DISTRIBUTED LINEAR BLIND SOURCE SEPARATION OVER WIRELESS SENSOR NETWORKS WITH ARBITRARY CONNECTIVITY PATTERNS

S.R. Mir Alavi and W. Bastiaan Kleijn
School of Engineering and Computer Science, Victoria University of Wellington, New Zealand

Introduction

- Communication is performed by transmitting signals through a medium
- Signals can be mixed in the transport medium before they are received
- The operation of separating source signals without prior information about the sources is referred to as blind source separation
- Wireless sensor networks (WSN) form a natural platform for effective, low cost BSS
  - facilitate a good coverage of an area
  - low deployment cost
- In the context of sensor networks, research on blind source separation can be divided into:
  - centralized approaches
  - de-centralized approaches

Problem

- In centralized approaches, the recordings of all sensors (microphones) are transmitted to a centralized processor. Drawbacks:
  - lack of scalability
  - high power consumption
  - the need for hardware that can transmit over long distances
- Drawbacks of the existing de-centralized approaches include:
  - the need for a full connectivity pattern over the graph of the network
  - high computational load

Contributions

- Developing a distributed BSS algorithm with the following features:
  - applicable to any connected graph with any connectivity patterns
  - low computational load
  - low power consumption

Adaptive Linear BSS

- Linear generative model:

$$b_{N\times 1}(t) = A_{N\times M} s_{M\times 1}(t) + n_{N\times 1}(t) \quad (1)$$

- Likelihood function:

$$\mathrm{p}(b|A) = \int \mathrm{p}(b|s,A)\,\mathrm{p}(s)ds \quad (2)$$

- Natural gradient learning rule:

$$\triangle A \propto AA^T \frac{\partial}{\partial A} \log \mathrm{p}(b|A)$$
$$\text{or} \quad A(t+1) = A(t) - \mu A(t) F[y(t)] \quad (3)$$

where $F[y(t)] = I - \psi[y(t)]y(t)^T$,
$y(t) = A^{-1}(t)b(t)$ and

$$\psi[y] = [\psi_1[y_1], ..., \psi_M[y_M]]^T$$
$$\psi_i[y_i] = -\frac{d}{dy_i} \ln \mathrm{p}(y_i) \quad (4)$$

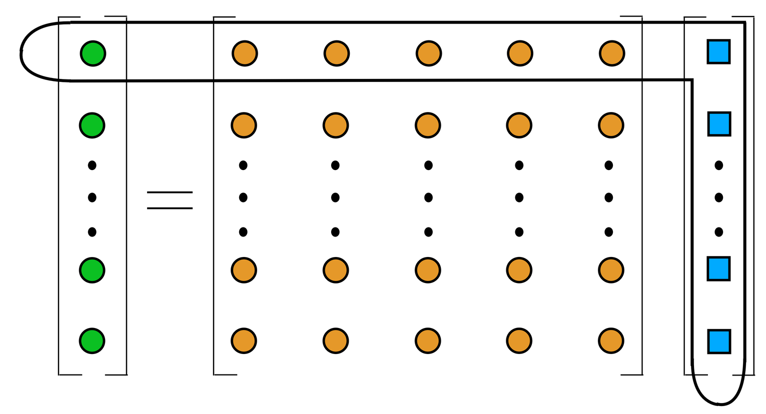
- A more numerically stable algorithm:

$$F[y(t)] = \Lambda(t) - \psi[y(t)]y(t)^T \quad (5)$$

where $\Lambda(t) = \mathrm{diag}[\mathrm{diag}[\psi[y(t)]y(t)^T]]$

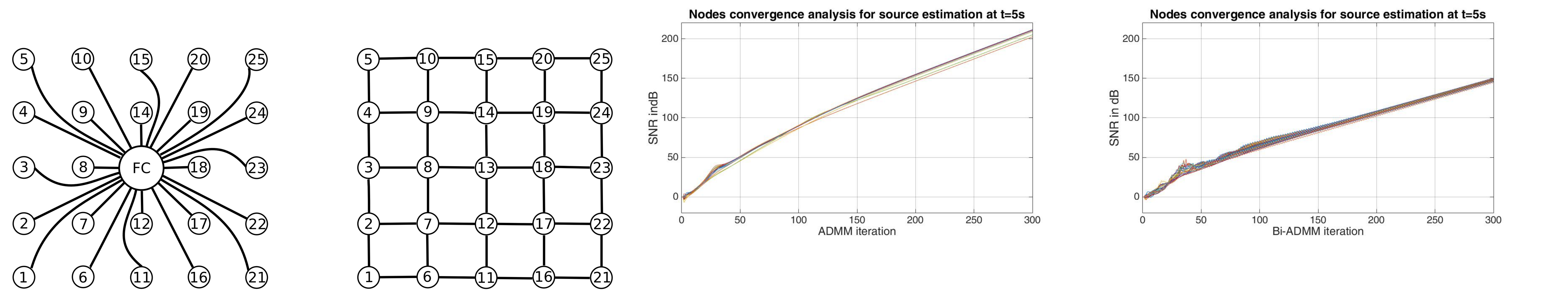
Split Objective Function

- Finding $y(t) = A^{+}(t)b(t)$ is equivalent to finding y that minimizes
$f(y) = \frac{1}{2}||A(t)y - b(t)||_2^2$
Splitting the objective function on a row by row basis as $f_i(y^{\{i\}}) = \frac{1}{2}||A_i(t)y^{\{i\}} - b_i(t)||_2^2$
- Visualization of $b(t) = A(t)y(t)$ and data splitting:



- Preserving the above splitting structure the update $A(t+1) = A(t) - \mu A(t)F[y(t)]$ can be carried out in parallel on a row by row basis.

