In my opinion, there is a famous saying that fits his way to cultivate his students, which goes “Give a man a fish and you feed him for a day; teach a man to fish and you feed him for a lifetime.” I’ve learned the most valuable experience from him that contributes not only to my academic progress, but also to my everyday life.

I also want to thank Dr. Osvaldo Simeone, who has been my co-advisor for two years when I was working on the project of blind source separation. During that period, I’m constantly amazed by how broad his knowledge is and how deep he understands all the problems. I am also thankful for the excellent example he has provided as a dedicate and enthusiastic professor towards academic researches.

Special thanks to Dr. Ali Abdi, Dr. Jörg Kliewer, Dr. Octavia Dobre and Dr. Osvaldo Simeone for actively participating in my committee and spending their time to arrange both my proposal and dissertation defense.

I gratefully acknowledge the funding sources that made my Ph.D. work possible. I was funded by Battelle Memorial Institute for the first two years and the fifth year, and by Booz Allen Hamilton Incorporated for my third and fourth year. Here I would like to thank Jason Dabin, who connects our research group to the sponsors. His advice on how our research should develop and feedback from the sponsors are priceless.

The members of the Center for Wireless Information Processing have contributed immensely to my personal and professional time at NJIT. The group has been a source of friendship as well as good advice and collaboration. My special thanks goes to Nil

Garcia, who has been an excellent mentor for my first year as a Ph.D. student. He guided me through the overwhelming transition from a master student to a Ph.D. student and showed me what is it about to become a Ph.D. student. I also want to thank Ms. Angela Retino and Ms. Kathy Bosco for the paperwork, information, help and those fun center parties. They really make the center feel like home to me.

My time at NJIT was made enjoyable in large part due to the many friends and groups that became a part of my life. I am grateful for time spent with roommates and friends, for the hiking trips into the mountains, for the regular weightlifting and basketball games with the group, and for many other people and memories.

Lastly, I would like to thank my family for all their love and encouragement. Many thanks to my parents who raised me with love and supported me with all the sacrifices in all my pursuits. Most of all for my loving, caring, supportive and encouraging wife, Xuan Zhou whose faithful support during the pursuit of this Ph.D. is so appreciated. Thank you.

Annan Dong

NJIT

May 2019

LIST OF ABBREVIATIONS

ADMM:	Alternating Direction Method of Multipliers
BAC:	Binary Asymmetric Channel
BSS:	Blind Source Separation
DL:	Dictionary Learning
DOA:	Direction of Arrival
EM:	Expectation Maximization
EVM:	Error Vector Magnitude
FDD:	Frequency Division Duplexing
HMM:	Hidden Markov Model
ICA:	Independent Component Analysis
IoT:	Internet of Things
LASSO:	Least Absolute Shrinkage and Selection Operator
MAP:	Maximum a Posteriori
MDU:	Multiple Dictionary Update
MIMO:	Multiple Input Multiple Output
ML:	Maximum Likelihood
MOD:	Method of Optimal Directions
NB-IoT:	Narrow Band IoT
OMP:	Orthogonal Matching Pursuit
PCA:	Principal Component Analysis
PSF:	Per-source Stochastic Filtering
SL-ADMM:	ADMM-based Smooth LASSO
SL-SEQ:	Sequential Smooth LASSO
SNR:	Signal to Noise Ratio
SVD:	Singular Value Decomposition

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Blind Source Separation

Blind source separation (BSS) refers to the separation of a set of source signals from a set of mixed signals, without resorting to any a priori information about the source signals or the mixing process [1]. BSS exploits only the information carried by the received signals themselves, hence the term *blind*. BSS has numerous applications in speech recognition [2, 3], image extraction [4, 5], and surveillance [6, 7]. Different metrics are used to evaluate the performance of BSS methods depending on the applications. For example, signal-to-interference ratio is used in [2] for speech separation, and a performance index is introduced in [4] for image feature extraction. Based on these metrics, many approaches have been proposed to solve BSS problems, such as independent component analysis (ICA) [8], principal component analysis (PCA) [9], and singular value decomposition (SVD) [10].

This dissertation addresses BSS in wireless networks. We are specifically interested in the set-up with a fusion center, in which the fusion center observes a number of radio sources via noisy sensor measurements over unknown channels. The system may model an Internet-of-Things (IoT) system, such as LoRa, Sigfox, or Narrow Band-IoT (NB-IoT) [11, 12].

In wireless networks involving multiple terminals operating over flat fading channels, the signals received at a terminal are linear mixtures of the signals emitted by the transmitting terminals. The need for BSS arises in non-collaborative applications in which the signals and the channels through which they are received at a terminal are both unknown.

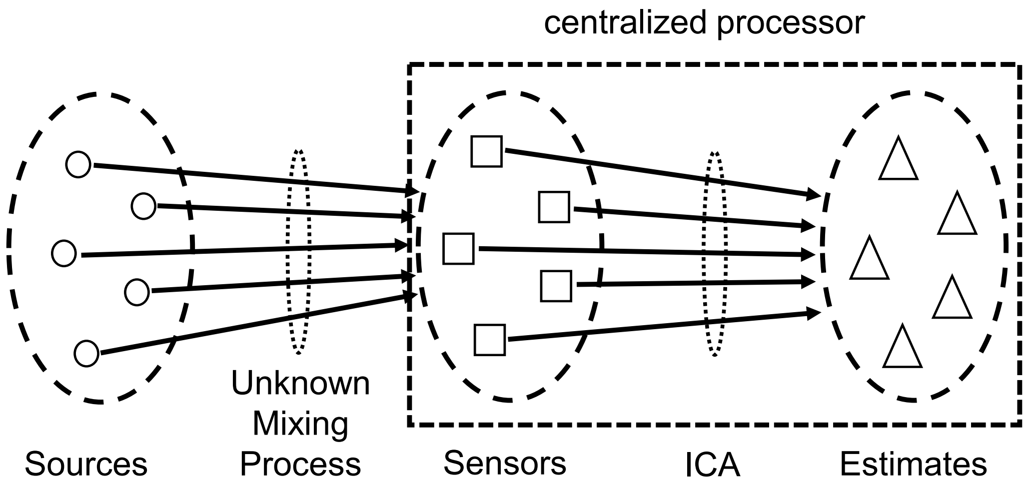

Figure 1.1 Centralized Processing ICA model.

1.2 Independent Component Analysis

ICA is a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals. ICA defines a generative model for the observed data, which is typically given as a large database of samples. In the model, the data variables are assumed to be linear or nonlinear mixtures of some unknown latent variables, and the mixing process is also unknown. The latent variables are assumed nongaussian and mutually independent; thus, they are called the independent components of the observed data [13]. These independent components, also called sources or factors, can be found by ICA for solving BSS problems.

Depending on the location of the processors that separate the observed data, two processing models are presented in ICA, namely centralized processing [14] and decentralized processing [15]. The set-up of centralized processing ICA is illustrated in Figure 1.1, in which all the processors are located in the same place and can exchange information during the source separation. Decentralized processing ICA has been a popular topic in recent years. Compared to centralized processing ICA, each processor in decentralized processing ICA estimates sources signals locally and

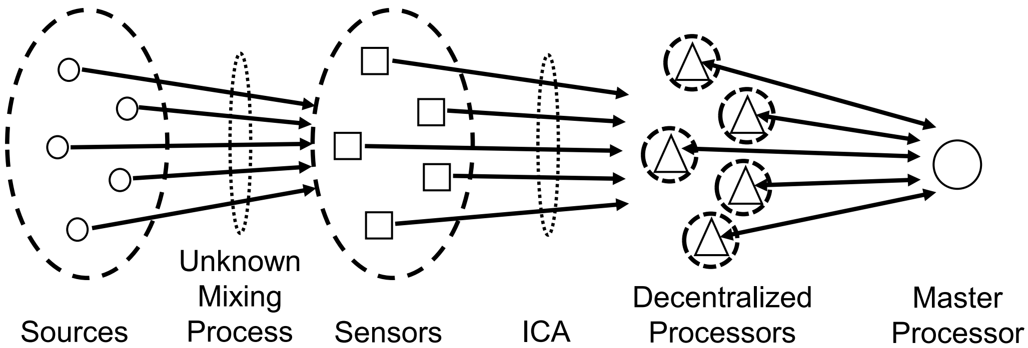

Figure 1.2 Decentralized Processing ICA model.

exchanges only limited information with a master processor, which is demonstrated in Figure 1.2. This feature allows the sensors to be deployed in a distributed pattern as well as to have a low computational cost.

In wireless networks field, ICA has been widely applied to solve BSS problems [16, 17, 18], since it yields a useful decomposition with only scaling, and permutation ambiguities [19]. To achieve signal separation, ICA relies on the statistical independence and on the non-Gaussian distribution of the components of the mixture. Key assumptions made in the implementation of the various forms of ICA are that the underlying mixing process has the same number of inputs and outputs, and that all sources are active throughout the observation interval. These assumptions are limiting and not suitable for the applications under study in this work, as discussed next.

1.3 Intermittent Sources

With the aim of capturing key aspects of IoT systems, this dissertation focuses on practical wireless scenarios in which the number of latent sources is generally larger than the number of sensors, but the sources are active intermittently with bursty transmission patterns. The cumulative time a source is active is a small fraction of the overall observation time, and the sources' on-off patterns vary slowly. In order to capture these properties, the sources are modeled as hidden Markov processes with known or unknown parameters. Source memory will be seen to be instrumental in enabling source separation. The wireless network operates over slow flat-fading channels; sensors communicate with a fusion center over ideal channels; and all nodes are time-synchronized to the same clock by the fusion center.

The general BSS problem with more sources than sensors can be formulated as an underdetermined linear system $\mathbf{Ax} = \mathbf{y}$ in the absence of noise. The columns of the matrix $\mathbf{A}$ serve as a basis for expressing the observations $\mathbf{y}$. The set of basis signals that form the matrix $\mathbf{A}$ is called a dictionary. Underdetermined linear systems of equations of the form $\mathbf{Ax} = \mathbf{y}$ have infinitely many solutions when the matrix $\mathbf{A}$ is full rank. Regularization may introduce additional conditions on the solutions, for example favoring smaller values of $\mathbf{x}$, leading to unique solutions of the underdetermined linear system. Sparse representations for which the solution $\mathbf{x}$ is unique have been the subject of intensive research resulting in a large body of literature [20, 21, 22]. Pre-defined dictionaries, such as based on Fourier transforms, are convenient and computationally fast, but in the BSS problem, both the dictionary $\mathbf{A}$ and the signals $\mathbf{x}$ are unknown.

1.4 Sparse Dictionary Learning

Sparse representation problems for which the dictionary is unknown require *dictionary learning* (DL) in addition to signal recovery [23, 24]. Many DL methods [25, 26] are second-order iterative *batch* procedures, accessing the whole training set at each iteration in order to minimize a cost function under some constraints [27]. They have been successfully utilized for both reconstruction and discriminative tasks [28]. The advantage of DL is to enable a system to learn a dictionary adaptively from a set of observations rather than assume a prescribed rigid dictionary. It has been experimentally shown that these adaptive dictionaries outperform the non-adaptive ones in many signal processing applications [20, 29].

Among other problems, DL methods have been applied to joint direction of arrival (DOA) estimation and array calibration [30], linear transceiver design [31], cloud K-singular value decomposition (K-SVD) for big, distributed data [32], and channel representation for frequency-division duplexing (FDD) massive MIMO system [33]. In these examples, the DL algorithms solve BSS problems in which the number of sources is larger than the number of sensors, but the methods are agnostic to time variability properties of sources with memory. In [34] and [35], the authors set up a hidden Markov model (HMM) to solve a BSS problem. However, in [34] the memory of the sources is not accounted for, while in [35] a simplified model is assumed whereby only one source can appear or disappear at any given time.

1.5 Contributions

In this dissertation, we propose a two-stage algorithm for solving the BSS problem for sources with memory modeled by an HMM and observed over slow flat-fading channels. The proposed algorithm comprises a DL stage and a Per-source Stochastic Filtering (PSF) stage. The DL stage of the proposed algorithm exploits knowledge about the source sparsity and memory to aid with the source separation. The effect of source memory is introduced by a penalty term that discourages solutions with short-duration transmissions by means of a smooth LASSO algorithm. The input to the DL stage are observations from the sensors. The output from the DL stage are channel estimates, source signal estimates and source states (active or inactive). The source state estimates produced by the DL algorithm may be viewed as the output of a binary asymmetric channel in which some of the states are "flipped" with respect to the true states. The error probabilities associated with the flipped states are referred to as flipping probabilities. The PSF stage consists of a forward-backward step along with an Expectation Maximization (EM) step that estimate the unknown HMM transition probabilities and flipping probabilities.

The main contributions of this dissertation are summarized as follows:

- A two-stage architecture is introduced for solving the problem of blind source estimation of HMM sources over slow flat-fading channel. The sources feature intermittent activity, and the number of latent source may be larger than the number of sensors;
- A smooth DL algorithm is proposed that is capable of exploiting source memory to support channel estimation, signal estimation and source detection. Two simplified versions of the smooth DL algorithms are introduced to reduce the computational complexity;
- An PSF algorithm is introduced that is capable of operating in the absence of a priori information about the HMM parameters and the state estimation flipping probabilities. This algorithm integrates an forward-backward step with an EM step.

