Binary is Good: A Binary Inference Framework for Primary User Separation in Cognitive Radio Networks

Huy Nguyen and Rong Zheng
Department of Computer Science
University of Houston
Houston, TX 77204
E-mail: *nahuy@cs.uh.edu, rzheng@cs.uh.edu*

Zhu Han
Department of Electrical and Computer Engineering
University of Houston
Houston, TX 77204
E-mail: *zhan2@uh.edu*

***Abstract*—Primary users (PU) separation concerns with the issues of distinguishing and characterizing primary users in cognitive radio (CR) networks. We argue the need for PU separation in the context of collaborative spectrum sensing and monitor selection. In this paper, we model the observations of monitors as boolean OR mixtures of underlying binary latency sources for PUs, and devise a novel binary inference algorithm for PU separation. Simulation results show that without prior knowledge regarding PUs' activities, the algorithm achieves high inference accuracy. An interesting implication of the proposed algorithm is the ability to effectively represent n independent binary sources via (correlated) binary vectors of logarithmic length.**

I. Introduction

With the ongoing growth in wireless services, the demand for the radio spectrum has significantly increased. However, the spectrum resources are scarce and most of them have been already licensed to existing operators. Recent studies have shown that despite claims of spectral scarcity, the actual licensed spectrum remains unoccupied for long periods of time [1]. Thus, cognitive radio (CR) systems have been proposed [2], [3], [4] in order to efficiently exploit these spectral holes. CRs or secondary users (SUs) are wireless devices that can intelligently monitor and adapt to their environment, hence, they are able to share the spectrum with the licensed primary users (PUs), operating whenever the PUs are idle.

One key challenge in CR systems is spectrum sensing, i.e., SUs attempts to learn the environment and determine the presence and characteristics of PUs. Spectrum sensing can be done at SUs individually or cooperatively [5], [6], with or without the assistance of infrastructure supports such as dedicated monitor nodes and cognitive pilot channel (CPC) [7], [8], [9], [10]. At individual nodes, energy detection is commonly adopted to detect the presence of PUs, in which the detector computes the energy of the received signal and compares it to a certain threshold value to decide whether the desired signal is present or not. Energy detection has the advantage of short detection time but suffers from low accuracy compared to feature-based approaches such as cyclostationary detection [3], [4]. From the prospective of a CR system, it is often insufficient to detect PU activates in a single SU's vicinity ("is there any PU near me?"). Rather, it is important to determine the identity of PUs ("who is there?") as well as the distribution of PUs in the field ("where are they?"). We shall call these issues the PU separation problem.

To motive the need for PU separation, let us consider the following scenarios:

- Multiple SUs cooperatively infer the activities of PUs, some of which may be observable to only a subset of SUs. In this case, the SUs need to identify the PU-SU adjacency relationships. Blindly assuming all PUs are observable to all SUs would lead to inferior detection results.
- Dedicated monitors are employed for spectrum sensing. There exists redundancy in monitors' observations due to common PUs across multiple monitors. Such redundancy can be reduced by judiciously selecting a subset of monitors to report their spectrum sensing results. Furthermore, some monitors can be put to low-power modes for energy conservation.

Clearly, PU separation is a more challenging problem compared to node-level PU detection. The conventional wisdom would be that sophisticated methods such as feature-based detection are necessary. On the contrary, we find that through cooperation among monitors or SUs, not only the accuracy of energy detection can be improved as been demonstrated in several existing work, interestingly, PUs can be identified using solely binary information (due to thresholding in energy detection). The key to this somewhat surprising result is a binary inference framework that models the observations of SUs and monitors as boolean OR mixtures of underlying binary latency sources for PUs. It allows us to exploit the correlation structure among distributed binary observations. We develop an iterative algorithm, called Boolean Independent Component Analysis (bICA), to determine the underlying latent sources (i.e., PUs) and their active probability. In bICA, no prior information regarding the PUs' activities or observation noise is assumed. Given m monitors or SUs, up to $2^m - 1$ PUs can be inferred resulting in great efficiency. This is analogous to compressive sensing techniques [11], [12], where sub-Nyquist sampling rates are sufficient for spatially sparse data. Evaluation results show the effectiveness of bICA under realistic settings.

