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

Huy Nguyen, Rong Zheng, and Zhu Han*

Department of Computer Science

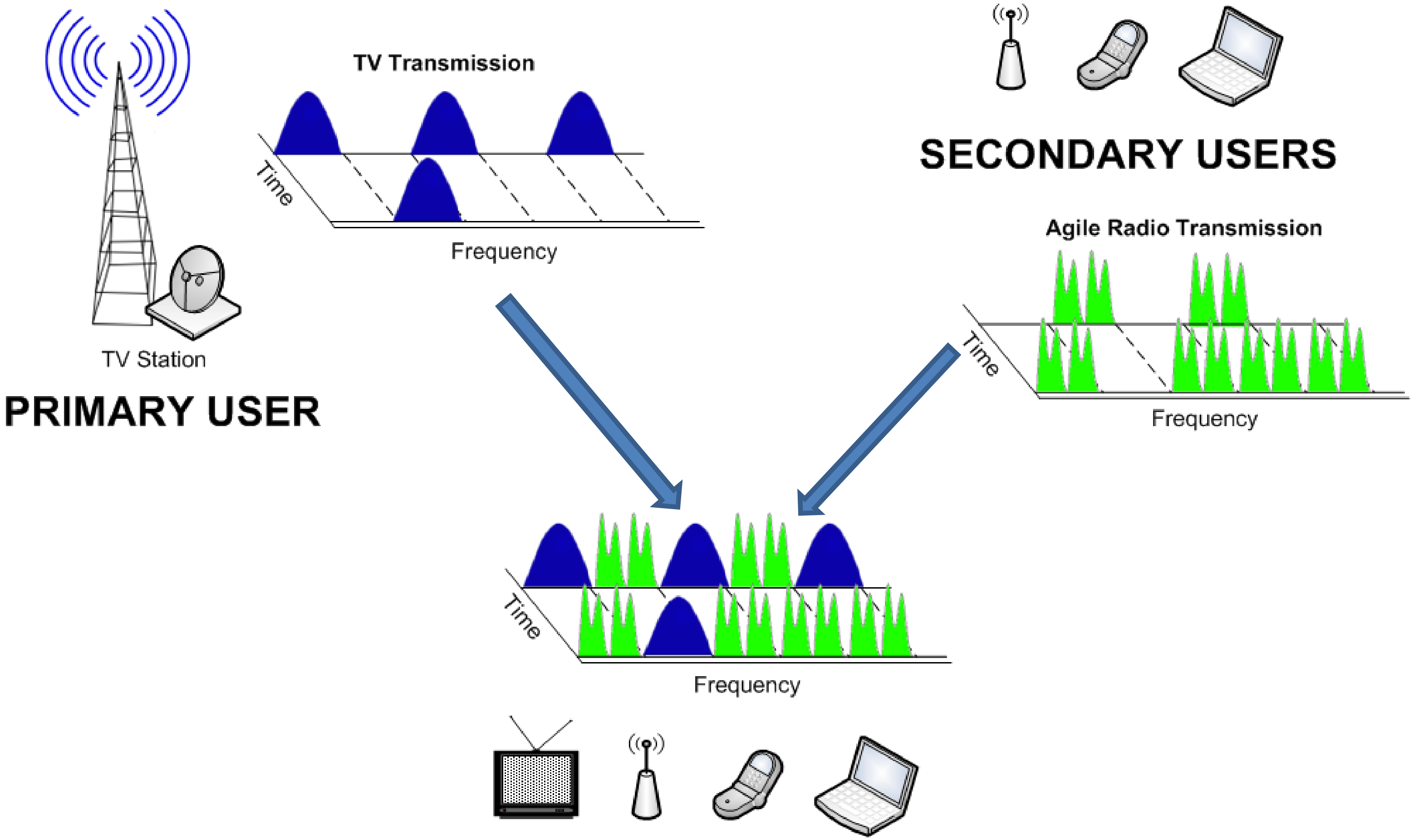
*Department of Electrical & Computer Engineering

University of Houston, TX 77204

Outline

1. Introduction
2. Problem Statement
3. Binary Inference Algorithm
4. Simulation and Experiments
5. Conclusion and Future Work

1. Cognitive Radio Systems



1. Spectrum Sensing

- Key challenge in CR systems
- Determine presence and characteristics of PUs
- Can be done at SUs individually / cooperatively
- Motivation scenarios:
  - Some PUs are visible to only a subset of SUs in a SU cooperative environment
  - Redundancy in dedicated monitors' observations

→ PU Separation Problem

1. PU Separation Problem

- Address the following questions:
  - What is the identity of PUs activate within a SU's vicinity?
  - What is the distribution of PUs in the field?
  - What is the characteristic of each identified PU?
- SUs cooperatively detect PUs using only binary information (thresholding energy detection)
- Can be effectively solved by Binary Independent Component Analysis (bICA)

