Decentralized Sparsity Pattern Recovery using 1-bit Compressed Sensing

Swatantra Kafle, Bhavya Kailkhura, Thakshila Wimalajeewa, Pramod K. Varshney

Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244.

Supported by the National Science Foundation (NSF) under Grant No. ECCS-1307775

GlobalSIP, 2016

Outline

- Motivation
- Problem Formulation
- Binary Iterative Hard Thresholding (BIHT) Algorithm
- Centralized BIHT (C-BIHT) Algorithm
- Decentralized Algorithms
- Conclusion

Motivation

- Compressed Sensing (CS) allows us to reconstruct high dimensional sparse signals from low dimensional measurements
- CS is promising in resource constrained communication networks
- Signal reconstruction is not required in many applications
- Spectrum Sensing in Cognitive radios
  - Assumption: Only a few PUs are present
  - Usage of frequency slots: sparse vector
  - Multiple SUs try to detect the presence or absence of PUs
  - Decide the zero and non-zero locations of the sparse vector
- Sparsity Pattern Recovery Problem (SPRP)

- Questions:
  - How to approach the problem if the network has additional restrictions on the resources?
  - Can we further compress the compressed measurements? (1-bit quantization)
  - How does the performance of decentralized algorithms compare to centralized algorithms?
  - Can performance be improved with collaboration?
- **Goal**: Decentralized Algorithms for SPRP using 1-bit CS

Problem Formulation

- Consider a distributed network with P nodes
- At node p, for $p = 1, \cdots, P$,

$$\mathbf{y}_p = \mathbf{\Phi}_p(\mathbf{s}_p + \mathbf{n}_p).$$

  where $\mathbf{\Phi}_p \in \mathbb{R}^{M \times N}, \boldsymbol{s}_p \in \mathbb{R}^N, \boldsymbol{n}_p \in \mathbb{R}^N$ is i.i.d. Gaussian noise with covariance matrix $\sigma_n^2\ \mathbf{I}_N$
- $\boldsymbol{s}_p$ is K-sparse and is assumed to have same sparse support for $p = 1, \cdots, P$ with possibly different signal amplitudes
- Element wise quantization of $\mathbf{y}_p$

$$q_{ip} = \text{sign}(y_{ip}) = \begin{cases} -1, & \text{if } -\infty < y_{ip} < 0 \\ 1, & \text{if } 0 \le y_{ip} < \infty. \end{cases}$$

- The matrix of quantized measurement

$$\mathbf{Q} = [\mathbf{q}_1 | \mathbf{q}_2 | ... | \mathbf{q}_P]$$

Binary Iterative Hard Thresholding (BIHT) [1]

- BIHT reconstructs the signal from the element-wise quantized vector
- BIHT aims to decrease the cost function

$$\mathcal{J}(s) = \|[\mathbf{q} \odot (\boldsymbol{\Phi}\mathbf{s})]_-\|_1$$

where $[\cdot]_-$ denote negative function, i.e., $[z]_- = z$ if $z < 0$ and 0 else.

- k-th iteration of BIHT algorithm

$$\mathbf{s}^k = \boldsymbol{\Theta}_{\mathbf{K}}\left(\mathbf{s}^{k-1} - \tau \underbrace{\boldsymbol{\Phi}^T(\text{sign}(\boldsymbol{\Phi}\mathbf{s}^{k-1}) - \mathbf{q})}_{\in \text{ sub-differential of } \|[\mathbf{q}\odot(\boldsymbol{\Phi}\mathbf{s})]_-\|_1}\right)$$

where $\boldsymbol{\Theta}_{\mathbf{K}}$ is the hard-thresholding operator

- Susceptible to noise
- Motivates the use of more measurement vectors

[1] "Robust 1-Bit Compressive Sensing via Binary Stable Embeddings of Sparse Vectors", L. Jacques et al., IEEE trans. Inf. Theory, 2013

Centralized-BIHT (C-BIHT)

- All the measurement vectors are available at the fusion center (FC)

$$\mathbf{Y} = \mathbf{\Phi}(\mathbf{S} + \mathbf{N}), \qquad \mathbf{Q} = \text{sign}(\mathbf{Y}).$$

where $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_P]$ and $\mathbf{N} = [\mathbf{n}_1, \cdots, \mathbf{n}_P]$