The rest of the paper is organized as follows. In Section II, the observation model is introduced. In Section III, we present

the bICA algorithm to determine the statistics of PU activities and the inference algorithm to decide which set of PUs are active. Evaluation results are detailed in Section IV followed by conclusions and future work in Section VI.

II. Model and Preliminaries

Consider a slotted system where the transmission activities of n PUs are modeled as a set of independent binary variables $\mathbf{y}$ with active probabilities $\mathcal{P}(\mathbf{y})$. The binary observations due to energy detection at the m monitor nodes (for the remaining of the paper, we do not distinguish monitor nodes and SUs) are modeled as an m-dimension binary vector with joint distribution $\mathcal{P}(\mathbf{x})$. It is assumed that the presence of any active PU in the surrounding of a monitor leads to positive detection. An (unknown) binary mixing matrix $\boldsymbol{G}$ is used to represent the relationship between the observable variables in $\mathbf{x}$ and the latent binary variables in $\mathbf{y} = [y_1, y_2, \ldots, y_n]$ as follows:

$$x_i = \bigvee_{j=1}^{n} (g_{ij} \wedge y_j),\ i = 1, \ldots, m, \tag{1}$$

where $\wedge$ is Boolean *AND* and $\vee$ Boolean *OR*, and g_{ij} the entry in the ith row and jth column of $\boldsymbol{G}$. For the ease of presentation, we introduce a short-hand notation as

$$\mathbf{x} = \boldsymbol{G} \oplus \mathbf{y}. \tag{2}$$

In essence, g_{ij} encodes whether monitor i can detect the transmissions of PU j. For a monitor i, the energy detection returns 1 when one or more of the PUs that the monitor can detect are active. $\boldsymbol{G}$ can be thought of as the adjacency matrix of an undirected bi-partite graph $G = (U, V, E)$, where $U = \{x_1, x_2, \ldots, x_m\}$ and $V = \{y_1, y_2, \ldots, y_n\}$ (Figure 1). An edge $e = (x_i, y_j)$ exists if $g_{ij} = 1$.

Consider a matrix $m \times T$ matrix X, which is the collection of T realizations of vector $\mathbf{x}$. The goal of binary independent component analysis (bICA) is to determine the latent independent random variables $\mathbf{y}$ and the binary mixing matrix $\boldsymbol{G}$ from X such that X can be decomposed into the mixing of realizations of $\mathbf{y}$. From $\boldsymbol{G}$ and $\mathbf{y}$, we can identify the PUs via the set of monitors, to which they are observable, and additionally infer the active probability of PUs.

Note that the above model differs from blind deconvolution with binary sources [13] in that in bICA, the mixing matrix is binary, and the mixtures are generated from OR operations instead of linear transforms over binary sources. Independent component analysis (ICA) has been studied in the past as a computational method for separating a multivariate signal into additive subcomponents supposing the mutual statistical independence of the non-Gaussian source signals. Most ICA methods assume linear mixing of continuous signals [14].

III. Binary Inference Framework

In this section, we first discuss the identifiability of binary independent sources for PUs from OR mixtures observed at the monitors, and then present an inference algorithm that determines the unknown mixing matrix and underlying sources.