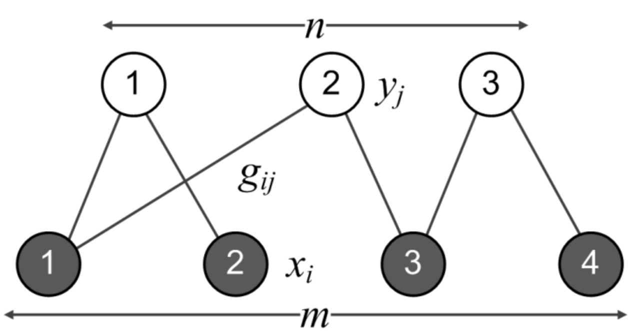
2. Problem Formulation

- *n* independent PUs: $\mathbf{y} = [y_1, y_2, \dots, y_n]$
- *m* binary monitor nodes (SUs): $\mathbf{x} = [x_1, x_2, \dots, x_m]$
- Binary mixing matrix:

$$\boldsymbol{G} = g_{ij} \in \{0, 1\}, i = [1, \dots, m], j = [1, \dots, n]$$

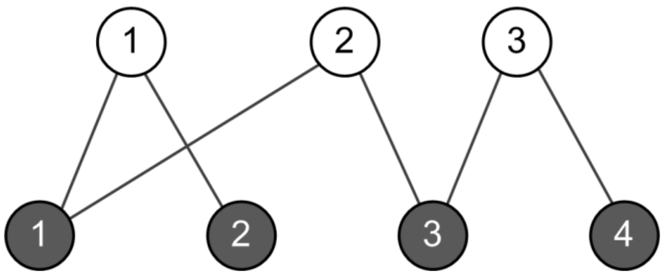
- Relationship between PUs and SUs

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



$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

2. A Toy Example



$$\mathbf{x} = \boldsymbol{G} \otimes \mathbf{y}$$

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & \ldots \\ 1 & 0 & 0 & 0 & 0 & \ldots \\ 1 & 0 & 1 & 1 & 1 & \ldots \\ 0 & 0 & 0 & 1 & 1 & \ldots \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \ldots \\ 1 & 0 & 1 & 1 & 0 & \ldots \\ 0 & 0 & 0 & 1 & 1 & \ldots \end{bmatrix}$$

$\mathbf{x}$ $\boldsymbol{G}$ (unknown) $\mathbf{y}$ (unknown)

Outline

1. Introduction
2. Problem Statement
3. Binary Inference Algorithm
4. Simulation and Experiments
5. Conclusion and Future Work

3. Binary Independent Component Analysis

- Goal: Infer the mixing matrix $\boldsymbol{G}$ and active probability vector $\boldsymbol{p}$ from observation matrix X
- Use ICA/PCA then apply a quantization method to convert to binary simply won't work
- Initialize: $\boldsymbol{G} = m \times 2^m - 1$ matrix

  $\boldsymbol{p} = 1 \times 2^m - 1$ vector
- $2^m - 1$: all possible PU connections (no PU have a same set of connections to SUs)

3. The Inference Algorithm

- **Input**: Observation matrix X
- **Output**: Mixing matrix ***G***, active prob. ***p***

```
FindBICA ()
if m = 1 then
    p_0 = F(x_1 = 0)
    p_1 = F(x_1 = 1)
else
    p^0_{1:2^{m-1}-1} = FindBICA (X^0_{(m-1)×T})
    p^*_{1:2^{m-1}-1} = FindBICA (X_{(m-1)×T})
    for l = 1, ... 2^{m-1} - 1 do
        p_{l+2^{m-1}} = 1 - (1-p^*_l)/(1-p^0_l)
```

$$\begin{array}{l}
\texttt{FindBICA}\ () \\
\textbf{if } m = 1 \textbf{ then} \\
\quad p_0 = \mathcal{F}(x_1 = 0) \\
\quad p_1 = \mathcal{F}(x_1 = 1) \\
\textbf{else} \\
\quad p^0_{1:2^{m-1}-1} = \texttt{FindBICA}\ (X^0_{(m-1)\times T}) \\
\quad p^*_{1:2^{m-1}-1} = \texttt{FindBICA}\ (X_{(m-1)\times T}) \\
\quad \textbf{for } l = 1, \ldots 2^{m-1} - 1 \textbf{ do} \\
\quad\quad p_{l+2^{m-1}} = 1 - \frac{1-p^*_l}{1-p^0_l}
\end{array}$$