- Decision on sparsity pattern is made at the FC
- Algorithm

Algorithm 1 Centralized Binary- IHT algorithm (C-BIHT)

Inputs : $\mathbf{\Phi}, K, \mathbf{Q}, \tau$

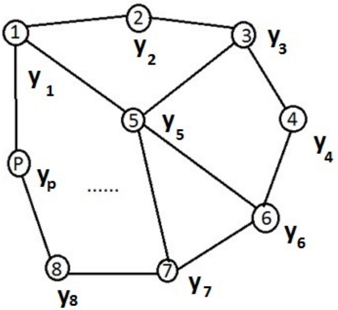
1. Initialize $\mathbf{S}^0$
2. For iteration j until the stopping criteria
3. $\mathbf{S}^j = \mathbf{S}^{j-1} + \tau \mathbf{\Phi}^T (Q - \text{sign}(\mathbf{\Phi}\mathbf{S}^{j-1}))$
4. $\mathbf{T}^j$ = DetectSupport($\mathbf{S}^j, K$)
5. $\mathbf{S}^j$ = Threshold ($\mathbf{S}^j, \mathbf{T}^j$)
6. End For
7. $\hat{\mathbf{S}} = \mathbf{S}^{j^*}$ and $\hat{\mathbf{T}} = \mathbf{T}^{j^*}$ when stopping at iteration j^*

- Vulnerable to the failure of FC.

Decentralized Algorithms

- Avoids the use of FC
- Measurements are sent to and received from one hop neighbors only
- Embed collaboration and fusion among nodes
- Algorithms can be structured involving two distinct stages
  - Information Fusion
  - Index Fusion

Decentralized Algorithms (contd...)



- Network topology is represented as an undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$
- $neigh(i) = \{j \mid (i, j) \in \mathbf{E}\}$ the set of neighboring nodes of node i
- Node p has access to the measurement matrix $\mathbf{Q}_p = [\mathbf{Q}_{neigh(p)},\ \mathbf{q}_p]$
- Let $\hat{\mathbf{T}}_p$ be local support estimate at node p

Decentralized BIHT 1 (D-BIHT 1)

Algorithm 2 Decentralized BIHT 1 (D-BIHT 1)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau\mathbf{\Phi}^T(\mathbf{Q}_p - \text{sign}(\mathbf{\Phi}\mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j$ = DetectSupport($\mathbf{S}_p^j, K$)
6. $\mathbf{S}_p^j$ = Threshold ($\mathbf{S}_p^j, \mathbf{T}_p^j$)
7. End For
8. $\hat{\mathbf{S}_p} = \mathbf{S}^{j^*}$ and $\hat{\mathbf{T}}_p = \mathbf{T}^{j^*}$ when stopping at iteration j^*
9. Global Communication
   For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
10. $\hat{\mathbf{T}}$ = Majority($\hat{\mathbf{T}}_1$,$\hat{\mathbf{T}}_2, .., \hat{\mathbf{T}}_P$)

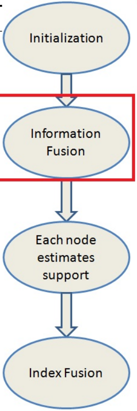
D-BIHT 1

Information Fusion

Algorithm 2 Decentralized BIHT 1 (D-BIHT 1)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   - Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   - Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau \mathbf{\Phi}^T(\mathbf{Q}_p - \text{sign}(\mathbf{\Phi}\mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j$ = DetectSupport($\mathbf{S}_p^j, K$)
6. $\mathbf{S}_p^j$ = Threshold ($\mathbf{S}_p^j, \mathbf{T}_p^j$)
7. End For
8. $\hat{\mathbf{S}}_p = \mathbf{S}^{j^*}$ and $\hat{\mathbf{T}}_p = \mathbf{T}^{j^*}$ when stopping at iteration j^*
9. Global Communication
   - For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   - Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
10. $\hat{\mathbf{T}}$ = Majority($\hat{\mathbf{T}}_1, \hat{\mathbf{T}}_2, .., \hat{\mathbf{T}}_P$)