1.6 How To Read This Dissertation

The rest of the dissertation is organized as follows. The system model and two source models with memory and intermittent activities are presented in Chapter 2. Background on existing DL algorithms that do not utilize source memory is provided in Chapter 3. In Chapter 4, we propose a two-stage DL-based algorithm to solve the BSS problem for sources with memory in wireless networks. The PSF is described in detail in Chapter 5 for the two cases with known and unknown source parameters. Finally in Chapter 6, simulation numerical results are shown to support that our proposed algorithm can separate sources and recover source signals with higher accuracy than existing DL algorithms.

CHAPTER 2

SYSTEM MODEL

Consider a system that includes M receiving antennas, or radio sensors, and N sources, as illustrated in Figure 2.1. The number of sources N is generally larger than the number of receive antennas M. All sensors are connected to a fusion center, which may be implemented in a cloud processor, via backhaul links. Equivalently, the fusion center has access to N receive antennas. Models with a fusion center reflect the architecture of IoT networks, such as LoRa, Sigfox, and NB-IoT [11, 12]. In our model, each source transmits intermittently, and hence is generally active only for a subset of the T symbol periods over which the sensors collect data.

Assuming that all nodes are time-synchronous, the discrete-time signal received by the M sensors over T symbol periods is given in matrix form as

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{Z}, \tag{2.1}$$

where $\mathbf{Y} = [\mathbf{y}(1), \ldots, \mathbf{y}(T)]$ is an $M \times T$ matrix collecting as columns the $M \times 1$ received signals $\mathbf{y}(t)$ across all T symbols $t = 1, \ldots, T$; $\mathbf{X} = [\mathbf{x}(1), \ldots, \mathbf{x}(T)]$ is an

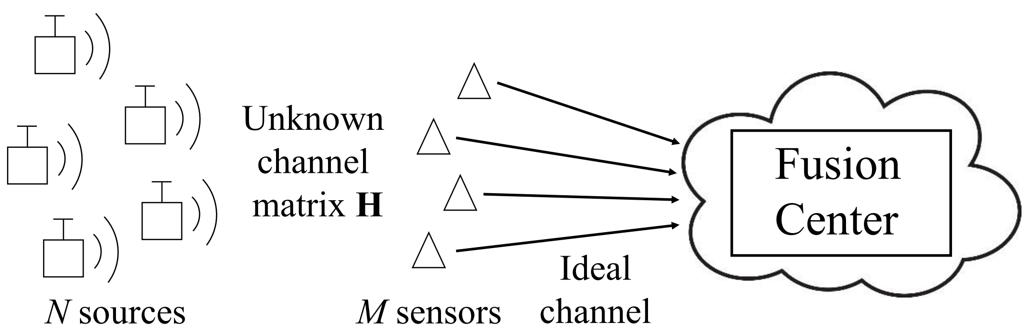

Figure 2.1 Wireless Sources transmit sporadically over a flat fading channel. The fusion center in the “cloud” wishes to estimate the activity and transmitted signals based on the signals received by the distributed radio sensors.

$N \times T$ matrix that gathers the $N \times 1$ signals $\mathbf{x}(t)$ transmitted from all N sources over time; $\mathbf{Z} = [\mathbf{z}(1), \ldots, \mathbf{z}(T)]$ contains independent zero-mean complex Gaussian noise entries with variance σ^2; and $\mathbf{H}$ is the $M \times N$ complex fading channel matrix. The channel matrix $\mathbf{H}$ is assumed to be constant for T symbol periods. According to [36], at any time sample, when the signal is k sparse, meaning there are k sources active at that time sample, the minimum number of sensors that the system requires to recover the data satisfies

$$m \geq Ck\log(N/k), \tag{2.2}$$

where C is a constant independent of m, N, and k. Hence, in our system, we assume $M \geq m$ across all time samples.

2.1 Source Model

Given the intermittent nature of the traffic pattern of the sources, the t-th column vector $\mathbf{x}(t)$ that collects the M symbols transmitted by the sources at time t, is generally sparse. In other words, only a subset of the entries of $\mathbf{x}(t)$ is non-zero. The signals $\mathbf{y}(t)$, for $t = 1, \ldots, T$, are collected at the fusion center. Based on the received signals $\mathbf{Y}$, the goal of the fusion center is to detect the sources' activity and to recover the signals $\mathbf{x}(t)$, for $t = 1, \ldots, T$, or equivalently the matrix $\mathbf{X}$, in the absence of information about the channel matrix $\mathbf{H}$.

For each source n, we define the activation pattern as a binary sequence $s_n(t)$, where the binary state $s_n(t)$ indicates whether a source is active or not. Specifically, when the state of source n is $s_n(t) = 1$, then the source is active, while it is inactive when $s_n(t) = 0$. The binary state is assumed to be described by the following models.

1. *Intermittent and smooth deterministic model*: Each source is active for a small fraction of time, and its on-off patterns tend to have few switches between on

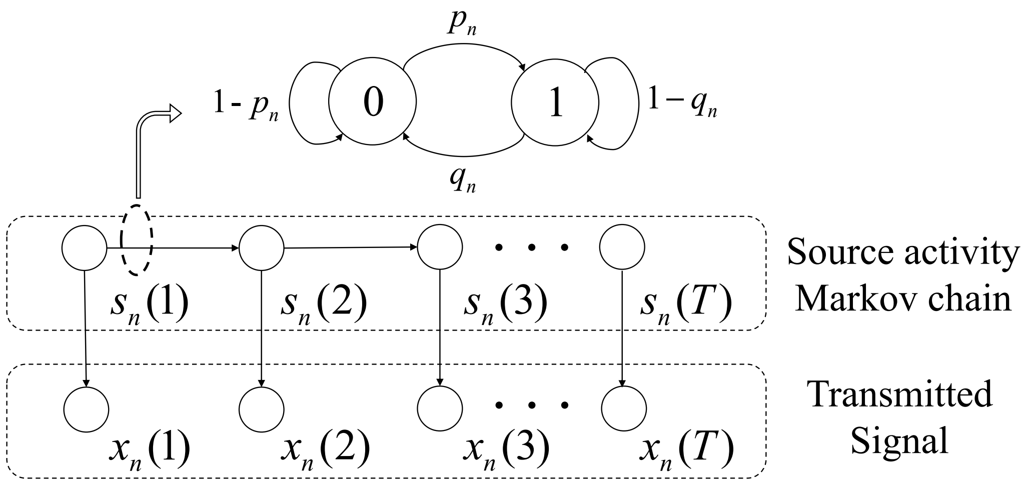

Figure 2.2 Hidden Markov Model (HMM) for a source n.

and off states. As a result, the sequence $s_n(t)$ is "smooth", i.e., it has a small number of transitions between on and off states;

2. *Probabilistic hidden Markov model*: Probabilistic hidden Markov model: As illustrated in Figure 2.2, the activity $s_n(t)$ of each source n follows a two-state Markov chain. The transition probabilities of the two-state Markov chain are defined as $p_n = \Pr(s_n(t) = 1|s_{n-1}(t) = 0)$ and $q_n = \Pr(s_n(t) = 0|s_{n-1}(t) = 1)$. We will consider both the cases in which the probabilities p_n and q_n are known and unknown.

When a source n is active, i.e., when $s_n(t) = 1$, it transmits an independent sample $x_n(t) \sim f_n(t)$ with a given distribution $f_n(t)$, e.g., Gaussian or binary. Instead, when the source n is inactive, i.e., $s_n(t) = 0$, it does not transmit and we have $x_n(t) = 0$.

2.2 Synchronization

A remark is in order concerning synchronization requirements. The described model (2.1) applies even in the absence of time synchronization among the sources, since the vector $\mathbf{x}(t)$ can model a generic sample at discrete time t of the transmitted signals.

However, if time synchronization is assumed, and the sources transmit digitally modulated signals, the vector $\mathbf{x}(t)$ can be assumed to contain the constellation points transmitted at the t-th symbol period. We note that time and frequency synchronization in sensor networks is a topic of great interest, and it has been investigated in blind scenarios with no pilot symbols as well [37, 38]. However, the literature about BSS often assumes that sensors and sources are synchronized in order to focus on the technical challenges of BSS [30, 31, 32, 33].

2.3 Summary

A system model containing a fusion center and two source models with memory and intermittent activities are introduced in this chapter. We emphasize that the two source models are not mutually exclusive and can be assumed to hold simultaneously. In particular, we will propose to leverage the deterministic model in order to improve the performance of source separation and the stochastic model to refine the estimates obtained by sources separation.

CHAPTER 3

PRELIMINARIES: DICTIONARY LEARNING

The DL method proposed in this work leverages prior information about the memory of the sources. We start by reviewing DL methods that do not exploit such information. These methods use only the fact that the signal $\mathbf{x}(t)$ is sparse at any time t. Prior information about the memory of each source $x_n(t)$, to be considered in the next chapter, includes smoothness properties or statistical models.

Assuming only information about the sparseness of $\mathbf{x}(t)$ at each time t, a standard approach is to utilize the channel matrix $\mathbf{H}$ as a *dictionary* to be learned to recover $\mathbf{X}$. DL techniques approximate the solution of the maximum likelihood (ML) problem if the fusion center acts as a master clock synchronizing all other nodes.

$$\underset{\mathbf{H},\mathbf{X}\in\mathcal{X}}{\text{minimize}} \; ||\mathbf{Y}-\mathbf{H}\mathbf{X}||^2, \tag{3.1}$$

where $\mathcal{X}$ is the set of matrices with sparse columns, that is, with columns containing a limited number of non-zero entries. This problem is not convex with respect to the pair ($\mathbf{H}$, $\mathbf{X}$). DL methods use an iterative procedure, whereby the signal $\mathbf{X} \in \mathcal{X}$ and the channel $\mathbf{H}$ are optimized alternately [39]. The block diagram of dictionary learning algorithm is illustrated in Figure 3.1. In the following, we first discuss solutions for the optimization over the signal $\mathbf{X}$ for a given channel matrix $\mathbf{H}$, and then over the channel $\mathbf{H}$ for a given signal matrix $\mathbf{X}$.

DL methods are subject to inherent permutation and sign ambiguities [40]. The scaling ambiguity can be instead resolved if one imposes that the channel matrix columns are normalized [41, 42, 43, 44].

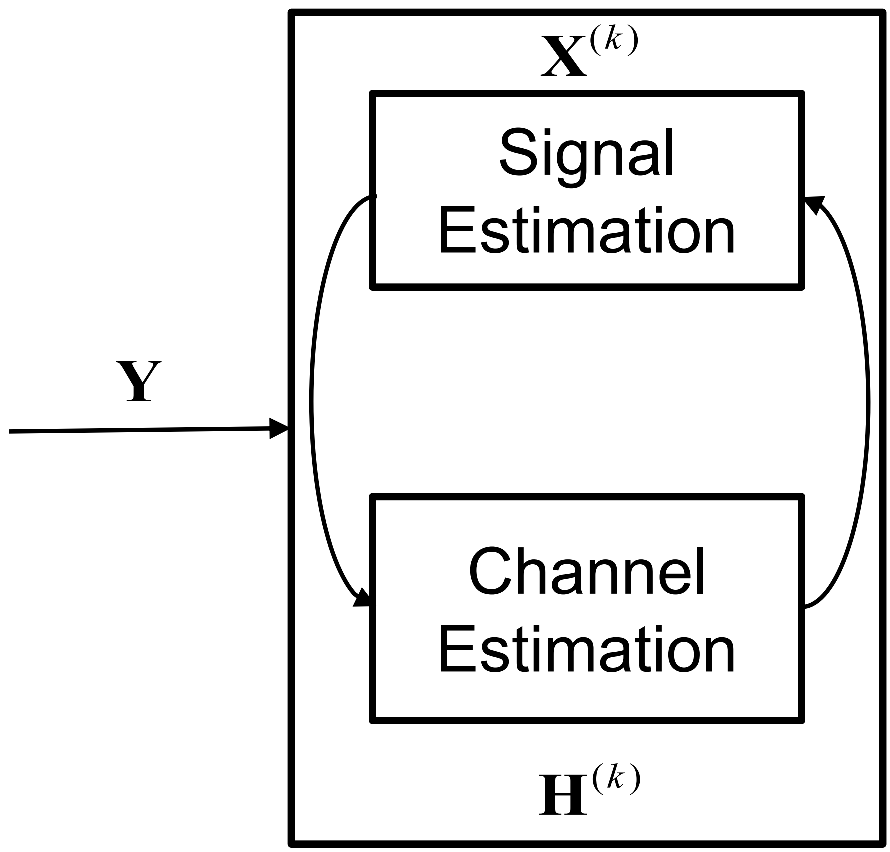

Figure 3.1 Block diagram of dictionary learning algorithm.

3.1 Signal Estimation

For any fixed iterate $\mathbf{H}^{(k)}$ at the k-th iteration, from (3.1), the problem of estimating the signal $\mathbf{X}$ reduces to

$$\mathbf{X}^{(k+1)} = \underset{\mathbf{X}\in\mathcal{X}}{\operatorname{argmin}} ||\mathbf{Y} - \mathbf{H}^{(k)}\mathbf{X}||^2. \tag{3.2}$$

Standard sparse optimization estimators, such as orthogonal matching pursuit (OMP) [20, §3.1.2] can be used to address problem (3.2). Alternatively, one can use the LASSO algorithm [45] to solve the convex problem

$$\underset{\mathbf{x}(t)}{\text{minimize}} \; ||\mathbf{y}(t) - \mathbf{H}^{(k)}\mathbf{x}(t)||_2 + \lambda||\mathbf{x}(t)||_1, \qquad t = 1, \ldots, T, \tag{3.3}$$

separately for each t, where the weight λ is a parameter to be determined as a function of the sparsity of vector $\mathbf{x}(t)$.