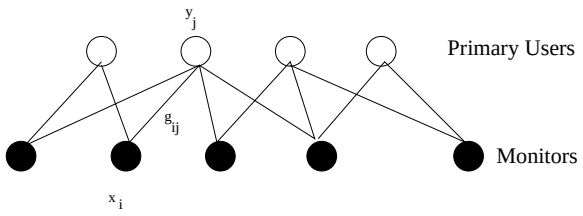


Fig. 1: Illustration of Monitors' Observations of PU

A. Identifiability

For m-dimension binary random vector $\mathbf{x}$, the number of possible combinations of binary observations are 2^m. From the data matrix X, the distribution of $\mathbf{x}$ can be estimated in a non-biased manner as the number of observations goes to infinity. We can initialize $n = 2^m - 1$ and $\boldsymbol{G}$ matrix of dimension $m \times 2^m - 1$ with rows being all possible binary combinations of length n (with the exception of all-0 entries). This results in a complete bipartite graph, where an edge exists between any two vertices in U and V, respectively. For a random variable $y_j \in V$, its neighbors in U is given by the non-zero entries in $g_{:,j}$ (i.e., the jth column of the $\boldsymbol{G}$ matrix). Thus, at most $2^m - 1$ independent components can be identified.

Define $p_i \triangleq \mathcal{P}(y_i = 1)$. Let the set

$$Y(\mathbf{x}) = \{\mathbf{y} \mid \bigvee_{j=1}^{n} (g_{ij} \wedge y_j) = x_i, \forall i\}.$$

Therefore,

$$\begin{aligned} \mathcal{P}(\mathbf{x}) &= \mathcal{P}(\mathbf{y} \in Y(\mathbf{x})) = \textstyle\sum_{\mathbf{y} \in Y(\mathbf{x})} \mathcal{P}(\mathbf{y}), \\ &= \textstyle\sum_{\mathbf{y} \in Y(\mathbf{x})} \prod_{l=1}^{2^m-1} p_l^{y_l} (1 - p_l)^{1-y_l} \end{aligned} \tag{3}$$

where $\mathcal{P}(\mathbf{y})$ is the joint probability of $\mathbf{y}$. The last equality is due to the independence of y_i's.

Given the distribution of random vectors $\mathbf{x} \in \{0, 1\}^m$, $2^m - 1$ independent equations can be obtained from (3) due to the OR mixture model in (1). As there are $2^m - 1$ unknowns (i.e., $p_i, i = 1, \ldots, n$), the probability of y_j can be determined. Clearly, ambiguity exists if two or more independent sources have the same set of neighbors in U (or equivalently, identical columns in $\boldsymbol{G}$). In this case, binary information is insufficient to distinguish these sources.

The set of equations in (3) are polynomials of sum product forms, which are difficult to solve. This necessitates the design of specialized algorithms. In the rest of the paper, we fix $\boldsymbol{G}$ to be a $m \times 2^m - 1$ adjacency matrix for the complete bipartite graph. Furthermore, we order $\boldsymbol{G}$ such that $g_{kl} = 1$ if $l \ll k = 1$, for $k = 0, \ldots, m - 1$, where $\ll$ is the bit shift to the left. If the resulting $p_l = 0$ for some l, this implies the corresponding column $g_{:,l}$ can be removed from $\boldsymbol{G}$. Abusing the notation slightly, we consider a binary random vector of a set of independent sources and possibly some zero elements a binary independent random vector.

B. The inference algorithm

Before proceeding to the details of the algorithm, we first present a few technical lemmas.

Lemma 1: Consider a set $\mathbf{x} = [x_1, x_2, \ldots, x_{k-1}, x_k]$ generated by the data model in (1), i.e., $\exists$ binary independent sources $\mathbf{y}$, s.t., $\mathbf{x} = G \oplus \mathbf{y}$. The conditional random vector $\mathbf{x}_{x_k=0} = [x_1, x_2, \ldots, x_{k-1} | x_k = 0]$ corresponds to realizations of $\mathbf{x}$ when $x_k = 0$. Then, $\mathbf{x}_{x_k=0} = \boldsymbol{G'} \oplus \mathbf{y'}$, where $\boldsymbol{G'} = \boldsymbol{G}_{:,1:2^{k-1}}$ and $\mathcal{P}(y'_l = 1) = \mathcal{P}(y_l = 1)$ for $l = 1, \ldots, 2^{k-1}$.

