Distributed learning of joint sparse signals in wireless sensor networks

Saurabh Khanna

SPC Lab, ECE, IISc

Contents

- Problem Statement
  - Estimation of joint sparse signals in WSNs
- Centralized scheme
  - Centralized joint sparse signal recovery algorithm
  - Some performance results
- Distributed scheme
  - Distributed joint sparse signal recovery algorithm
  - Performance comparison between centralized and distributed schemes
  - Convergence analysis of distributed scheme
  - Optimization of convergence rate
- Future work

Problem Statement

- Consider a WSN consisting of L sensor nodes $(s_0, s_1 \ldots s_L)$.
- Sensor node s_j wants to estimate signal vector $\mathbf{x}_j \in \mathbb{R}^n$.
  - $\mathbf{x}_j$ are sparse.
  - $\mathbf{x}_j$ share common support (joint sparsity model-2 [1])
  - For $j \neq k$, non zero entries of $\mathbf{x}_j$ & $\mathbf{x}_k$ are uncorrelated.
- Sensor node $s_j, j \in (1, 2 \ldots L)$ takes m noisy linear measurements of signal vector of interest $\mathbf{x}_j$.

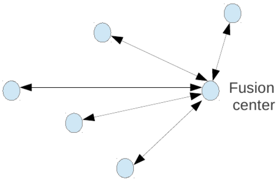
$$\mathbf{y}_j = \Phi_j \mathbf{x}_j + \mathbf{w}_j$$

  - $\Phi_j \in \mathbb{R}^{m \times n}$
  - $\mathbf{w}_j \sim N(0, \sigma_j^2 I)$
- Sensor node $s_j, j \in (1, 2 \ldots L)$ takes m noisy linear measurements of signal vector of interest $\mathbf{x}_j$.
- Goal: Estimate joint sparse signal vectors $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$ from $\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L$.

---

[1] Distributed Compressed Sensing, Duarte, Sarvotham, Baron, Wakin & Baranuik, 2005

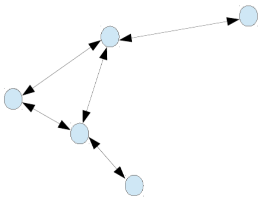
Centralized and distributed schemes in WSNs



Centralized scheme

- Each sensor node s_j transmits its local measurement vector $\mathbf{y}_j$ and measurement matrix Φ_j to fusion center (FC).
- FC runs joint sparse signal recovery algorithm to estimate $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$.
- FC transmits recovered sparse vector $\mathbf{x}_j$ to sensor node s_j.
- Advantages:
  - Sensor node design is simplified, computationally intensive recovery algorithm offloaded to FC
  - Number of messages exchanged is low.
- Disadvantages:
  - If FC breaks down, WSN collapses.
  - Less sensing range.

Centralized and distributed schemes in WSNs



Distributed scheme

- Each sensor node s_j can perform computations needed for recovering $\mathbf{x}_j$ from local measurement $\mathbf{y}_j$ and then some more..
- *Question:* Can the WSN converge to centralized solution while ensuring:
  - processing at each sensor node is kept as simple as possible.
  - sensor nodes exchange messages with only single-hop neighbours.
  - no exchange of $\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L$ and local estimates $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots \hat{\mathbf{x}}_L$.

Centralized algorithm

- Goal:
  Estimate joint sparse vectors $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$ from measurements across the network $\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L$.
- Measurement model:

$$\mathbf{Y} = \Phi \mathbf{X} + \mathbf{W}$$

  - $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L]$, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L]$, $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2 \ldots \mathbf{w}_L]$
- We assume $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$ are i.i.d and $N(0, \Gamma^{-1})$, where $\Gamma = diag(\gamma), \gamma \in \mathbb{R}_+^n$.
- If $\gamma(i) = \infty$, then $\mathbf{x}_1(i) = \mathbf{x}_1(i) \cdots = \mathbf{x}_L(i) = 0$.
- For fixed Γ, then LMMSE estimate of $\hat{\mathbf{x}}_j \sim N(\boldsymbol{\mu}_j, \Sigma_j)$.

$$\begin{aligned} \Sigma_j &= (\Gamma + \frac{\Phi^T \Phi}{\sigma_j^2})^{-1} \\ \mu_j &= \sigma_j^{-2} \Sigma_j \Phi^T \mathbf{y}_j \end{aligned}$$

- If we find γ, we can find $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$.

Centralized algorithm

- We find ML estimate of γ.

$$\gamma^* = \underset{\gamma}{argmax} \;\; p(\mathbf{Y}/\gamma)$$

$$\begin{aligned}
p(\mathbf{Y}/\gamma) &= \int_{\mathbf{x}_1,\mathbf{x}_2\ldots\mathbf{x}_L} p(\mathbf{y}_1,\mathbf{y}_2\ldots\mathbf{y}_L,\mathbf{x}_1,\mathbf{x}_2\ldots\mathbf{x}_L/\gamma) \\
&= \int_{\mathbf{x}_1,\mathbf{x}_2\ldots\mathbf{x}_L} \prod_{j=1}^{L} p(\mathbf{y}_j,\mathbf{x}_j/\gamma) \\
&= \prod_{j=1}^{L} \int_{\mathbf{x}_j} p(\mathbf{y}_j/\mathbf{x}_j)\cdot p(\mathbf{x}_j/\gamma) \\
&= \prod_{j=1}^{L} N(\mathbf{y}_j \; ; \; 0 \; , \; (\sigma_j^2 I + \Phi_j \Gamma^{-1} \Phi_j^T))
\end{aligned}$$

- No closed form expression for γ^* exists.
- How do we find γ^* ?
  - METHOD 1: Find γ^* using fixed point iterations (iterative re-estimation)
  - METHOD 2: Use EM algorithm to maximize $p(\mathbf{Y}/\gamma)$ by treating $\mathbf{x}_1,\mathbf{x}_2\ldots\mathbf{x}_L$ as hidden variables.