* *more details in the paper*

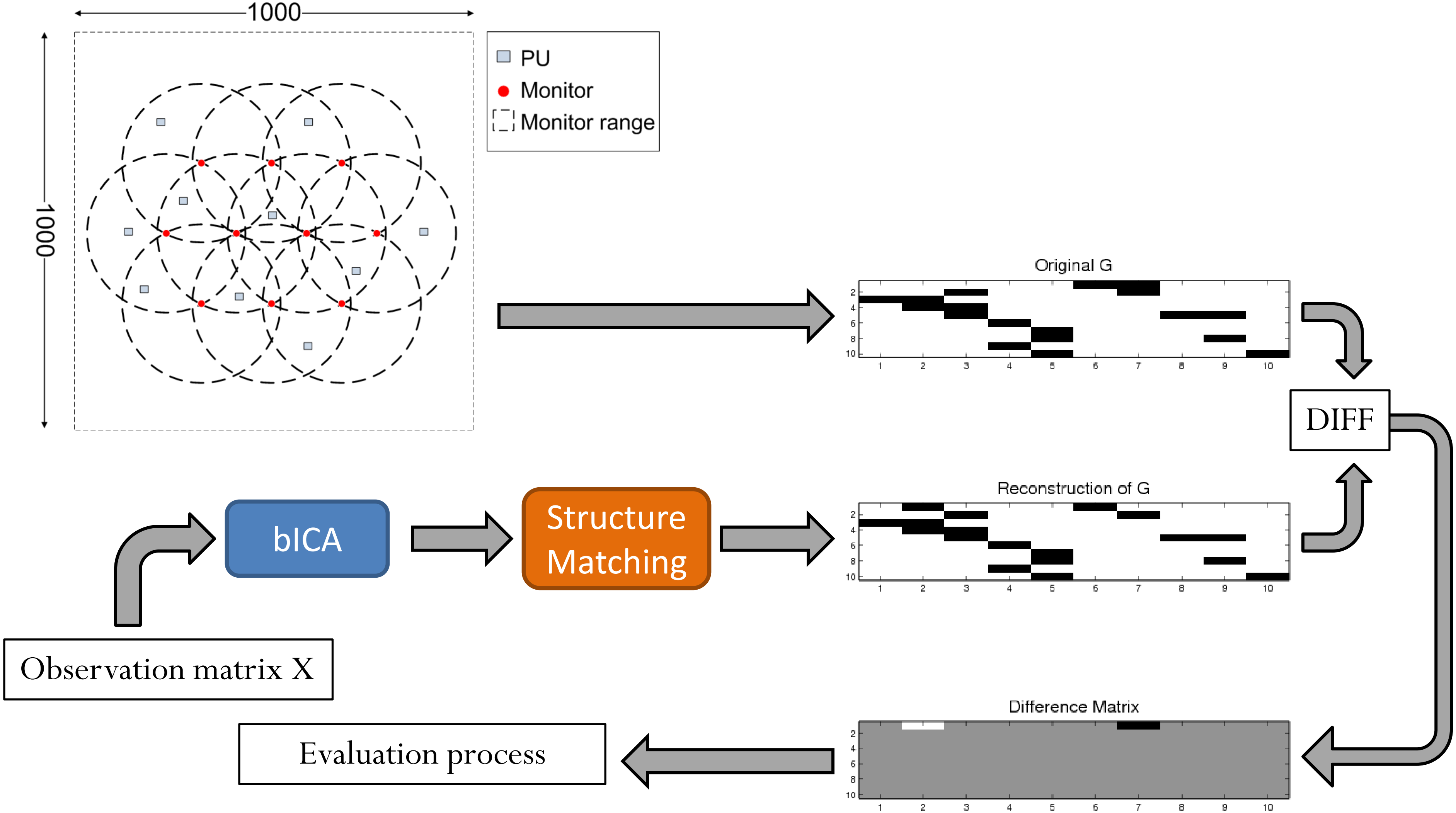
4. Simulation Setup

- 10 monitors (SUs) are deployed on an 1000x1000 square meter area
- 5 to 20 PUs are placed randomly on the area (random topology) – no PU observes a same set of monitors
- PUs' activities are modeled as 2-stage MC with transition probability in [0, 1]
- Number of monitor observations $T = 5000$
- Simulation platform: Matlab 2009b, Windows 7 running on Intel Core 2 Duo T5750@2.00GHz and 2GB RAM

4. Simulation Setup

- Noise: randomly flip an entry of X with probability p_e
- Structure Matching Problem:
  - Motivation: how to evaluate accuracy of bICA when:
    - Inferred result ***G*** may contain up to $2^m - 1$ components, and they could be in any order
    - Some inferred components may be slightly different compared to the ground truth (1-2 bits different)
  - Solution:
    - Select the top n components (with highest active prob.) in ***G***
    - Permute these n columns in ***G*** to find a best match with the ground truth (using Hungarian algorithm)