Proof:

$$\begin{aligned} & P(x_1, x_2, \ldots x_{k-1}, x_k = 0) \\ = \; & P(x_1, x_2, \ldots x_{k-1} | x_k = 0) P(x_k = 0) \\ = \; & \sum_{\mathbf{y} \in Y(\mathbf{x})} \prod_{l=1}^{2^k} p_l^{y_l} (1 - p_l)^{1 - y_l} \\ = \; & \sum_{\substack{\mathbf{y}_{1:2^{k-1}} \in Y(\mathbf{x}_{1:k-1}) \\ y_l = 0, \forall g_{kl} = 1}} \prod_{g_{kl}=0} p_l^{y_l} (1 - p_l)^{1 - y_l} \prod_{g_{kl}=1} (1 - p_l) \end{aligned} \tag{4}$$

since $P(x_k = 0) = \prod_{g_{kl}=1} (1 - p_l)$, we have

$$\begin{aligned} & P(x_1, x_2, \ldots x_{k-1} | x_k = 0) \\ = \; & \sum_{\mathbf{y}' \in Y(\mathbf{x}_{1:k-1})} \prod_{l=1} (p'_l)^{y'_l} (1 - p'_l)^{1 - y'_l} \\ = \; & \sum_{\substack{\mathbf{y}_{1:2^{k-1}} \in Y(\mathbf{x}_{1:k-1}) \\ y_l = 0, \forall g_{kl} = 1}} \prod_{g_{kl}=0} p_l^{y_l} (1 - p_l)^{1 - y_l} \end{aligned} \tag{5}$$

Clearly, by setting $\mathcal{P}(y'_l = 1) = \mathcal{P}(y_l = 1)$ for $l = 1, \ldots, 2^{k-1}$, the above equality holds. In another word, the conditional random vector $\mathbf{x}_{x_k=0} = \boldsymbol{G'} \oplus \mathbf{y'}$ for $\boldsymbol{G'} = \boldsymbol{G}_{:,1:2^{k-1}}$. ∎

The above lemma establishes that the conditional random vector $\mathbf{x}_{x_k=0}$ can be represented as an OR mixing of 2^{k-1} independent components. Furthermore, the set of the independent components are the same as the first 2^{k-1} independent components of $\mathbf{x}$ (under proper ordering).

Consider a sub-matrix of the data matrix X, $X^0_{(k-1)\times T}$ where the rows correspond to observations of $x_1, x_2, \ldots, x_{k-1}$ for $t = 1, 2, \ldots, T$ such that $x_{kt} = 0$. Define $X_{(k-1)\times T}$, which consists of the first $k-1$ rows of X. Suppose that we have computed the BICA for data matrices $X^0_{(k-1)\times T}$ and $X_{(k-1)\times T}$. From Lemma 1, we know that $X^0_{(k-1)\times T}$ is realization of OR mixing of independent components, denoted by $\mathbf{y}^0_{k-1}$. Furthermore, $\mathcal{P}(\mathbf{y}^0_{k-1}(l) = 1) = \mathcal{P}(y_l = 1)$ for $l = 1, \ldots, 2^{k-1}$. Clearly, $X_{(k-1)\times T}$ is realization of OR mixing of independent components, denoted by $\mathbf{y}^*_{k-1}$ ('*' for *do not care*). Additionally, it is easy to see that the following holds:

$$\begin{aligned} & \mathcal{P}(\mathbf{y}^*_{k-1}(l) = 1) \\ = \; & 1 - (1 - \mathcal{P}(\mathbf{y}^0_{k-1}(l) = 1))(1 - p_{l+2^{k-1}}) \\ = \; & 1 - (1 - p_l)(1 - p_{l+2^{k-1}}), \end{aligned}$$