3.2 Channel Estimation

At the k-th iteration, for a fixed iterate $\mathbf{X}^{(k+1)}$, the channel estimation step can obtain the next channel iterate $\mathbf{H}^{(k+1)}$ by using different algorithms, such as the Method of Optimal Directions (MOD) [41], the Multiple Dictionary Update (MDU) [42], the Sequential Generalization of K-means (SGK) [43], or their enhanced versions proposed in [44]. Here we summarize the enhanced MDU approach, which was shown in [44] to provide the best performance via simulation results.

The MDU approach estimates the channel matrix $\mathbf{H}$ for a given $\mathbf{X}^{(k+1)}$ by following an iterative approach. To elaborate, denote $\mathcal{S}(\mathbf{x})$ the set of indices of the non-zero elements in vector $\mathbf{x}$. Also, index (j, k) the j-th iteration of the MDU algorithm within the k-th step of the DL alternate optimization scheme. At the (j, k) iteration, for $j = 1, 2, \ldots,$ MDU computes

$$\mathbf{H}^{(j,k)} = \mathbf{Y}\mathbf{X}^{(j,k)T}(\mathbf{X}^{(j,k)}\mathbf{X}^{(j,k)T})^{-1}, \tag{3.4}$$

and

$$\mathbf{x}^{(j+1,k)}(t) = \mathbf{D}^{(k)}(\mathbf{H}^{(j,k)T}\mathbf{H}^{(j,k)})^{-1}\mathbf{H}^{(j,k)T}\mathbf{y}(t), \tag{3.5}$$

where for all t, $\mathbf{D}^{(k)}$ is a diagonal matrix with elements having indices in $\mathcal{S}(\mathbf{x}^{(k+1)}(t))$ equal to 1 and zero otherwise. The iteration is initialized with $\mathbf{X}^{(1,k)} = \mathbf{X}^{(k+1)}$. Therefore, for a fixed sparsity pattern $\mathcal{S}(\mathbf{x}^{(k+1)}(t))$, MDU alternately estimates channel and signals.

The enhancement proposed in [44] substitutes at iteration k the received signal $\mathbf{Y}$ in (3.4) and (3.5) with $\mathbf{Y}^{(k)} = \mathbf{Y} + \mathbf{H}^{(k)}\mathbf{X}^{(k)} - \mathbf{H}^{(k)}\mathbf{X}^{(k+1)}$.

$$\underset{\mathbf{H},\mathbf{X}\in\mathcal{X}}{\text{minimize}} \; ||\mathbf{Y}^{(k)} - \mathbf{H}\mathbf{X}||^2, \tag{3.6}$$

[44] claims this substitute makes (3.6) convex with respect to the pair $(\mathbf{H}, \mathbf{X})$. In this sense, the substitute can improve the performance of all three algorithms mentioned above.

A final remark is that the reviewed DL techniques are designated for flat-fading channels. Multipath scenarios would require convolutional dictionary learning [46], and would be the topic of a future study.

3.3 Numerical Comparison

In order to show the choice of enhanced MDU approach, we replicate the numerical result in [44] for all six DL algorithms. For signal estimation step, OMP is applied to solve a known fixed number of sparsity in the signal activities. For channel estimation step, all six algorithms are applied to compare the performance: MOD, MDU, SGK and their enhanced version, namely CvxMOD, Cvx MDU and CvxSGK. In the simulations, the numerical results are obtained for $N = 50$ sources, $M = 20$ sensors, signal to noise ratio (SNR) per source per sample 30 dB, $T = 1000$ time samples. Throughout this chapter, the number of iterations is fixed at $K = 45$.

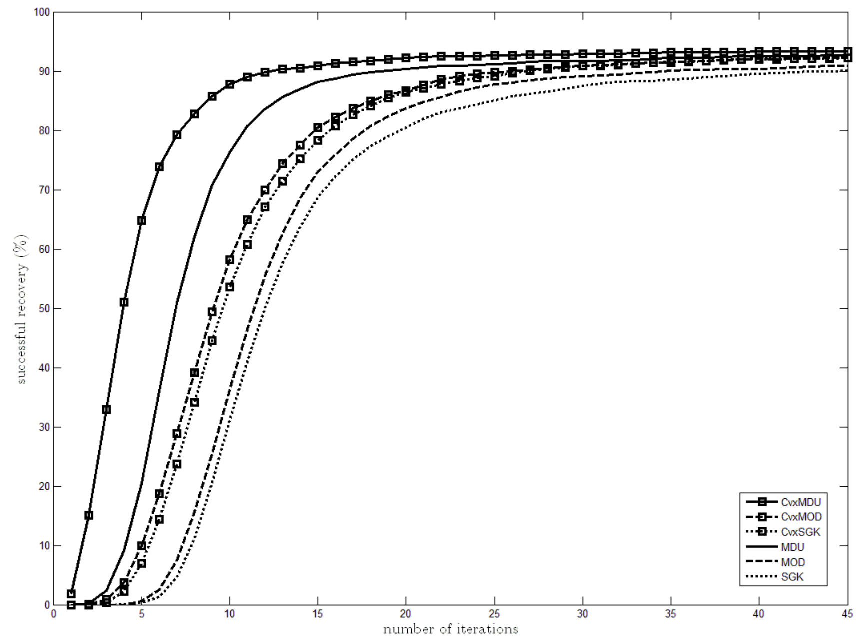

Figure 3.2 Successful recovery versus number of iterations for the DL algorithms introduced in this chapter and their convex forms (50 potential sources, 20 sensors, 1000 time samples, 3 active sources per time sample, 30 dB SNR).

Figure 3.2 shows the performance comparison for all six algorithms mentioned above. the first thing we observe here is the enhanced MDU, labeled CvxMDU in the figure, algorithm has the best performance in term of successful signal recovery percentage and it also has the best convergence rate. Since the channel estimation step is not the focus of this dissertation, we decide to apply enhanced MDU algorithm in the channel estimation step in our proposed algorithm.

3.4 Summary

The DL algorithms mentioned above focus on improving the performance in the channel estimation step, while our proposed algorithm concentrates on increasing the performance in the signal estimation step when the source memory is present, as discussed next. The main difference relies on their algorithms pay more attention on recovering the dictionary (the channel), while our algorithm want to reconstruct the transmitted signals to decipher the message contained in the signals.

CHAPTER 4

SMOOTH DICTIONARY LEARNING

In this chapter, we propose an improved DL-based source separation algorithm that exploits knowledge about the memory of the sources $n = 1, \dots, N$. In particular, we introduce a modified DL scheme that accounts for the intermittent and smooth deterministic model discussed in Chapter 2.

To account for the assumption of smoothness, the signal estimation step described in Section 3.1 is modified by substituting the LASSO algorithm (3.3) with a smooth LASSO algorithm [47]. Smooth LASSO adds a penalty term for on/off switches within the transmitted signals. This penalty reflects prior knowledge that the transmitted signals do not switch on/off an excessive number of times. Accordingly, at each iteration k, given the channel iterate $\mathbf{H}^{(k)}$, instead of using (3.3), the proposed method obtains the updated estimate $\mathbf{X}^{(k)}$ of the channel matrix by solving the problem

$$\underset{\mathbf{X}}{\text{minimize}}\ ||\mathbf{Y} - \mathbf{H}^{(k)}\mathbf{X}||_2^2 + \lambda \sum_{t=1}^{T} ||\mathbf{x}(t)||_1 + \mu \sum_{t=2}^{T} ||\mathbf{x}(t) - \mathbf{x}(t-1)||^2, \tag{4.1}$$

where μ is a weight parameter that is set depending the level of smoothness expected in the transmitted signals. Larger values of μ indicate fewer expected changes in the transmitted signal. Solving (4.1) directly has a complexity in the order of $O(MN^2T^2)$ [48]. To reduce the computational complexity, we introduce two approximations of (4.1), namely Sequential Smooth LASSO (SL-SEQ) and Alternating Direction Method of Multipliers-based Smooth LASSO (SL-ADMM). SL-SEQ solves (4.1) column-wise sequentially, and can be found in [49] as an approximation for (4.1); while SL-ADMM is an ADMM-based [50, 51] iterative approach we propose to tackle (4.1) with lower computational complexity.

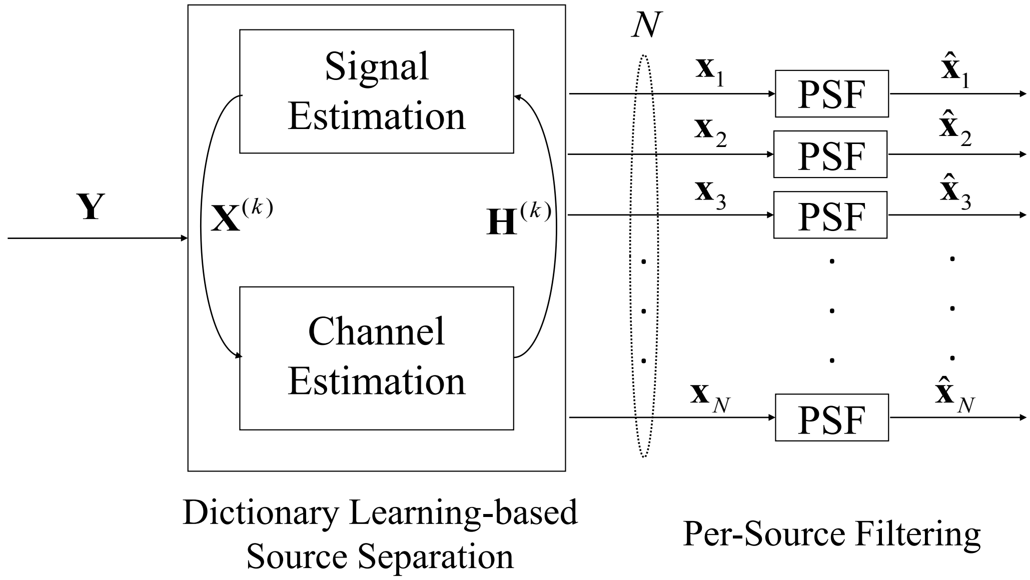

Figure 4.1 Block diagram of the proposed algorithm with Per-Source Filtering under the hidden Markov model for the sources' activity.

4.1 Sequential Smooth LASSO (SL-SEQ)

SL-SEQ solves the problem

$$\underset{\mathbf{x}(t)}{\text{minimize}} \ ||\mathbf{y}(t) - \mathbf{H}^{(k)}\mathbf{x}(t)||_2^2 + \lambda||\mathbf{x}(t)||_1 + \mu||\mathbf{x}(t) - \mathbf{x}(t-1)||^2, \tag{4.2}$$

sequentially for $t = 1, \ldots, T$, where $\mathbf{x}(t-1)$ is the solution obtained at the previous step [49]. With this scheme, the quadratic penalty term imposes on the current solution a proximity constraint with respect to the previous one. SL-SEQ has a complexity that in the order of $O(MN^2T)$ as opposed to $O(MN^2T^2)$ since it solves the T problems in (4.2) sequentially.

4.2 ADMM-Based Smooth LASSO (SL-ADMM)

We now propose an approximation to Smooth LASSO with ADMM approach [50, 51]. The novelty of SL-ADMM is to solve BSS problems with lower computational complexity than smooth LASSO for sources that have memory. To this end, we introduce a copy $\mathbf{x}'(t)$ of $\mathbf{x}(t)$ and rewrite problem (4.1) as

$$\begin{aligned} &\underset{\mathbf{X},\mathbf{X}'}{\text{minimize}}\ ||\mathbf{Y}-\mathbf{H}^{(k)}\mathbf{X}||_2+\lambda\sum_{t=1}^{T}||\mathbf{x}(t)||_1+\mu\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2, \\ &\qquad\qquad \text{subject to } \mathbf{x}'(t)=\mathbf{x}(t), \qquad t=1,\ldots,T-1, \end{aligned} \tag{4.3}$$

where we have defined the matrix $\mathbf{X}'=[\mathbf{x}'(1),\ldots,\mathbf{x}'(T-1)]$. Define the functions $l_1(\mathbf{x}(1))=||\mathbf{y}(1)-\mathbf{H}^{(k)}\mathbf{x}(1)||_2+\lambda||\mathbf{x}(1)||_1$ and $l_t(\mathbf{x}(t),\mathbf{x}'(t-1))=||\mathbf{y}(t)-\mathbf{H}^{(k)}\mathbf{x}(t)||_2+\lambda||\mathbf{x}(t)||_1+\mu||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2$, for $t\geq 2$. Then, the augmented Lagrangian [52] for problem (4.3) can be written as:

$$\begin{aligned} \mathcal{L}(\mathbf{x},\mathbf{x}',\boldsymbol{\alpha}) &= l_1(\mathbf{x}(1))+\sum_{t=2}^{T}l_t(\mathbf{x}(t),\mathbf{x}'(t-1)) \\ &+\sum_{t=1}^{T-1}\boldsymbol{\alpha}^T(t)(\mathbf{x}(t)-\mathbf{x}'(t))+\rho\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2 \\ &= l_1(\mathbf{x}(1))+\boldsymbol{\alpha}^T(1)\mathbf{x}(1) \\ &+\sum_{t=2}^{T-1}(l_t(\mathbf{x}(t),\mathbf{x}'(t-1))+\boldsymbol{\alpha}^T(t)\mathbf{x}(t)-\boldsymbol{\alpha}^T(t-1)\mathbf{x}'(t-1)) \\ &+l(\mathbf{x}(T),\mathbf{x}'(T-1))-\boldsymbol{\alpha}^T(T-1)\mathbf{x}'(T-1) \\ &+\rho\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2, \end{aligned} \tag{4.4}$$

where $\boldsymbol{\alpha}(t)$ is the $N\times 1$ Lagrange multipliers for the constraints in (4.3) and $\rho\geq 0$ is a parameter. The proposed method tackles problem (4.3) via a primal-dual subgradient method that is summarized for each iteration i of ADMM in Table 4.1.