4. bICA Framework



4. Performance Metrics

Measure accuracy and speed of the proposed method

1. Normalized Hamming Distance

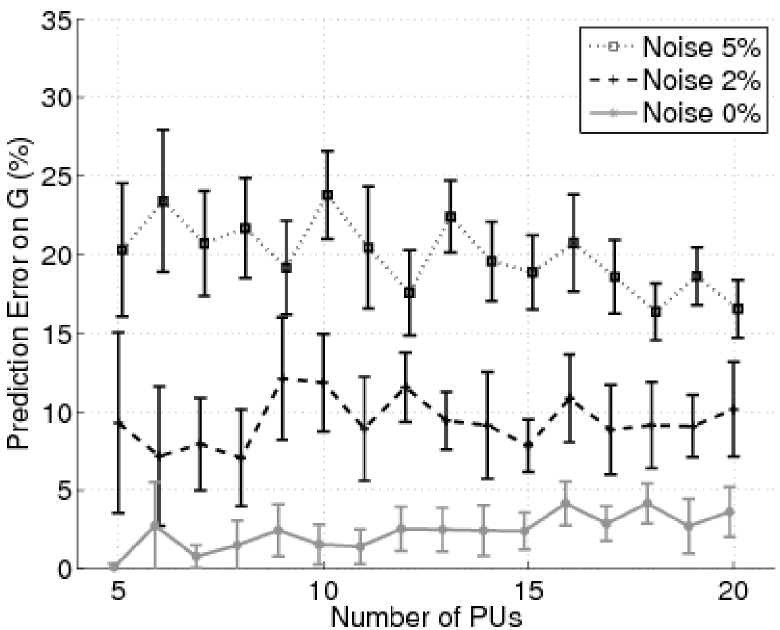
$$\bar{H}(\boldsymbol{G}, \widehat{\boldsymbol{G}}) \triangleq \frac{1}{mn} \sum_{i=1}^{n} d^{H}(g_{:,i}, \widehat{g}_{:,i})$$

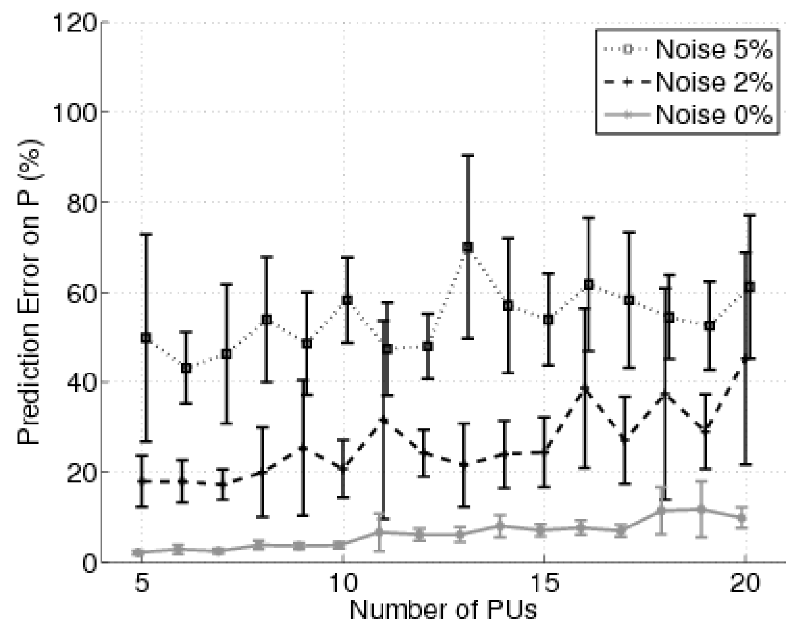
2. Root Mean Square Error Ratio

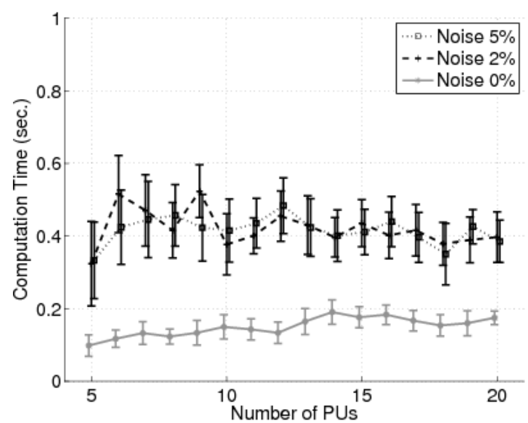
$$\bar{P} \triangleq \sqrt{\frac{\sum_{i=1}^{n} (\widehat{p}_i' - p_i)^2}{n}} \Big/ \frac{\sum_{i=1}^{n} p_i}{n}$$

3. Computation Time

4. Evaluation Results







5. Conclusion and Future Work

- Derive and address PU separation problem in CR systems with binary data
- Propose bICA – a binary inference framework
- Simulation results show that bICA can effectively solve PU separation problem
- Future Work
  - Real environment experiments
  - The inverse problem
  - Reducing noise effect