where $l = 0, 1, \ldots, 2^{k-1}$. Therefore,

$$p_l = \mathcal{P}(\mathbf{y}^0_{k-1}(l) = 1), l = 1, \ldots, 2^{k-1}, and \tag{6}$$

$$p_{l+2^{k-1}} = 1 - \frac{1 - \mathcal{P}(\mathbf{y}^*_{k-1}(l) = 1)}{1 - \mathcal{P}(\mathbf{y}^0_{k-1}(l) = 1)}, l = 1, \ldots, 2^{k-1} \tag{7}$$

Define $\mathcal{F}(\mathbf{x})$ as the frequency of the random vector $\mathbf{x}$ from the data matrix X. To this end, we have the following iterative algorithm as illustrated in Algorithm 1.

```
FindBICA()
input : Data matrix X_{m×T}
init  : n = 2^m − 1;
p_k = 0, ∀k = 1, ..., n;
Set G be a m × (2^m − 1) matrix with rows corresponding all
possible binary vector of length m.
if m = 1 then
    p_0 = F(x_1 = 0);
    p_1 = F(x_1 = 1);
else
    p[0 : 2^{m−1} − 1] = FindBICA(X^0_{(m−1)×T});
    p'[0 : 2^{m−1} − 1] = FindBICA(X_{(m−1)×T});
    for l = 0, ... 2^{m−1} − 1 do
        p_{l+2^{m−1}} = 1 − (1−p'_l)/(1−p_l)
```

Algorithm 1: Iterative bICA inference algorithm

Computation complexity: Let $S(m)$ be the computation time for finding BICA given $X_{m\times T}$. From Algorithm 1, we have,

$$S(m) = 2S(m-1) + c2^m,$$

where c is a constant. It is easy to verify $S(m) = cm2^m$. Therefore, Algorithm 1 has an exponential computation complexity with respect to m. This is clearly undesirable for large m's. However, we notice that in practice, correlations among x_i's exhibit locality, and the $\boldsymbol{G}$ matrix tends to be sparse. Instead of using a complete bipartite graph to represent $\boldsymbol{G}$, the degree of vertices in V (or the number of non-zero elements in $g_{:,k}$) tend to be much less than m. More specifically, for every pair i and k, computer

$$Cov(i,k) = \frac{\sum_t^T X_{it} X_{kt}}{T} - \frac{\sum_t^T X_{it}}{T} \frac{\sum_t^T X_{kt}}{T}.$$

If $Cov(i,k) < \epsilon$, where ϵ is a small value (e.g., upper confidence bound of $Cov(i,k)$ estimate), we can remove the corresponding columns in $\boldsymbol{G}$ and elements in $\mathbf{y}$.

IV. Evaluation

In this section, we first introduce the performance metrics and then present evaluation results on a synthetic data set with varying number of PUs.

A. Performance metrics

Denote $\hat{p}$ and $\hat{\boldsymbol{G}}$ the inferred PUs' active probabilities and the mixing matrix, respectively. Note that order of PUs in the inferred model may be different from that in the original model. To measure the closeness of the inferred model with the

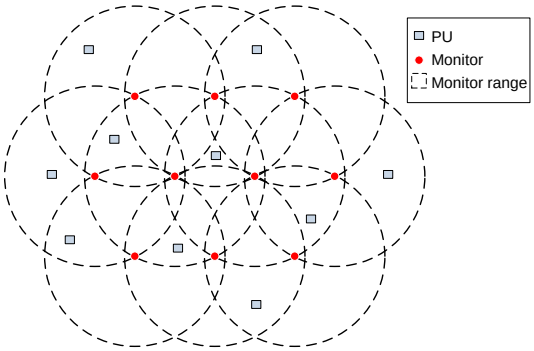


Fig. 2: An example scenario with 10 PUs and 10 monitors.

ground truth, we first select the top n (with highest probability) components of the inferred model and remove the remaining columns in $\hat{\boldsymbol{G}}$, where n is number of PUs. Then, we permutate columns of $\hat{\boldsymbol{G}}$ (corresponding to different PUs) such that a good match between $\hat{\boldsymbol{G}}$ and $\boldsymbol{G}$ can be found. Once the ordering is decided, active probabilities are compared.