We emphasize that SL-SEQ has a computational complexity in the order of $O(MN^2T)$, and SL-ADMM has a complexity in the order of $O(MN^2TI)$, where I is the number of iterations in the ADMM approach. In general, I is chosen to be a small value compared to T. It is seen that the computational complexity is greatly reduced with SL-SEQ and SL-ADMM compared to solving (4.1) directly.

Table 4.1 SL-ADMM Algorithm

- For the current iterates $\boldsymbol{\alpha}^{(i)}(t)$, solve in parallel the T problems:

· $\underset{\mathbf{x}(1)}{\text{minimize}}\ l_1(\mathbf{x}(1)) + \boldsymbol{\alpha}^{(i)T}(1)\mathbf{x}(1),$ (4.5a)

· $\underset{\mathbf{x}(t),\mathbf{x}'(t-1)}{\text{minimize}}\ l_t(\mathbf{x}(t), \mathbf{x}'(t-1)) + \boldsymbol{\alpha}^{(i)T}(t)\mathbf{x}(t) - \boldsymbol{\alpha}^{(i)T}(t-1)\mathbf{x}'(t-1) + \rho||\mathbf{x}(t) - \mathbf{x}'(t-1)||^2,$ for $t = 2, \ldots, T-1,$ (4.5b)

· $\underset{\mathbf{x}(T),\mathbf{x}'(T-1)}{\text{minimize}}\ l_T(\mathbf{x}(T), \mathbf{x}'(T-1)) - \boldsymbol{\alpha}^{(i)T}(T-1)\mathbf{x}'(T-1) + \rho||\mathbf{x}(T) - \mathbf{x}'(T-1)||^2,$ (4.5c)

· obtaining the new iterates $\mathbf{x}^{(i)}(t)$ for $t = 1, \ldots, T$ and $\mathbf{x}'^{(i)}(t)$ for $t = 2, \ldots, T$;

- Update the Lagrange multipliers as

$$\boldsymbol{\alpha}^{(i+1)}(t) \leftarrow \boldsymbol{\alpha}^{(i)}(t) + \rho(\mathbf{x}^{(i)}(t) - \mathbf{x}'^{(i)}(t-1)), \tag{4.6}$$

where $(\cdot)^T$ represents the transpose of a vector or a matrix.

CHAPTER 5

PER-SOURCE STOCHASTIC FILTERING

In this chapter, we propose a further improvement to the DL-based source separation schemes discussed thus far which as shown in Figure 4.1, consists of a per-source post-processing step. Specifically, once we have obtained an estimate $\tilde{x}_n(t), t = 1, \ldots, T$, of the signal of any source n based on one of the DL-based methods discussed above, a perform per-source stochastic filtering (PSF) is applied that leverages the HMM of the sources to remove some of the "noise" in the estimates $\tilde{x}_n(t)$. The "noise" flips the state of the signal from active to inactive and vice-versa. Filtering of the noise leads to a more accurate activity map of the signals.

PSF is based on the hidden Markov model of Figure 2.2. We first consider the case in which the transition probabilities (p_n, q_n) of the hidden Markov model are known, and then study the scenario in which this information is not available.

5.1 Known Model Parameters

Adopting a Bayesian formulation, a first approach would be to interpret the signal $\tilde{\mathbf{x}}_n(t)$ obtained from each source n via DL-based source separation as the output of a memoryless channel $p(\tilde{\mathbf{x}}_n|\mathbf{x}_n)$ whose input is the current transmitted signal $\mathbf{x}_n$. Composing this channel with the hidden Markov model of Figure 2.2, one could apply the forward-backward message passing algorithm on the resulting Bayesian network in order to compute the posterior probabilities $p(x_n(t)|\tilde{x}_n(1), \ldots, \tilde{x}_n(T))$ for all $t = 1, \ldots, T$ [53]. Here we do not follow this approach since it is generally unclear how to define the channel $p(\tilde{\mathbf{x}}_n|\mathbf{x}_n)$, and the resulting message passing algorithm typically requires computationally expensive integrations. In contrast, we take a more pragmatic low-complexity approach, which is described next.

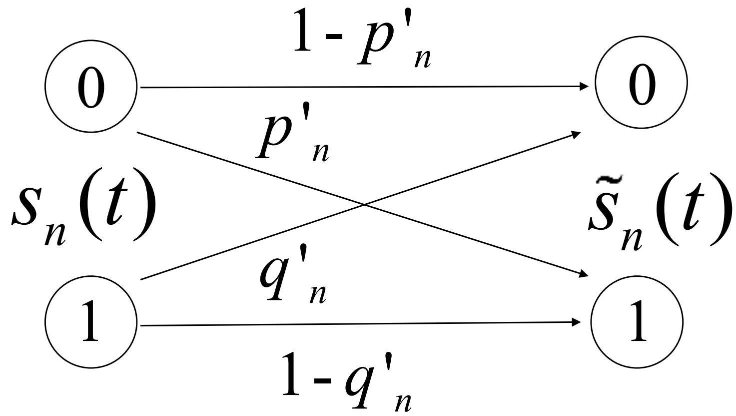

Figure 5.1 Binary Asymmetric Channel (BAC) describing the relationship between the true state $s_n(t)$ and the observed state $\tilde{s}_n(t)$ obtained from the DL-based source separation step that is assumed by PSF.

First, we perform a binary quantization of the available estimates $\tilde{x}_n(t), t = 1, \ldots, T$, by thresholding the absolute value of $\tilde{x}_n(t)$. This provides an estimate $\tilde{s}_n(t)$ of the state $s_n(t)$, since small absolute values of $\tilde{\mathbf{x}}_n(t)$ suggest the absence of signal for source n at time t. As a result, we obtain the binary sequence $\tilde{s}_n(t), t = 1, \ldots, T$, where $\tilde{s}_n(t) = 1$ if $|\tilde{x}_n(t)| > \gamma$ for a threshold γ, and $\tilde{s}_n(t) = 0$ otherwise. The selection of the threshold γ will allow us to obtain different points on the trade-off between the probability of false alarm and missed detection, as discussed in Chapter 6.

To formulate the filtering problem, we model the observations $\tilde{s}_n(t), t = 1, \ldots, T$, as being received at the output of a binary asymmetric channel (BAC) with flip probabilities p'_n and q'_n, as illustrated in Figure 5.1. The BAC models the errors made by the DL-based source separation algorithm that produced $\tilde{x}_n(t), t = 1, \ldots, T$, in detecting the state variables $s_n(t)$. Parameters p'_n and q'_n can be used as additional degrees of freedom in exploring the trade-off points between the probability of false alarm and missed detection. As an alternative, we will discuss in the next subsection how to estimate them from the data.

Given the observations $\tilde{s}_n(t), t = 1, \ldots, T$, and the parameters p_n, q_n, p'_n and q'_n, we compute the posterior distribution $p(s_n(t)|\tilde{s}_n(1), \ldots, \tilde{s}_n(T))$ by using the forward-backward algorithm [54]. Accordingly, the posterior probability can be written as

$$\Pr(s_n(t) = 1|\tilde{s}_n(1), \ldots, \tilde{s}_n(T)) = \alpha_t \beta_t, \tag{5.1}$$

where the probability $\alpha_t = \Pr(s_n(t) = 1|\tilde{s}_n(1), \ldots, \tilde{s}_n(t))$ is obtained as a result of the forward pass, while the probability $\beta_t = \Pr(\tilde{s}_n(t+1), \ldots, \tilde{s}_n(T)|s_n(t) = 1)$ is obtained from the backward pass as explained in [54, §17.4.3]. We then estimate $s_n(t)$ using the maximum a posteriori (MAP) approach, i.e., we set $\tilde{s}_n(t) = 1$ if the inequality $p(s_n(t)|\tilde{s}_n(1), \ldots, \tilde{s}_n(T)) > 0.5$ holds, and $\tilde{s}_n(t) = 0$ otherwise. To recover an estimate $\hat{x}_n(t)$ of the transmitted signal $x_n(t), t = 1, \ldots, T$, we finally null all entries of $\tilde{x}_n(t)$ corresponding to states $s_n(t)$ that are estimated to be zero, i.e., $\hat{s}_n(t) = 0$, while leaving unaltered the values of $\tilde{x}_n(t)$ for times t at which $s_n(t)$ is estimated to be $\hat{s}_n(t) = 1$. This can be summarized as

$$\hat{x}_n(t) = \begin{cases} \tilde{x}_n(t) & \text{if } \hat{s}_n(t) = 1, \\ 0 & \text{if } \hat{s}_n(t) = 0. \end{cases} \tag{5.2}$$

5.2 Unknown Model Parameters

We now study the case in which the parameters p_n and q_n of the HMM are unknown, and need to be estimated. We also jointly estimate the parameters p'_n and q'_n in the BAC of Figure 5.1 that is used to derive the forward-backward filtering algorithm. To this end, we apply the Expectation-Maximization (EM) algorithm [54], as described next.

Given the estimated states $\tilde{s}_n(t)$ of each source n obtained from the source separation and quantization steps, we would like to estimate the state sequence $s_n(t),\ t = 1, \ldots, T$, the probabilities p_n and q_n in the transition matrix of the Markov chain, as well as the probabilities p'_n and q'_n in the BAC. The EM algorithm can be detailed as follows.

- **Initialization:** Initialize $p_n^{(0)}$, $q_n^{(0)}$, $p_n'^{(0)}$ and $q_n'^{(0)}$.

For each iteration $\nu = 1, 2, \ldots$:

- **E Step:** Given the probabilities $p_n = p_n^{(\nu-1)}$, $q_n = q_n^{(\nu-1)}$, $p'_n = p_n'^{(\nu-1)}$ and $q'_n = q_n'^{(\nu-1)}$, apply the forward-backward algorithm to calculate the probabilities α_t and β_t and hence the posterior probabilities (5.1), for $n = 1, \ldots, N$.

- **M Step:** Update the probability parameters by averaging the sufficient statistics with respect to the posterior distributions identified during the E step [55]. This leads to the updates

$$p_n^{(\nu+1)} \leftarrow \frac{\sum_{t=2}^{T} \Pr(s_n(t) = 1, s_n(t-1) = 0 | \tilde{\mathbf{s}}_n)}{\sum_{t=2}^{T} \Pr(s_n(t-1) = 0 | \tilde{\mathbf{s}}_n)}, \tag{5.3}$$

$$q_n^{(\nu+1)} \leftarrow \frac{\sum_{t=2}^{T} \Pr(s_n(t) = 0, s_n(t-1) = 1 | \tilde{\mathbf{s}}_n)}{\sum_{t=2}^{T} \Pr(s_n(t-1) = 1 | \tilde{\mathbf{s}}_n)}, \tag{5.4}$$

where $\tilde{\mathbf{s}}_n = [\tilde{s}_n(1), \ldots, \tilde{s}_n(T))]$, $\Pr(s_n(t-1) = 0|\tilde{\mathbf{s}}_n) = 1 - \Pr(s_n(t-1) = 1|\tilde{\mathbf{s}}_n)$ is obtained from (5.1), and the posterior joint probabilities in (5.3) and (5.4) are computed as detailed below; and

$$p_n'^{(\nu+1)} \leftarrow \frac{\sum_{t=1}^{T} \mathbb{1}(\tilde{s}_n(t) = 1) \Pr(s_n(t) = 0 | \tilde{\mathbf{s}}_n)}{\sum_{t=1}^{T} \Pr(s_n(t) = 0 | \tilde{\mathbf{s}}_n)}, \tag{5.5}$$

$$q_n'^{(\nu+1)} \leftarrow \frac{\sum_{t=1}^{T} \mathbb{1}(\tilde{s}_n(t) = 0) \Pr(s_n(t) = 1 | \tilde{\mathbf{s}}_n)}{\sum_{t=1}^{T} \Pr(s_n(t) = 1 | \tilde{\mathbf{s}}_n)}, \tag{5.6}$$

where $\mathbb{1}(a)$ is the indicator function, i.e. $\mathbb{1}(a) = 1$ if a is true and 0 otherwise.

- **Stopping Criterion:** The iteration stops when the parameters converge, i.e. $|p_n^{(\nu)} - p_n^{(\nu-1)}| < \epsilon$, where ϵ is a small value.