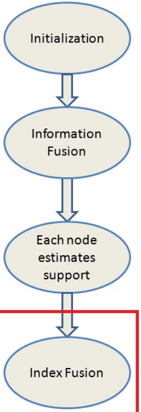
D-BIHT 1

Index Fusion

Algorithm 2 Decentralized BIHT 1 (D-BIHT 1)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau \mathbf{\Phi}^T (\mathbf{Q}_p - \text{sign}(\mathbf{\Phi} \mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j$ = DetectSupport($\mathbf{S}_p^j, K$)
6. $\mathbf{S}_p^j$ = Threshold ($\mathbf{S}_p^j, \mathbf{T}_p^j$)
7. End For
8. $\hat{\mathbf{S}}_p = \mathbf{S}^{j^*}$ and $\hat{\mathbf{T}}_p = \mathbf{T}^{j^*}$ when stopping at iteration j^*
9. Global Communication
   For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
10. $\hat{\mathbf{T}}$ = Majority($\hat{\mathbf{T}}_1, \hat{\mathbf{T}}_2, .., \hat{\mathbf{T}}_P$)



Modified Decentralized BIHT (D-BIHTm)

- Modified form of D-BIHT 1 algorithm
- For networks with severe restrictions on bandwidth usage
- Information Fusion stage is omitted
- Each node makes decisions based on its own measurements (Self Decision Stage)
- Followed by Index fusion stage

Decentralized BIHT 2 (D-BIHT 2)

Algorithm 3 Decentralized BIHT: Algorithm 2 (D-BIHT 2)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau \mathbf{\Phi}^T(\mathbf{Q}_p - \text{sign}(\mathbf{\Phi}\mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j$ = DetectSupport($\mathbf{S}_p^j, K$)
6. Global Communication
   For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
7. $\hat{\mathbf{T}}^j$ = Majority($\hat{\mathbf{T}}_1^j, \hat{\mathbf{T}}_2^j, .., \hat{\mathbf{T}}_P^j$)
8. $\mathbf{S}_p^j$ = Threshold ($\mathbf{S}_p^j, \mathbf{T}^j$) , for all $p \in \mathbf{V}$
9. EndFor
10. $\hat{\mathbf{T}} = \hat{\mathbf{T}}^{j^*}$ when stopping at iteration j^*

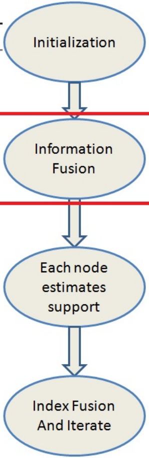
D-BIHT 2

Information Fusion

Algorithm 3 Decentralized BIHT: Algorithm 2 (D-BIHT 2)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau \mathbf{\Phi}^T(\mathbf{Q}_p - \text{sign}(\mathbf{\Phi}\mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j = \text{DetectSupport}(\mathbf{S}_p^j, K)$
6. Global Communication
   For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
7. $\hat{\mathbf{T}}^j = \text{Majority}(\hat{\mathbf{T}}_1^j, \hat{\mathbf{T}}_2^j, .., \hat{\mathbf{T}}_P^j)$
8. $\mathbf{S}_p^j = \text{Threshold}\ (\mathbf{S}_p^j, \mathbf{T}^j)$, for all $p \in \mathbf{V}$
9. EndFor
10. $\hat{\mathbf{T}} = \hat{\mathbf{T}}^{j^*}$ when stopping at iteration j^*



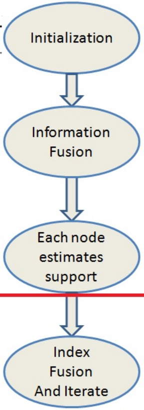
D-BIHT 2

Index Fusion

Algorithm 3 Decentralized BIHT: Algorithm 2 (D-BIHT 2)