To evaluate performance of the proposed algorithm, we introduce the following two metrics.

- *Normalized Hamming Distance:* This metric measures accuracy of the inferred mixing matrix defined as the Hamming distance between $\hat{\boldsymbol{G}}$ and $\boldsymbol{G}$ divided by the size of $\boldsymbol{G}$.

$$\bar{H} \triangleq \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}|\hat{g}_{ij}-g_{ij}|. \tag{8}$$

  Clearly, the smaller the $\bar{H}$ the better.
- *Root Mean Square Error Ratio (RMSE Ratio):* RMSE Ratio measures the deviation of inferred PU activities from original values.

$$\bar{P} \triangleq \sqrt{\frac{\sum_{i=1}^{n}(\hat{p}_i-p_i)^2}{n}}\Big/\frac{\sum_{i=1}^{n}p_i}{n} \tag{9}$$

B. Experimental results

10 monitors and n PUs are deployed in an 1000x1000 square meter area. with n varying from 5 to 20. Locations of PUs are chosen arbitrarily with a restriction that no two PUs can be observed by the same set of monitors. The PUs' transmit power is 20mW, the noise level is -70dbm, and the propagation loss factor is 3. The SNR threshold for monitors to detect primary users is 5dB. Elements in the binary mixing matrix $\boldsymbol{G}$ are either 1 or 0, depending whether or not received signals are higher than the threshold. PUs' activities are modeled as a two-stage Markov chain with transition probabilities uniformly distributed over $[0, 1]$. A monitor reports the channel occupancy if any detectable PU is active. Number of monitors' observations T is 5000.

One example scenario with 10 PUs is illustrated in Fig. 2 and the inferred result is available in Fig. 3. Non-zero entries and zero entries of $\boldsymbol{G}$ and $\hat{\boldsymbol{G}}$ are shown as black and white dots, respectively. The entry-wise difference matrix $|\boldsymbol{G}-\hat{\boldsymbol{G}}|$ is given in the bottom graph. In this case, only the first row (corresponding to the first monitor) contains some errors.

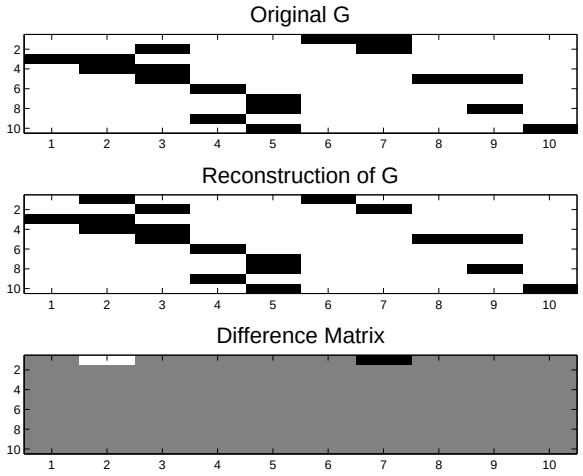
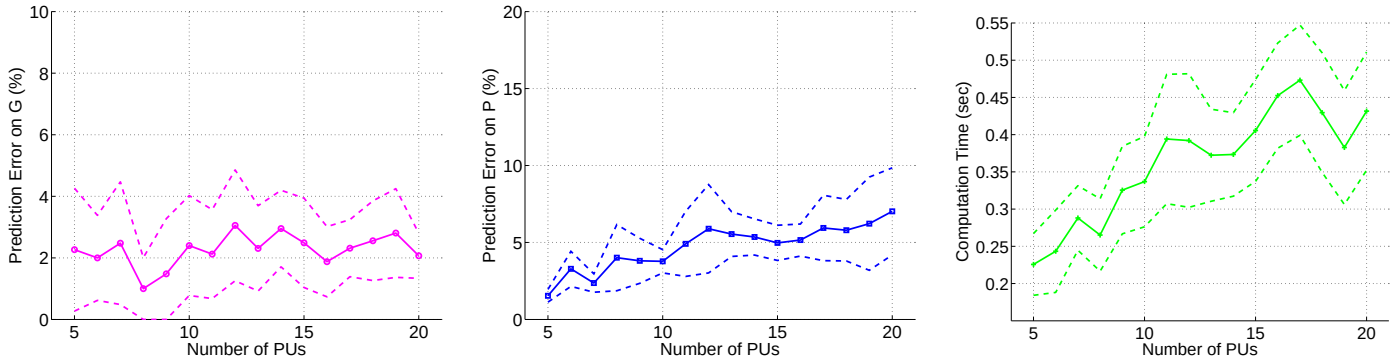