The posterior joint probabilities in (5.3) and (5.4) are computed as [56]

$$\begin{aligned} &\Pr(s_n(t) = i, s_n(t-1) = j | \tilde{s}_n(1), \ldots, \tilde{s}_n(T)) = \\ &\alpha_{t-1,j+1} \Phi_{j+1,i+1}^{(\nu)} \Pr(\tilde{s}_n(t) | s_n(t) = i) \beta_{t,i+1}, \qquad i, j = 0, 1, \end{aligned} \tag{5.7}$$

where $\alpha_{t,j}$ is the j-th element of vector $\boldsymbol{\alpha}_t = [1-\alpha_t, \alpha_t]^T$; $\beta_{t,i}$ is the i-th element of vector $\boldsymbol{\beta}_t = [1-\beta_t, \beta_t]^T$; $\Phi_{j,i}^{(\nu)}$ is the (j,i)-th element of the transition matrix $\boldsymbol{\Phi}_n^{(\nu)}$ in the ν-th iteration; and matrix $\boldsymbol{\Phi}_n^{(\nu)}$ is defined as

$$\boldsymbol{\Phi}_n^{(\nu)} = \begin{bmatrix} 1-p_n^{(\nu)} & p_n^{(\nu)} \\ q_n^{(\nu)} & 1-q_n^{(\nu)} \end{bmatrix}. \tag{5.8}$$

CHAPTER 6

NUMERICAL RESULTS

In this chapter, we present numerical results to obtain insights into the performance of different DL-based source separation schemes and on the advantage of PSF. We consider the following source separation schemes, all implemented with and without the post-processing PSF step: OMP, LASSO, SL-SEQ and SL-ADMM. As performance criteria, we adopt the probability of false alarm P_{fa} and probability of detection P_d. The detection probability P_d is the ratio between the number of correctly detected active sources and the total number of active sources over the T symbols, and the false alarm probability P_{fa} is the ratio between the number of incorrectly detected active sources and the total number of inactive sources over the T time samples. We also consider the performance of source estimation in terms of error vector magnitude (EVM) to synchronized digital transmission (see Chapter 2).

Unless stated otherwise, the numerical results were obtained for $N = 30$ sources, $M = 20$ sensors, signal to noise ratio (SNR) per source per sample 30 dB, $T = 1000$ time samples. We assume an HMM for the state $s_n(t)$, which is defined by transition probabilities $p_n = 0.0022$, $q_n = 0.02$, for all N sources, so that an average $Np_n/(p_n+q_n) = 3$ sources are active at each time sample t, and the average duration of transmission is $1/q_n = 50$ time samples. Also, unless stated otherwise, the algorithms with PSF are implemented with fixed values $p'_n = 0.02$ and $q'_n = 0.27$. We optimize numerically over the multipliers λ and μ in order to satisfy given constants on the probability of detection P_d or the probability of false alarm P_{fa}. For PSF with unknown model parameters, we initialize the transition probabilities of the Markov model as $p_n = 0.5$, $q_n = 0.5$, and the flip probabilities of the BAC as $p'_n = 0.1$, $q'_n = 0.2$. Throughout this chapter, the number of iterations for SL-ADMM was fixed at $K = 30$; the Lagrange multipliers $\boldsymbol{\alpha}$ in (4.5) are initialized as the all-one vector; ρ is set as 0.1 and γ is fixed at 0.5.

6.1 Tuning Multipliers λ and μ

We start by discussing the choice of the multipliers λ and μ in optimization problems (4.2) and (4.3) for SL-SEQ and SL-ADMM. These parameters penalize the number of non-zero elements in the solution vector and the number of state changes of each source, respectively. The choice of these two multipliers is scenario-dependent, and a discussion on their optimal selection can be found in [57]. In our extensive simulations, we found that the rule-of-thumb choices $\lambda = 1/\text{SNR}$ and $\mu = 0.1/q_n$, where q_n is the transition probability from an active to an inactive state (see Figure 2.2), works well in practice. The selection $\lambda = 1/\text{SNR}$ reflects a decrease in the relevance of the sparsity prior as the quality of the observations increases [57]. In contrast, the multiplier $\mu \propto 1/q_n$ increases as the average transmission duration $1/q_n$ increases, implying fewer state changes.

As an example, Figures 6.1 and 6.2 show the probability of detection P_d for a fixed probability of false alarm $P_{fa} = 0.1$ versus λ and μ, respectively. From Figure 6.1, we observe that, when varying the SNR level, the optimal selection of λ changes according to the mentioned inverse proportionality rule 1/SNR. In a similar manner, Figure 6.2 confirms the validity of our choice for multiplier μ.

6.2 Source Activity Detection

We then investigate the trade-off between the probability of detection P_d versus the probability of false alarm P_{fa} in Figure 6.3. The transmitted signal $x_n(t)$ is assumed here to be distributed as $x_n(t) \sim \mathcal{N}(0, 1)$ whenever $s_n(t) = 1$. We also assume known source statistics p_n and q_n and keep the performance from known channels as references. A first observation is that both SL-SEQ and SL-ADMM significantly outperform all other source separation schemes, with the latter obtaining

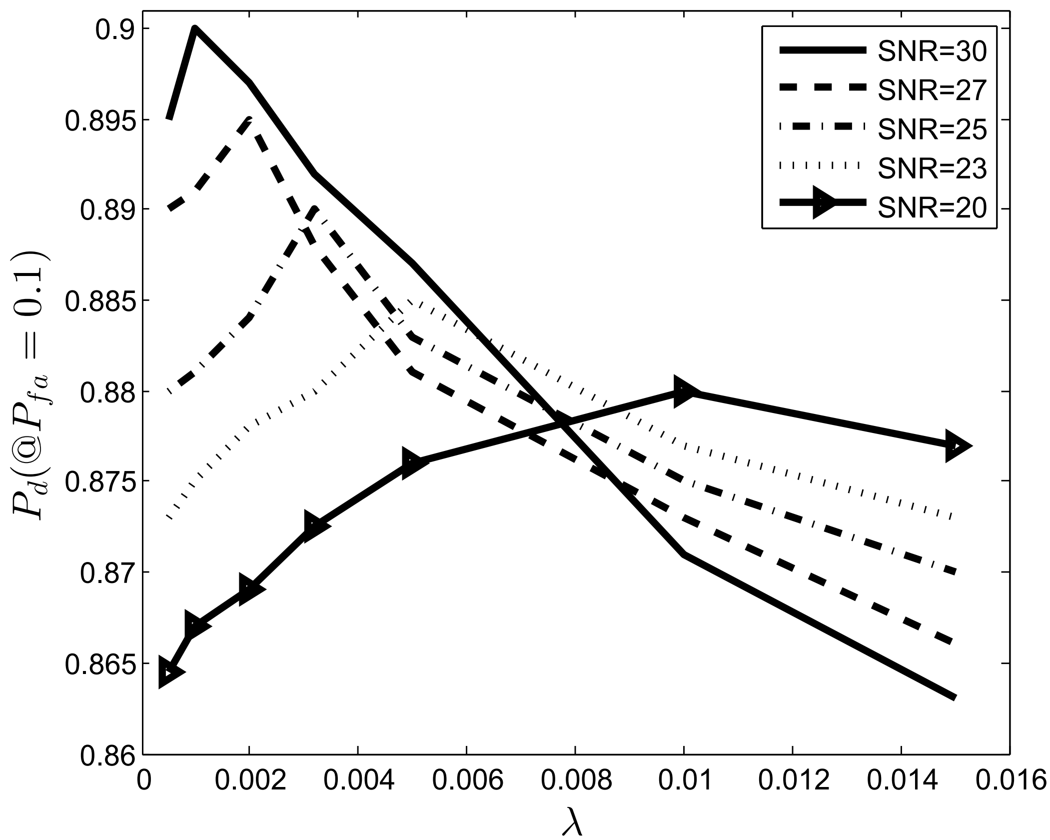

Figure 6.1 Probability of detection P_d when the probability of false alarm is $P_{fa} = 0.1$ versus λ for the SL-ADMM algorithm ($N = 30$, $M = 20$, $T = 1000$, $p_n = 0.0022$, $q_n = 0.02$).

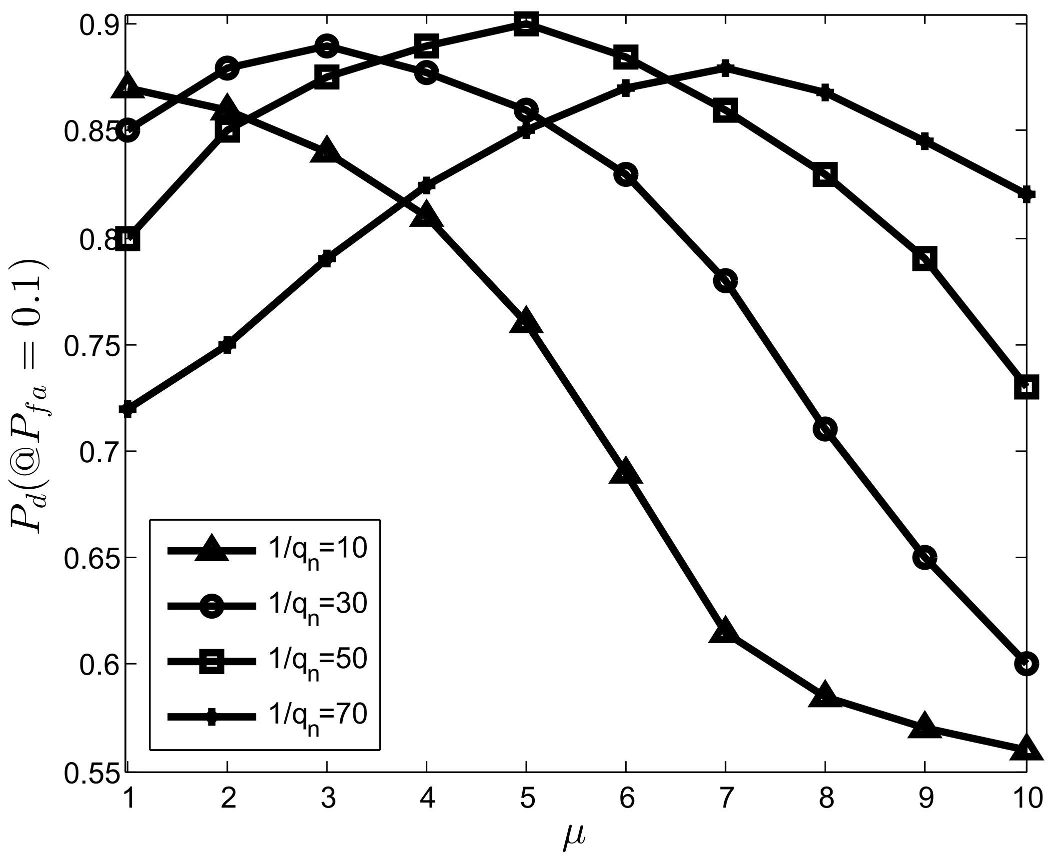

Figure 6.2 Probability of detection P_d when the probability of false alarm is $P_{fa} = 0.1$ versus μ for the SL-ADMM algorithm ($N = 30$, $M = 20$, $T = 1000$, SNR= 30 dB.

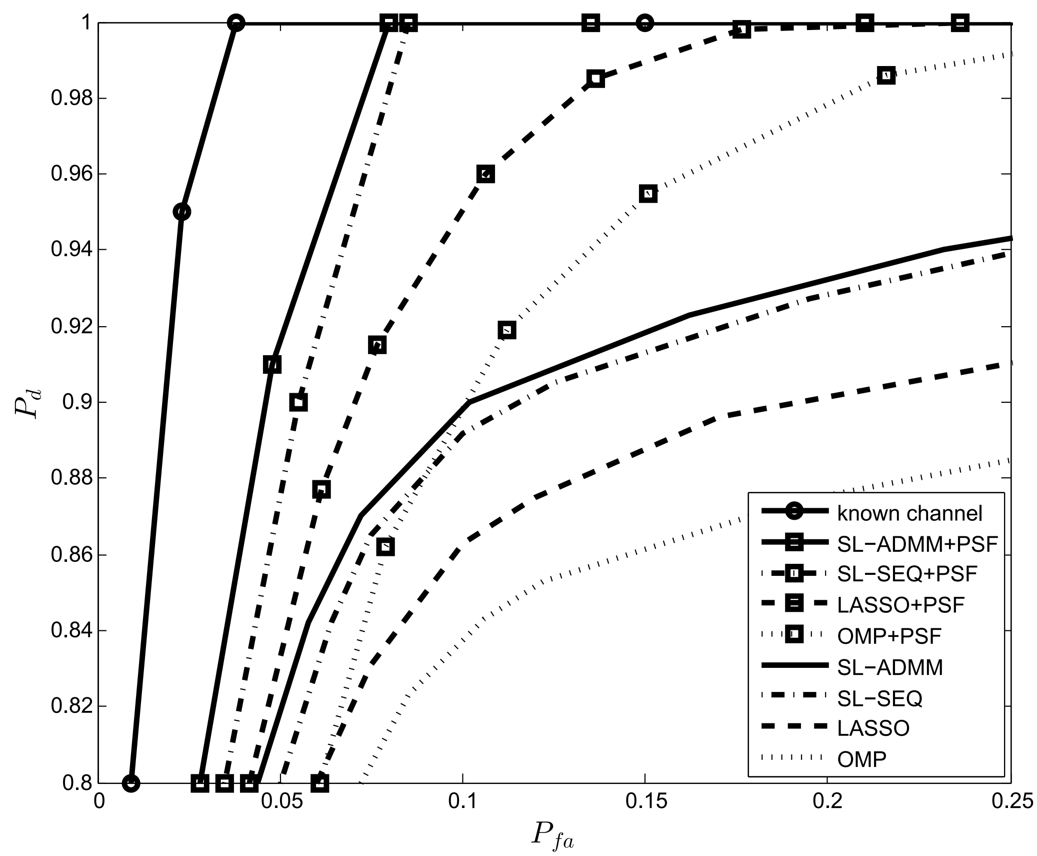

Figure 6.3 Probability of detection P_d versus probability of false alarm P_{fa} for the considered algorithms ($N = 30$, $M = 20$, $T = 1000$, SNR= 30 dB, $p_n = 0.0022$, $q_n = 0.02$).