Inputs : $\mathbf{\Phi}, K, \tau, neigh(p)$ for all $p \in \mathbf{V}$

1. Initialize $\mathbf{S}_p^0$ for all $p \in \mathbf{V}$
2. Local Communication at node p , for all $p \in \mathbf{V}$
   Transmit $\mathbf{q}_p$ to its one-hop neighborhood
   Receive $\mathbf{q}_r$ where $r \in neigh(p)$ and form $\mathbf{Q}_p$
3. For iteration j until the stopping criteria
4. $\mathbf{S}_p^j = \mathbf{S}_p^{j-1} + \tau\mathbf{\Phi}^T(\mathbf{Q}_p - \text{sign}(\mathbf{\Phi}\mathbf{S}_p^{j-1}))$
5. $\mathbf{T}_p^j$ = DetectSupport($\mathbf{S}_p^j, K$)
6. Global Communication
   For all $p \in \mathbf{V}$, transmit $\hat{\mathbf{T}}_p$ to $\mathbf{V}$
   Receive $\hat{\mathbf{T}}_i$ for $\forall i \neq p$
7. $\hat{\mathbf{T}}^j$ = Majority($\hat{\mathbf{T}}_1^j, \hat{\mathbf{T}}_2^j, .., \hat{\mathbf{T}}_P^j$)
8. $\mathbf{S}_p^j$ = Threshold ($\mathbf{S}_p^j, \mathbf{T}^j$) , for all $p \in \mathbf{V}$
9. EndFor
10. $\hat{\mathbf{T}} = \hat{\mathbf{T}}^{j^*}$ when stopping at iteration j^*



Simulation Results

- Performance Metrics:

$$\text{Percentage of Sparsity Pattern Recovery} = \frac{\#\ \text{Correct Support Recovered}}{\#\ \text{Correct Support}}$$

$$\text{Probability of Exact Sparsity Pattern Recovery(PESPR)} = \frac{\#\ \text{all the support estimated}}{\#\ \text{of run of experiments}}$$

Simulation Results (Contd...)

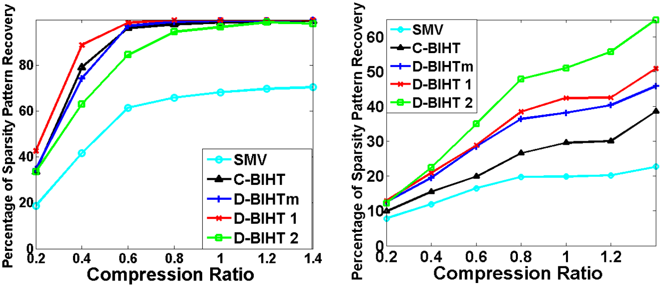
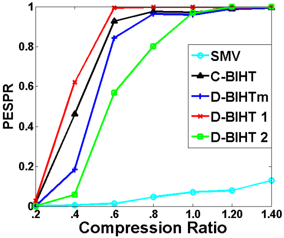


Figure 1: Percentage of Sparsity Pattern Recovery for D-BIHT 1, D-BIHT 2, D-BIHTm, C-BIHT, BIHT (SMV) algorithms in a network of Node degree 3 for high SNR (noise variance =0.0004) and low SNR (noise variance = .0625) respectively.

Simulation Results (Contd...)



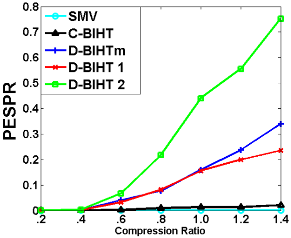


Figure 2: Probability of Exact Sparsity Pattern Recovery (PESPR) for D-BIHT 1, D-BIHT 2, D-BIHTm, C-BIHT, BIHT (SMV) algorithms in a network of Node degree 6 for high SNR (noise variance =.002025) and low SNR (noise variance =.01) respectively.

Simulation Results (Contd...)

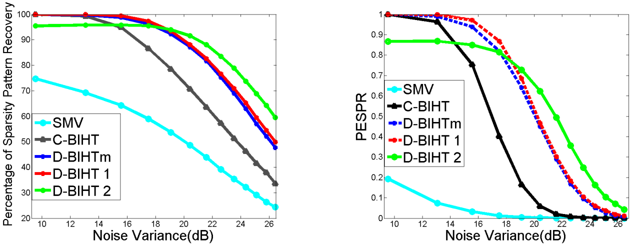


Figure 3: Percentage of Sparsity Pattern Recovery and Probability of Exact Sparsity Pattern Recovery (PESPR) for D-BIHT 1, D-BIHT 2, D-BIHTm, C-BIHT, BIHT (SMV) in a network of Node degree 4 for M = 40.

Conclusion

- Proposed two decentralized algorithms using 1-bit CS
- Strategic collaboration and fusion can lead to improved results
- D-BIHT 1 performs better in the high SNR regime
- D-BIHT 2 performs better in the low SNR regime
- **Future work**: Theoretical results on performance bounds and guarantees

Thank you