Fig. 3: G, $\hat{G}$ and the difference matrix of the example scenario in Fig. 2. Rows correspond to monitors and columns to PUs.

We have implemented our algorithm in Matlab. All experiments are conducted on a desktop PC with an Intel Core 2 Duo T5750@2.00GHz processor and 2GB RAM. Experiment results over 20 runs for each PU setting are shown in Fig. 4. From Fig. 4(a), we observe that the inferred mixing matrix $\hat{\boldsymbol{G}}$ is mostly correct even for a large number of PUs. As the number of PUs increases, errors in the inferred active probabilities tend to increase though within 10% on average as shown in Fig. 4(b). Recall that PUs are ordered based on their respective columns in $\hat{\boldsymbol{G}}$. Therefore, errors in $\hat{\boldsymbol{G}}$ may have a cascading effect since we may compare the wrong pair of PUs as a result. From Fig. 4(c), we see the computation time is negligible (under 1 second). Due to space limit, experimental results with varying numbers of samples are omitted. We observe that having more samples beyond 5000 has marginal impact on the accuracy of the inference.

V. Related Work

Most ICA methods assume linear mixing of continuous signals [14]. A special variant of ICA, called binary ICA, considers boolean mixing (e.g., OR, XOR etc.) of binary signals. Existing solutions to BICA mainly differ in their assumptions of prior distribution of the mixing matrix, noise model, and/or hidden causes. In [15], Yeredor considers BICA in XOR mixtures and investigates the identifiability problem. A deflation algorithm is proposed for source separation based on entropy minimization. In [15] the number of independent random sources K is assumed to be known. Furthermore, the mixing matrix is an K-by-K invertible matrix. In [16], infinite number of hidden causes following the same Bernoulli distribution are assumed. Reversible jump Markov chain Monte Carlo and Gibbs sampler techniques are applied. In contrast, in our model, the hidden causes may follow different distribution and the mixing matrix tends to be sparse. Streith *et al.* [17] study the problem of multi-assignment clustering for boolean data, where the observations either draw from a signal



(a) Normalized hamming distance (b) RMSE ratio (c) Computation time

Fig. 4: Normalized Hamming distance between the true and estimated $\boldsymbol{G}$ matrix (left), RMSE Ratio between the true and inferred PUs' active probability (middle) and CPU Computation Time (right) as number of PUs varies from 5 to 20. Solid lines indicate average result over 20 runs with different seeds and dashed lines are standard deviations.

following OR mixtures or from a noise component. The key assumption made in this work is that elements of the matrix X are conditionally independent given the model parameters. This greatly reduces the computational complexity and makes the scheme amenable to gradient descent optimization solution this assumption is in general invalid. In [18], the problem of factorization and de-noise of binary data due to independent continuous sources is considered. The sources are assumed to be following beta distribution. Finally, [16] consider under-presented case of less sensors than sources with continuous noise, while [18], [17] deals with over-determined case, where number of sensors are much larger. In this work, we consider primarily the under-presented cases encountered in data networks.