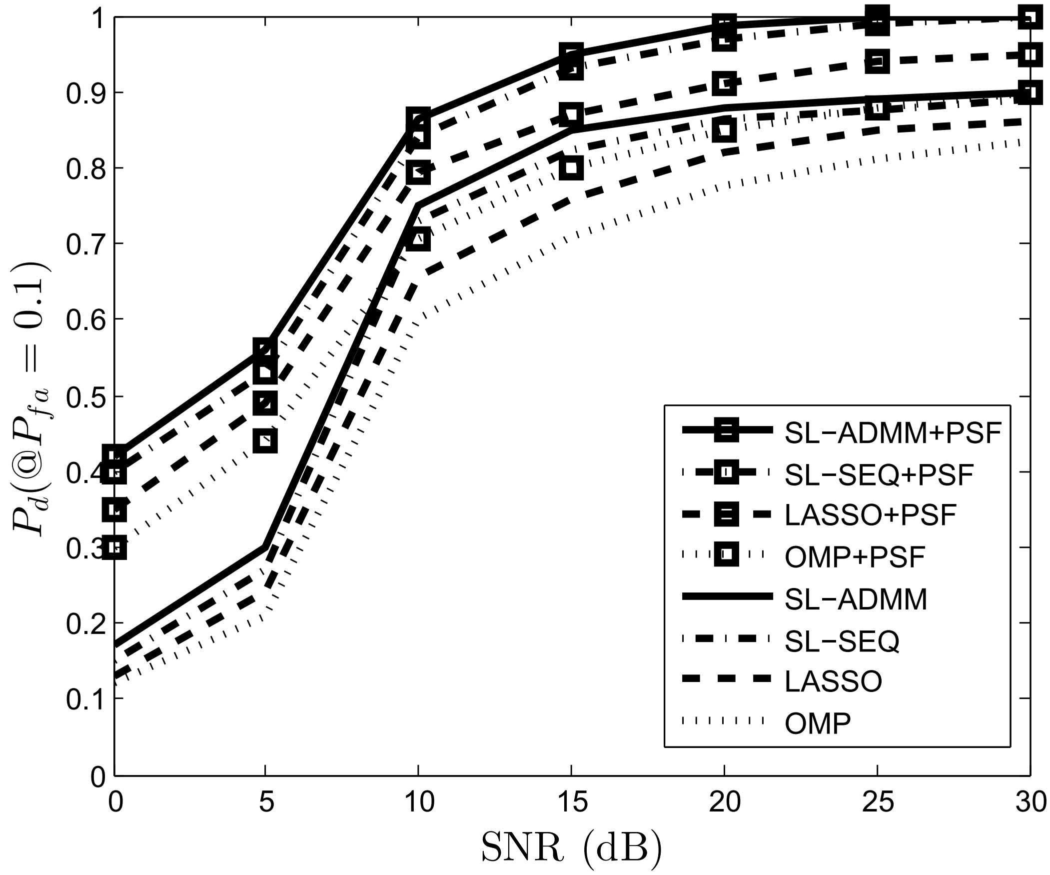

Figure 6.4 Probability of detection P_d when the probability of false alarm is $P_{fa} = 0.1$ versus SNR for the considered algorithms ($N = 30$, $M = 20$, $T = 1000$, $p_n = 0.0022$, $q_n = 0.02$).

some performance gain over the former. Furthermore, PSF provides a significant performance boost for all algorithms. For example, at $P_{fa} = 0.07$, the probability of detection with SL-ADMM is increased from 0.87 to 0.98.

We now explore the impact of different operating regimes on the probability of detection when the probability of false alarm is constrained to be smaller than 0.1. In particular, Figure 6.4 shows performance with respect to different SNR values, and Figure 6.5 investigates the impact of the sparsity level, which is defined as the ratio between the average number $Np_n/(p_n + q_n)$ of active sources and the total number N of sources. The former is modified by changing p_n. Both figures confirm the main conclusions obtained above in terms of the relative performance of the considered

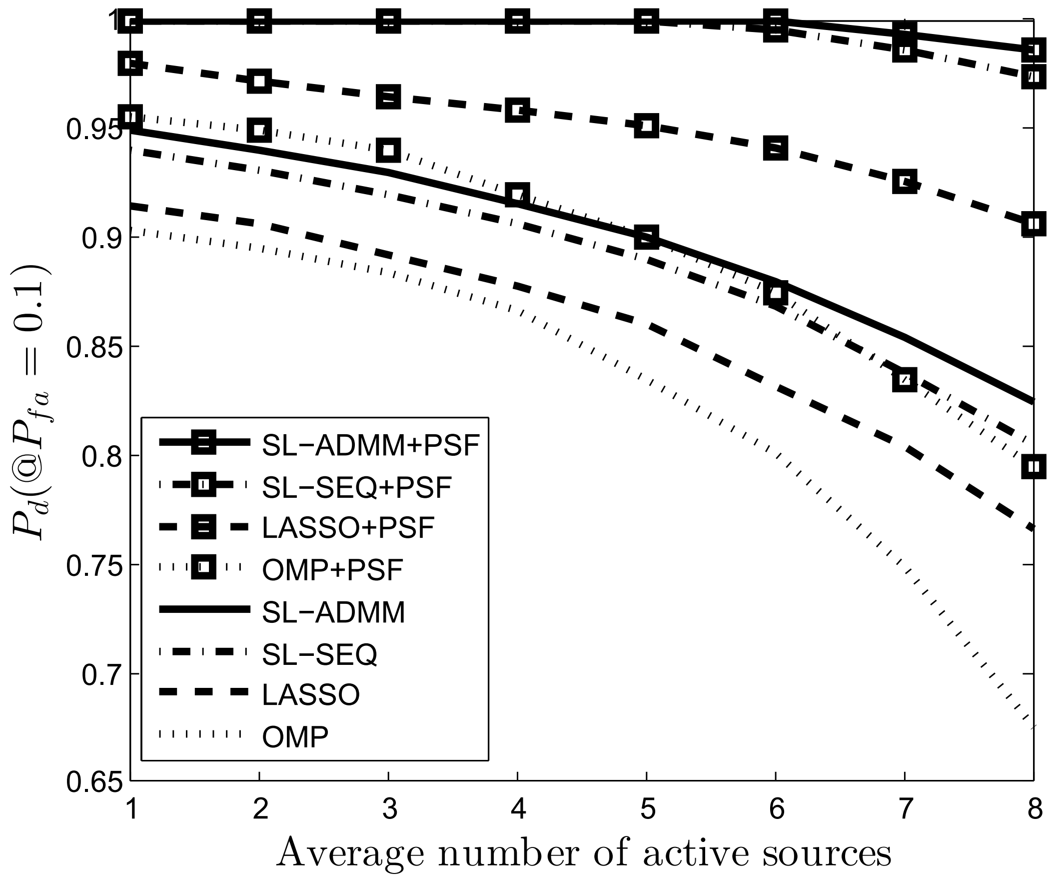

Figure 6.5 Probability of detection P_d when the probability of false alarm is $P_{fa} = 0.1$ versus average number of active sources ($N = 30$, $M = 20$, $T = 1000$, SNR= 30 dB, $q_n = 0.02$).

schemes. Moreover, Figure 6.4 suggests that the performance undergoes a threshold phenomenon with respect to the SNR, particularly if implemented without PSF. It is also seen that SL with PSF is able to obtain a vanishing probability of missed detection as the SNR increases, unlike the other schemes whose probability of detection reaches a ceiling lower than one. Finally, Figure 6.5 indicates that SL with PSF is robust to the sparsity level, while the other schemes are extremely sensitive to an increase in the average number of active sources.

Another parameter that we can consider is the average duration of transmission, which is the average length of each source being active $1/q_n$. Figure 6.6 investigates the impact of the average duration of transmission. The figure confirms the main

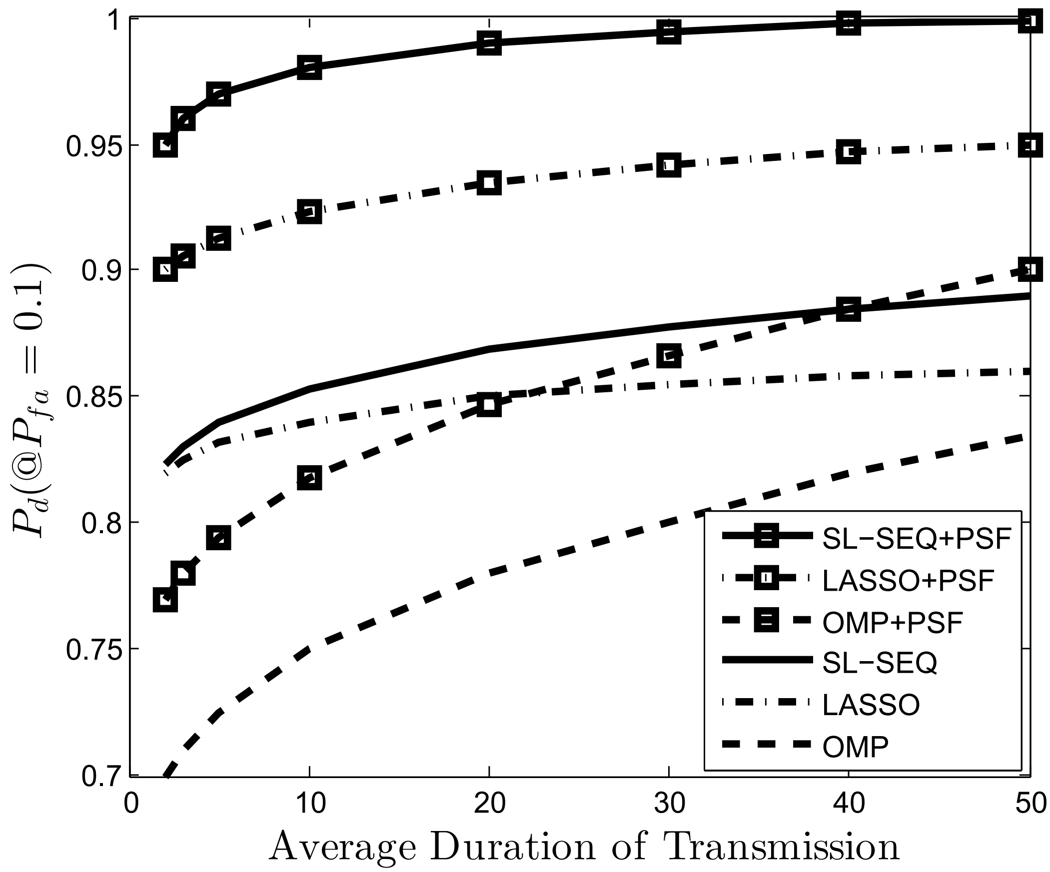

Figure 6.6 Probability of detection P_d when the probability of false alarm is $P_{fa} = 0.1$ versus average duration of transmission for the considered algorithms ($N = 30$, $M = 20$, $T = 1000$, SNR= 30dB.

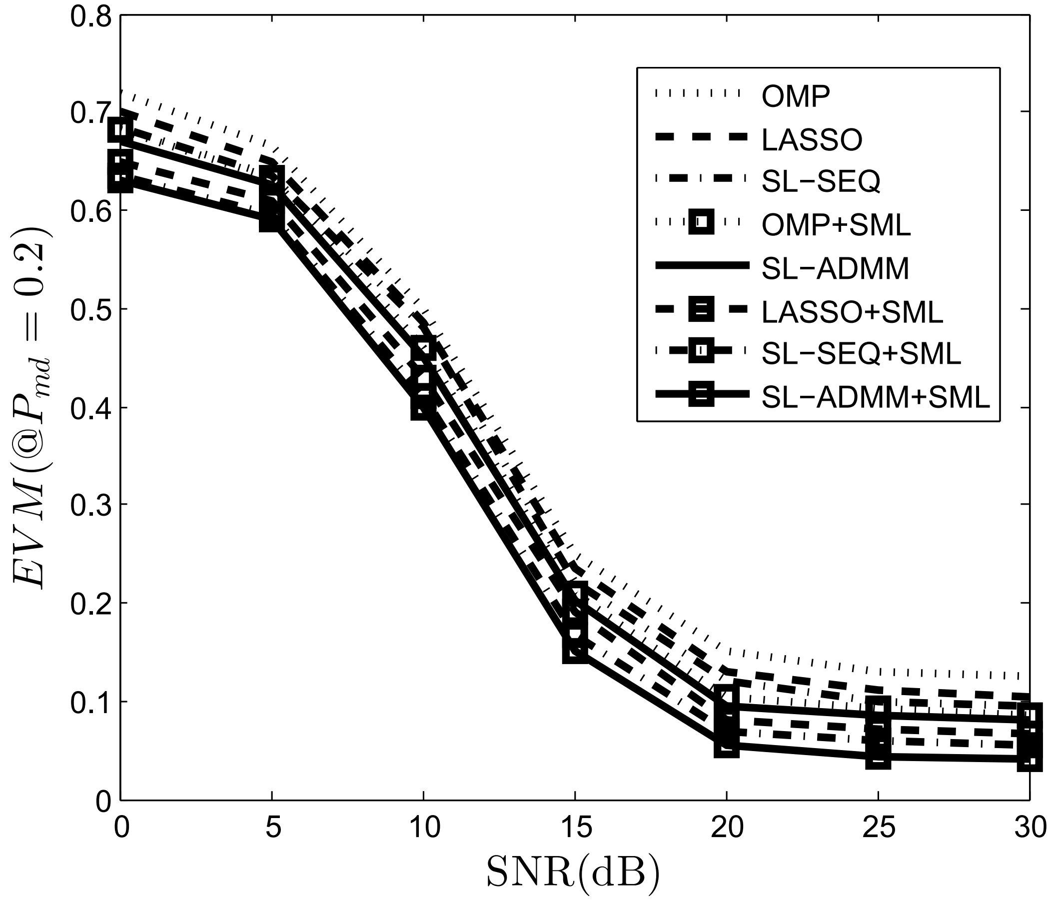

Figure 6.7 EVM versus SNR ($N = 30$, $M = 20$, $T = 1000$, $p_n = 0.0022$, $q_n = 0.02$).

conclusions obtained above in terms of the relative performance of the considered schemes.