Graphical models and consensus achievements



Distributed processing

1. Distributed processing of source estimation:
   - Fusion-center approach

$$\min_{y} \sum_{i=1}^{N} f_i(y^{\{i\}}) \;\; \text{s.t.} \;\; y^{\{i\}} - z = 0 \quad (6)$$

ADMM solution:

$$y^{\{i,k+1\}} = \arg\min_{y^{\{i\}}} (f_i(y^{\{i\}}) + \frac{\rho}{2}||y^{\{i\}} - z^k + u_i^k||_2^2)$$
$$z^{k+1} = \frac{1}{N}\sum_{i=1}^{N}(y^{\{i,k+1\}} + u_i^k)$$
$$u_i^{k+1} = u_i^k + y^{\{i,k+1\}} - z^{k+1} \quad (7)$$

   - De-centralized approach
Over a graph $G = (\nu, \varepsilon)$

$$\min_{y} \sum_{i\in\nu} f_i(y^{\{i\}}) \;\; \text{s.t.} \;\; y^{\{i\}} = y^{\{j\}} \; \forall (i,j) \in \varepsilon \quad (8)$$

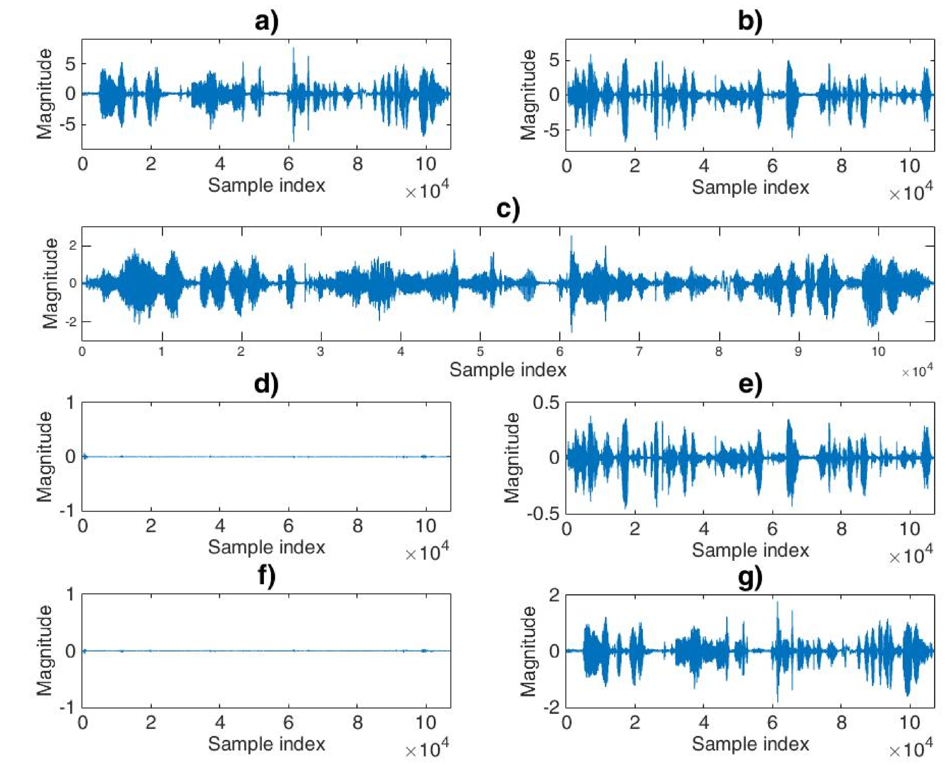
Bi-ADMM solution:

$$y^{\{i,k+1\}} = \arg\min_{y^{\{i\}}} [f_i(y^{\{i\}}) - {y^{\{i\}}}^T$$
$$(\sum_{j\in N(i)} \mathrm{sign}(j-i)\lambda_{j|i}^k) + \sum_{j\in N(i)} \frac{1}{2}||y^{\{i\}} - y^{\{j,k\}}||_2^2]$$
$$\lambda_{i|j}^{k+1} = \lambda_{j|i}^k + \gamma\,\mathrm{sign}(j-i)(y^{\{j,k\}} - y^{\{i,k+1\}}) \quad (9)$$

2. Distributed processing of the parameter update:
Both fusion-center and de-centralized approach update the parameters locally as:

$$A_i(t+1) = A_i(t) - \mu A_i(t) F[y^{\{i,k\}}(t)] \quad (10)$$

Speech Signals Separation



Power consumption

MSE	1×10^{-4}	4×10^{-6}	3×10^{-7}	1×10^{-8}
De-centralized	375ϵ	675ϵ	1025ϵ	1825ϵ
Fusion-center	1296ϵ	2376ϵ	3240ϵ	5400ϵ

SINR across noise levels

	Fusion-center		De-centralized		Centralized	
σ_n^2	S1	S2	S1	S2	S1	S2
0	59.39	21.73	49.80	21.60	49.98	21.71
2×10^{-3}	22.65	15.21	22.63	15.20	22.59	14.94
4×10^{-3}	19.79	11.77	19.79	11.76	19.82	12.02
6×10^{-3}	18.08	9.84	18.09	10.01	18.04	10.31
8×10^{-3}	16.83	8.39	16.84	8.51	16.82	9.16

Conclusions

- Two distributed adaptive linear BSS algorithms were proposed
- They benefit from fully shared computation
- The de-centralized algorithm:
  - can be implemented over any graph
  - is scalable
  - requires low transmission power