VI. Conclusion

In this paper, we propose a binary inference framework for PU separation in cognitive radio networks. Simulation results show the proposed algorithm can achieve less than 4% error in the inferred mixing matrix and less than 10% error for the active probability of PUs. PUs can thus be seperated based on the set of monitors to which they are observable. As ongoing work, we are investigating effect of noise on the inference accuracy and aiming to develop a robust inference framework.

Acknowledge

This work is funded in part by the National Science Foundation under grants CNS-0953377, CNS-0905556, CNS-091046, and CNS-0832084.

References

[1] Federal Communications Commission, "Spectrum policy task force report," *Report ET Docket no. 02-135*, Nov. 2002.

[2] J. Mitola and G. Q. Maguire, "Cognitive radio: Making software radios more personal," *IEEE Pers. Commun.*, vol. 6, pp. 13–18, Aug. 1999.

[3] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE JSAC*, vol. 23, pp. 201–220, Feb. 2005.

[4] D. Niyato, E. Hossein, and Z. Han, *Dynamic spectrum access in cognitive radio networks*. Cambridge, UK: Cambridge University Press, 2009.

[5] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *in Proc. of IEEE Symp. New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, USA, nov 2005.

[6] C. Sun, W. Zhang, and K. B. Letaief, "Cooperative spectrum sensing for cognitive radios under bandwidth constraint," in *in Proc. of IEEE Wireless Communications and Networking Conf.*, Hong Kong, China, mar 2007.

[7] P. Houz, S. B. Jemaa, and P. Cordier, "Common pilot channel for network selection," in *in Proc. of IEEE Vehicular Technology - Spring*, Melbourne, Australia, May 2006.

[8] M. Filo, A. Hossain, A. R. Biswas, and R. Piesiewicz, "Cognitive pilot channel: Enabler for radio systems coexistence," in *in Proc. of Second Int. Workshop on Cognitive Radio and Advanced Spectrum Management*, alborg, Denmark, May 2009.

[9] O. Sallent, J. Perez-Romero, R. Agusti, and P. Cordier, "Cognitive pilot channel enabling spectrum awareness," in *in Proc. Int. Conf. on Communications, Workshop on*, Dresden, Germany, Jun. 2009.

[10] J. Perez-Romero, O. Salient, R. Agusti, and L. Giupponi, "A novel on-demand cognitive pilot channel enabling dynamic spectrum allocation," in *in Proc. of IEEE Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Dublin, Ireland, Apr. 2007.

[11] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, vol. 52, no. 2, pp. 489–509, February 2006.

[12] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. on Information Theory*, vol. 52, no. 12, pp. 5406–5425, December 2006.

[13] Y. Li, A. Cichocki, and L. Zhang, "Blind source estimation of fir channels for binary sources: a grouping decision approach," *Signal Process.*, vol. 84, no. 12, pp. 2245–2263, 2004.

[14] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Netw.*, vol. 13, no. 4-5, pp. 411–430, 2000.

[15] A. Yeredor, "Ica in boolean xor mixtures," in *ICA*, 2007, pp. 827–835.

[16] F. W. Computer and F. Wood, "A non-parametric bayesian method for inferring hidden causes," in *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence (UAI*. AUAI Press, 2006, pp. 536–543.

[17] A. P. Streich, M. Frank, D. Basin, and J. M. Buhmann, "Multi-assignment clustering for boolean data," in *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*. New York, NY, USA: ACM, 2009, pp. 969–976.

[18] A. Kabn and E. Bingham, "Factorisation and denoising of 01 data: a variational approach," *Neurocomputing, special*, vol. 71, no. 10-12, 2009.