6.3 Signal Estimation

As seen above, the SL schemes with PSF have the best performance in terms of source activity detection. Here, we study the performance in terms of the quality of signal estimation. To this end, we adopt the criterion of the EVM. EVM is defined as

$$\mathrm{EVM}(\%) = \sqrt{\frac{\sum_{n=1}^{N}\sum_{t\in\mathcal{D}_n}(x_n(t) - \tilde{x}_n(t))^2}{\sum_{n=1}^{N}\sum_{t\in\mathcal{D}_n} x_n^2(t)}} \times 100\%, \tag{6.1}$$

where $\mathcal{D}_n$ is the set of time samples in which user n is correctly detected as active. Here we assume that, when $s_n(t) = 1$, the transmitted signal is binary, that is $x_n(t) =1$ or -1 with equal probability.

Assuming again known sources' transition probabilities, Figure 6.7 shows the EVM for all the considered algorithms with respect to SNR where the probability of missed detection P_{md} is constrained to be smaller than 0.2. The behavior of the EVM is in line with the discussion above regarding source activity detection. In particular, we observe a 40% decrease at 30 dB SNR that are achievable with SL and PSF, as well as the threshold behavior as a function of the SNR.

6.4 Parameter Estimation

Here, we evaluate the performance loss incurred when the parameters p_n and q_n in the Markov model defining the sources' activities, are not known. Using the framework of Section 5.2, We jointly estimate the parameters p_n, q_n, p'_n and q'_n with the EM algorithm. Figure 6.8 investigates the performance of the estimation of the hidden parameters p'_n and q'_n in the BAC. From the figure, the estimate is seen to be close to the real value. The parameters p_n and q_n in the HMM have similar estimation accuracy as p'_n and q'_n. Figure 6.9 shows the probability of detection P_d versus the probability of false alarm P_{fa} under the same conditions as Figure 6.3. A first observation is that, even with unknown source parameters, PSF can provide a significant performance boost. Nevertheless, the gain is somewhat reduced as compared to the case with perfect knowledge. For instance, for SL-ADMM, when $P_{fa} = 0.07$, the probability of detection P_d is reduced from 0.98 to 0.89. The loss is generally increases when T is smaller.

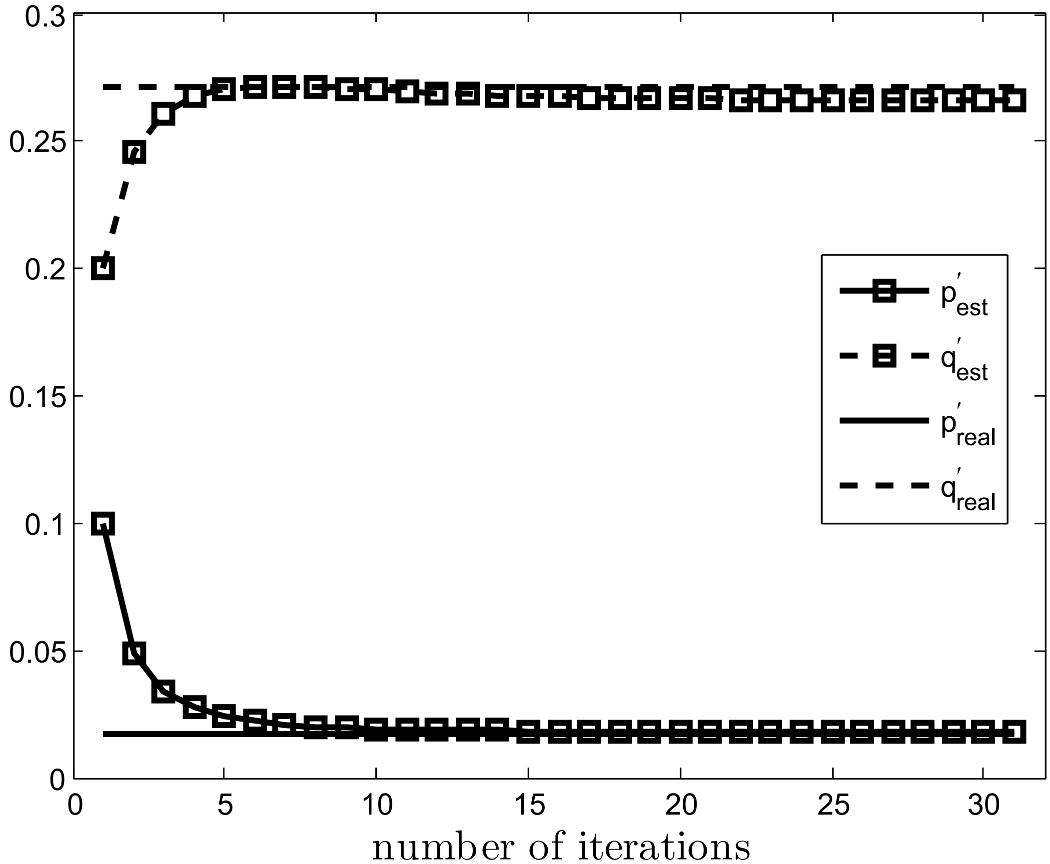

Figure 6.8 Estimation performance of p'_n and q'_n for BAC in the EM algorithm ($N = 30$, $M = 20$, $T = 1000$, SNR= 30 dB, $p_n = 0.0022$, $q_n = 0.02$).

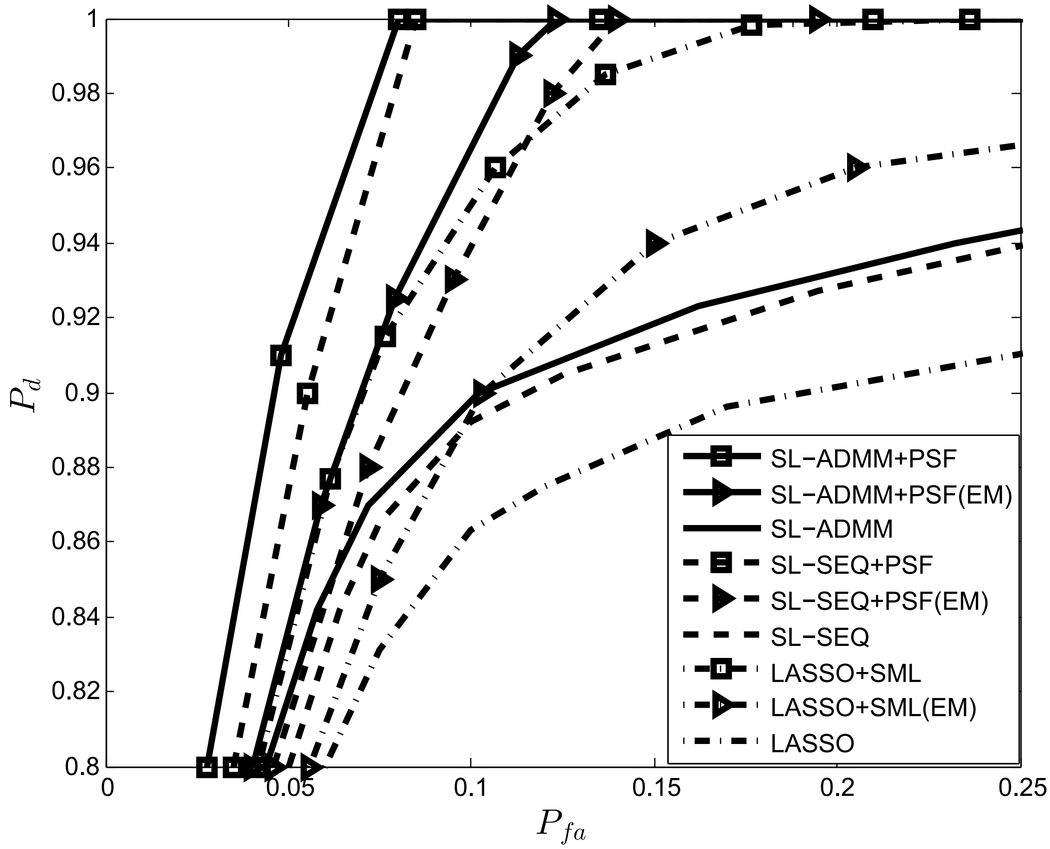

Figure 6.9 Probability of detection P_d versus probability of false alarm P_{fa} for the considered algorithms with EM algorithm ($N = 30$, $M = 20$, $T = 1000$, SNR= 30 dB, $p_n = 0.0022$, $q_n = 0.02$).

CHAPTER 7

CONCLUSIONS

In this dissertation, we have introduced a two-stage Dictionary Learning (DL)-based algorithm for solving the BSS problem as the presence of radio sources with memory that are observed over slow flat-fading wireless channels.

In Chapter 2, the wireless system model with a fusion center observing a number of radio sources via noisy sensor measurements over unknown flat fading channels is introduced. The sources' activity patterns can be described as "intermittent and smooth deterministic model" and "probabilistic hidden Markov model". These two models are not mutually exclusive and can be assumed to hold simultaneously. In Chapter 3, existing DL methods that do not exploit prior information about the memory of the sources are reviewed. These methods use only the fact that the transmitted signals are sparse at any given time.

In Chapter 4, the DL stage of the proposed algorithm is introduced with smooth LASSO algorithm. The DL stage exploits source memory information to aid with the source separation. The effect of source memory is accounted for by a penalty term that discourages short-duration transmissions. Two approximations of the smooth LASSO algorithm, namely namely Sequential Smooth LASSO and Alternating Direction Method of Multipliers-based Smooth LASSO are developed to reduce the computational complexity of the proposed algorithm.

In Chapter 5, the PSF stage of the proposed algorithm is introduced with forward-backward algorithm based on the hidden Markov model. The PSF stage utilizes source model information to learn about unknown source model parameters to further enhance source estimation. Both scenarios with known and unknown model parameters are discussed and an EM algorithm is also developed to obtain the unknown model parameters from the received signals.

In Chapter 6, Simulation numerical results are presented to obtain insights into the performance of different DL-based source separation schemes and on the advantage of PSF. Numerical results show that the proposed algorithm outperforms existing DL algorithms in terms of source activity detection as well as source signal estimation. As a representative numerical example, even with unknown source statistics, it is shown that with 30 potential sources, 20 sensors and average 3 sources active in each time sample, it is possible to increase the probability of detection from 0.9 without PSF to 0.96 with PSF at 0.1 probability of false alarm and 30 dB SNR. The proposed algorithm can also decrease the EVM from by 40% at 0.8 probability of detection at 30 dB SNR. In addition, the proposed algorithm is robust to the source sparsity level and the average duration of transmission time, while existing DL algorithms are extremely sensitive to an increase in the number of active sources and an shorter time of transmission.

APPENDIX A

ADMM-BASED SMOOTH LASSO

For SL-ADMM, we introduce a copy $\mathbf{x}'(t)$ of $\mathbf{x}(t)$ and rewrite problem Equation (4.1) as

$$\begin{aligned} \underset{\mathbf{X},\mathbf{X}'}{\text{minimize}}\; & ||\mathbf{Y}-\mathbf{H}^{(k)}\mathbf{X}||_2+\lambda\sum_{t=1}^{T}||\mathbf{x}(t)||_1+\mu\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2, \\ \text{subject to } & \mathbf{x}'(t)=\mathbf{x}(t), \qquad t=1,\dots,T-1, \end{aligned} \tag{A.1}$$

where we have defined the matrix $\mathbf{X}'=[\mathbf{x}'(1),\dots,\mathbf{x}'(T-1)]$. Define the functions $l_1(\mathbf{x}(1))=||\mathbf{y}(1)-\mathbf{H}^{(k)}\mathbf{x}(1)||_2+\lambda||\mathbf{x}(1)||_1$ and $l_t(\mathbf{x}(t),\mathbf{x}'(t-1))=||\mathbf{y}(t)-\mathbf{H}^{(k)}\mathbf{x}(t)||_2+\lambda||\mathbf{x}(t)||_1+\mu||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2$, for $t\geq 2$. Then, the augmented Lagrangian [52] for problem (4.3) can be written as:

$$\begin{aligned} \mathcal{L}(\mathbf{x},\mathbf{x}',\boldsymbol{\alpha}) &= l_1(\mathbf{x}(1))+\sum_{t=2}^{T}l_t(\mathbf{x}(t),\mathbf{x}'(t-1)) \\ &+\sum_{t=1}^{T-1}\boldsymbol{\alpha}^T(t)(\mathbf{x}(t)-\mathbf{x}'(t))+\rho\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2 \\ &= l_1(\mathbf{x}(1))+\boldsymbol{\alpha}^T(1)\mathbf{x}(1) \\ &+\sum_{t=2}^{T-1}(l_t(\mathbf{x}(t),\mathbf{x}'(t-1))+\boldsymbol{\alpha}^T(t)\mathbf{x}(t)-\boldsymbol{\alpha}^T(t-1)\mathbf{x}'(t-1)) \\ &+l(\mathbf{x}(T),\mathbf{x}'(T-1))-\boldsymbol{\alpha}^T(T-1)\mathbf{x}'(T-1) \\ &+\rho\sum_{t=2}^{T}||\mathbf{x}(t)-\mathbf{x}'(t-1)||^2, \end{aligned} \tag{A.2}$$

where $\boldsymbol{\alpha}(t)$ is the $N\times 1$ Lagrange multipliers for the constraints in Equation (4.3) and $\rho\geq 0$ is a parameter. The proposed method tackles problem Equation (4.3) via a primal-dual subgradient method that carries out the following steps at each iteration i of ADMM

- For the current iterates $\boldsymbol{\alpha}^{(i)}(t)$, solve in parallel the T problems:

  · $\underset{\mathbf{x}(1)}{\text{minimize}}\ l_1(\mathbf{x}(1)) + \boldsymbol{\alpha}^{(i)T}(1)\mathbf{x}(1),$ (A.3a)

  · $\underset{\mathbf{x}(t),\mathbf{x}'(t-1)}{\text{minimize}}\ l_t(\mathbf{x}(t), \mathbf{x}'(t-1)) + \boldsymbol{\alpha}^{(i)T}(t)\mathbf{x}(t) - \boldsymbol{\alpha}^{(i)T}(t-1)\mathbf{x}'(t-1) + \rho||\mathbf{x}(t) - \mathbf{x}'(t-1)||^2,$

  for $t = 2, \dots, T-1,$ (A.3b)

  · $\underset{\mathbf{x}(T),\mathbf{x}'(T-1)}{\text{minimize}}\ l_T(\mathbf{x}(T), \mathbf{x}'(T-1)) - \boldsymbol{\alpha}^{(i)T}(T-1)\mathbf{x}'(T-1) + \rho||\mathbf{x}(T) - \mathbf{x}'(T-1)||^2,$ (A.3c)

  obtaining the new iterates $\mathbf{x}^{(i)}(t)$ for $t = 1, \dots, T$ and $\mathbf{x}'^{(i)}(t)$ for $t = 2, \dots, T$;

- Update the Lagrange multipliers as

$$\boldsymbol{\alpha}^{(i+1)}(t) \leftarrow \boldsymbol{\alpha}^{(i)}(t) + \rho(\mathbf{x}^{(i)}(t) - \mathbf{x}'^{(i)}(t-1)), \tag{A.4}$$

where $(\cdot)^T$ represents the transpose of a vector or a matrix.

APPENDIX B

FORWARD-BACKWARD ALGORITHM

Given the observations $\tilde{s}_n(t), t = 1, \ldots, T$, and the parameters p_n, q_n, p'_n and q'_n, we compute the posterior distribution $p(s_n(t)|\tilde{s}_n(1), \ldots, \tilde{s}_n(T))$ by using the forward-backward algorithm [54]. Accordingly, the posterior probability can be written as

$$\Pr(s_n(t) = 1|\tilde{s}_n(1), \ldots, \tilde{s}_n(T)) = \alpha_t \beta_t, \tag{B.1}$$

where the probability $\alpha_t = \Pr(s_n(t) = 1|\tilde{s}_n(1), \ldots, \tilde{s}_n(t))$ is obtained as a result of the forward pass, which transmits in a left-to-right fashion recursively; while the probability $\beta_t = \Pr(\tilde{s}_n(t+1), \ldots, \tilde{s}_n(T)|s_n(t) = 1)$ is obtained from the backward pass, which is computed in a right-to-left fashion recursively.

The distribution $\alpha_t = \Pr(s_n(t) = 1|\tilde{s}_n(1), \ldots, \tilde{s}_n(t))$ is called the belief state at time t. The update can be written as

$$\boldsymbol{\alpha}_t \propto \boldsymbol{\Psi}_t \odot (\boldsymbol{\Phi}^T \boldsymbol{\alpha}_{t-1}), \tag{B.2}$$

where $\Psi_t(j) = \Pr(\tilde{s}_n(1), \ldots, \tilde{s}_n(t)|s_t = j)$ is the local evidence, $\Phi_{i,j} = \Pr(s_t = j|s_{t-1} = i)$ is the transition matrix and $\odot$ represents element-wise vector multiplication.

In the backward pass, $\beta_t = \Pr(\tilde{s}_n(t+1), \ldots, \tilde{s}_n(T)|s_n(t) = 1)$ is the conditional likelihood of the future evidence given the hidden state at time t. The update can be written as

$$\boldsymbol{\beta}_{t-1} \propto \boldsymbol{\Phi}(\boldsymbol{\Psi}_t \odot \boldsymbol{\beta}_t), \tag{B.3}$$

where $\Psi_t(j) = \Pr(\tilde{s}_n(1), \ldots, \tilde{s}_n(t)|s_t = j)$ is the local evidence, $\Phi_{i,j} = \Pr(s_t = j|s_{t-1} = i)$ is the transition matrix and $\odot$ represents element-wise vector multiplication.

The algorithm (B.1) shows the message passing from both sides and then combining at each node. In this fashion, the forward-backward algorithm can limit estimation errors from spreading.

BIBLIOGRAPHY

[1] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent Component analysis and Applications*. Cambridge, MA: Academic Press, 2010.

[2] S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari, "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, pp. 109–116, 2003.

[3] T. Yoshioka, T. Nakatani, M. Miyoshi, and H. G. Okuno, "Blind separation and dereverberation of speech mixtures by joint optimization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 69–84, 2011.

[4] A. Cichocki and S. Amari, *Adaptive blind signal and image processing: learning algorithms and applications*. Hoboken, NJ: John Wiley & Sons, 2002.

[5] J. Karhunen, A. Hyvärinen, R. Vigario, J. Hurri, and E. Oja, "Applications of neural blind separation to signal and image processing," *IEEE International Coference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 131–134, 1997.

[6] A. Jourjine, S. Rickard, and Ö. Yilmaz, "Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2000.

[7] S. Wehr, I. Kozintsev, R. Lienhart, and W. Kellermann, "Synchronization of acoustic sensors for distributed ad-hoc audio networks and its use for blind source separation," *IEEE Sixth International Symposium on Multimedia Software Engineering*, 2004.

[8] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, 2000.

[9] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, pp. 37–52, 1987.

[10] G. H. Golub and C. H. Reinsch, "Singular value decomposition and least squares solutions," *Numerische Mathematik*, vol. 14, pp. 403–420, 1970.

[11] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, "Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios," *IEEE Wireless Communications*, vol. 23, pp. 60–67, 2016.

[12] J. Gozalvez, "New 3GPP standard for IoT," *IEEE Vehicular Technology Magazine*, vol. 11, pp. 14–20, 2016.

[13] J. Stone, "Independent component analysis: an introduction," *Trends in Cognitive Sciences*, vol. 6, pp. 59–64, 2002.

[14] A. Hyvärinen, *Advances in Independent Component Analysis and Learning Machines.* Cambridge, MA: Academic Press, 2015.

[15] S. M. Alavi and W. B. Kleijn, "Distributed linear blind source separation over wireless sensor networks with arbitrary connectivity patterns," *IEEE Transactions on Acoustics, Speech and Signal Processing*, pp. 3171–3175, March 2016.

[16] S. S. Ivrigh, S. M.-S. Sadough, and A. A. Ghorashi, "A blind source separation technique for spectrum sensing in cognitive radio networks based on kurtosis metric," *1st International eConference on Computer and Knowledge Engineering (ICCKE)*, Oct. 2011.

[17] D. Obradovic, N. Madhu, A. Szabo, and C. S. Wong, "Independent component analysis for semi-blind signal separation in MIMO mobile frequency selective communication channels," *IEEE International Joint Conference on Neural Networks*, July 2004.

[18] S. R. Curnew and J. Ilow, "Blind signal separation in MIMO OFDM system using ICA and fractional sampling," *ISSSE '07. International Symposium on Signals, Systems and Electronics.*, Aug. 2007.

[19] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, 3rd ed. West Sussex, United Kingdom: John Wiley & Sons, 2004.

[20] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing.* New York, NY: Springer, 2010.

[21] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 45, pp. 600–616, 1997.

[22] L. Rebollo-Neira and D. Lowe, "Optimized orthogonal matching pursuit approach," *IEEE Signal Processing Letters*, vol. 9, pp. 137–140, 2002.

[23] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, pp. 1045–1057, 2010.

[24] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, pp. 27–38, 2011.

[25] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Research*, vol. 37, pp. 3311–3325, 1997.

[26] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," *19th International Conference on Neural Information Processing Systems*, pp. 801–808, Dec. 2006.

[27] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," *26th Annual International Conference on Machine Learning*, June 2009.

[28] S. Bahrampout, N. M. Nasrabadi, A. Ray, and W. K. Jenkins, "Multimodal task-driven dictionary learning for image classification," *IEEE Transcations on Image Processing*, vol. 25, pp. 24–38, 2016.

[29] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transcations on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 791–804, April 2012.

[30] H. Ghanbari, H. Zayyani, and E. Yazdian, "Joint DOA Estimation and Array Calibration Using Multiple Parametric Dictionary Learning," *ArXiv e-prints*, Jul. 2017.

[31] M. Razaviyayn, M. Sanjabi, and Z.-Q. Luo, "A Stochastic Successive Minimization Method for Nonsmooth Nonconvex Optimization with Applications to Transceiver Design in Wireless Communication Networks," *ArXiv e-prints*, Jul. 2013.

[32] H. Raja and W. U. Bajwa, "Cloud K-SVD: A collaborative dictionary learning algorithm for big, distributed data," *IEEE Transcations on Signal Processing*, vol. 64, pp. 173–188, 2015.

[33] Y. Ding and B. D. Rao, "Dictionary Learning Based Sparse Channel Representation and Estimation for FDD Massive MIMO Systems," *ArXiv e-prints*, Dec. 2016.

[34] A. Masnadi-Shirazi and B. Rao, "Independent vector analysis incorporating active and inactive states," *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2009.

[35] A. Masnadi-Shirazi, W. Wang, and B. Rao, "Glimpsing independent vector analysis: Separating more sources than sensors using active and inactive states," *IEEE International Conference on Acoustics, Speech and Signal Processing*, June 2010.

[36] C. J. Hillar and F. T. Sommer, "When can dictionary learning uniquely recover sparse data from subsamples?" *IEEE Transcations on Information Theory*, vol. 61, pp. 6290–6297, 2015.

[37] C. Llerena, R. Gil-Pita, D. Ayllon, H. Sanchez-Hevia, I. Mohino-Herranz, and M. Rosa, "Synchronization for lassical blind source separation algorithms in wireless acoustic sensor networks," *2016 IEEE Statistical Signal Processing Workshop (SSP)*, June 2016.

[38] D. Cherkassky and S. Gannot, "Blind synchronization in wireless acoustic sensor networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 651–661, 2017.

[39] J. Chen, Z. J. Towfic, and A. H. Sayed, "Dictionary learning over distributed models," *IEEE Transactions on Signal Processing*, vol. 63, pp. 1001–1016, 2015.

[40] R. Gribonval and K. Schnass, "Dictionary identification - sparse matrix-factorization via ℓ_1-minimization," *IEEE Trans. on Information Theory*, vol. 56, pp. 3523–3539, 2010.

[41] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," *Proc, IEEE ICASSP*, vol. 5, pp. 2443–2446, 1999.

[42] L. N. Smith and M. Elad, "Improving dictionary learning: Multiple dictionary updates and coefficient reuse," *IEEE Signal Processing Letters*, vol. 20, no. 1, pp. 79–82, 2013.

[43] S. K. Sahoo and A. Makur, "Dictionary training for sparse representation as generalization of k-means clustering," *IEEE Signal Processing Letters*, vol. 20, no. 6, pp. 587–590, 2013.

[44] M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, "Dictionary learning for sparse representation: A novel approach," *IEEE Signal Processing Letters*, vol. 20, no. 12, pp. 1195–1198, 2013.

[45] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society*, vol. 58, no. 1, pp. 267–288, 1996.

[46] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning," *IEEE Transcations on Computational Imaging*, vol. 4, pp. 366–381, 2018.

[47] M. Hebiri and S. van de Geer, "The smooth-lasso and other $\ell_1 + \ell_2$-penalized methods," *Electronic Journal of Statistics*, vol. 5, pp. 1184–1226, 2011.

[48] K. Koh, S. J. Kim, and S. Boyd, "An interior-point method for large-scale ℓ_1-regularized logistic regression," *Journal of Machine Learning Research*, vol. 8, pp. 1519–1555, 2007.

[49] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society*, vol. 67, pp. 301–320, 2005.

[50] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, pp. 1–122, 2011.

[51] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, pp. 123–231, 2014.

[52] M. R.Hestenes, "Multiplier and gradient methods," *Journal of Optimization Theory and Applications*, vol. 4, pp. 302–320, 1969.

[53] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques.* Cambridge, MA: MIT Press, 2009.

[54] K. P. Murphy, *Machine Learning: A Probabilistic Perspective.* Cambridge, MA: MIT Press, 2012.

[55] C. M. Bishop, *Pattern Recognition and Machine Learning.* New York, NY: Springer, 2006.

[56] Z. Ghahramani, "An introduction to hidden markov models and bayesian networks," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, pp. 9–42, 2001.

[57] G. Mouret, J.-J. Brault, and V. Partovinia, "Generalized elastic net regression," *the Joint Statistical Meetings*, Aug. 2017.