Centralized algorithm

- EM formulation for ML estimation of γ
  - Observed variables: $\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L$
  - Hidden variables: $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$
  - Complete data: $\{\mathbf{Y}, \mathbf{X}\}$
  - To be estimated: γ
  - E-Step:

$$Q(\gamma, \gamma^{(k)}) = E_{[X/Y, \gamma^{(k)}]} \log p(Y, X/\gamma)$$

  - M-Step:

$$\gamma^{(k+1)} = \underset{\gamma}{argmax} \;\; Q(\gamma, \gamma^{(k)})$$

Centralized algorithm

- Simplification of E-step cost function $Q(\gamma, \gamma^{(k)})$

$$
\begin{aligned}
Q(\gamma, \gamma^{(k)}) &= E_{[X/Y,\gamma^{(k)}]} \log p(Y, X/\gamma) \\
&= E_{[X/Y,\gamma^{(k)}]} [\log p(Y/X) + \log p(X/\gamma)] \\
&= E_{[X/Y,\gamma^{(k)}]} [\log p(X/\gamma)] \qquad (\textit{first term independent of } \gamma) \\
&= E_{[X/Y,\gamma^{(k)}]} \sum_{j=1}^{L} \log p(\mathbf{x}_j/\gamma) \\
&= \sum_{j=1}^{L} E_{\mathbf{x}_j/\mathbf{y}_j,\gamma^{(k)}]} \log p(\mathbf{x}_j/\gamma) \\
&= \sum_{j=1}^{L} E_{[\mathbf{x}_j/\mathbf{y}_j,\gamma^{(k)}]} \left[\frac{-n}{2} \log 2\pi - \frac{1}{2} \log |\Gamma^{-1}| - \frac{1}{2} \mathbf{x}_j^T \Gamma \mathbf{x}_j\right] \\
&= \frac{L}{2} \log |\Gamma| - \frac{1}{2} \sum_{j=1}^{L} E_{[\mathbf{x}_j/\mathbf{y}_j,\gamma^{(k)}]} \mathbf{x}_j^T \Gamma \mathbf{x}_j \\
&= \frac{L}{2} \sum_{i=1}^{n} \log \gamma(i) - \frac{1}{2} \sum_{j=1}^{L} \sum_{i=1}^{n} \gamma(i) (E_{[\mathbf{x}_j(i)/\mathbf{y}_j(i),\gamma^{(k)}]} \mathbf{x}_j^2) \\
&= \frac{L}{2} \sum_{i=1}^{n} \log \gamma(i) - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{L} \gamma(i) (\Sigma_j^{(k)}(i,i) + \boldsymbol{\mu}_j^{(k)}(i)^2)
\end{aligned}
$$

Centralized algorithm

- M-step

$$\gamma^{(k+1)} = \underset{\gamma}{argmax} \;\; Q(\gamma, \gamma^{(k)})$$

$$\gamma^{(k+1)}(i) = \frac{L}{\sum_{j=1}^{L} \Sigma_j^{(k)}(i,i) + \mu_j^{(k)}(i)^2} \qquad \forall \;\; i = 1 \; to \; n$$

Centralized algorithm

- EM-iteration in centralized algorithm
- E-Step:

$$\Sigma_j^{(k)} = (\Gamma^{(k)} + \frac{\Phi_j^T \Phi_j}{\sigma_j^2})^{-1}$$
$$\mu_j^{(k)} = \sigma_j^{-2} \Sigma_j^{(k)} \Phi_j^T \mathbf{y}_j$$

- M-Step:

$$\gamma^{(k+1)}(i) = \frac{L}{\sum_{j=1}^{L} \Sigma_j^{(k)}(i,i) + \mu_j^{(k)}(i)^2} \qquad \forall \;\; i = 1 \textit{ to } n$$

- Upon convergence, it is observed that most of $\gamma(i) \to \infty$, leading to sparse LMMSE estimates $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_1 \ldots \hat{\mathbf{x}}_L$.

Centralized algorithm

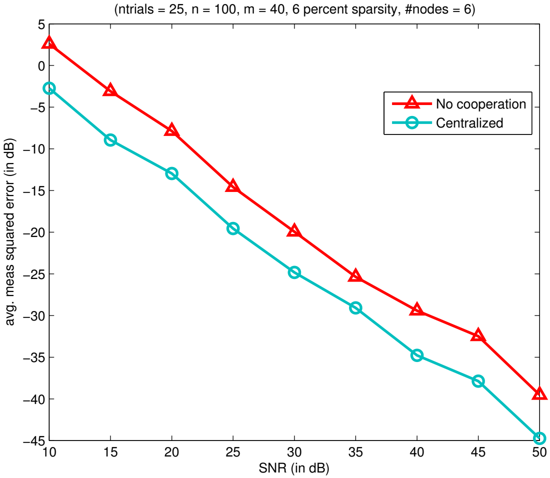


Figure: Average MSE vs SNR

Centralized algorithm

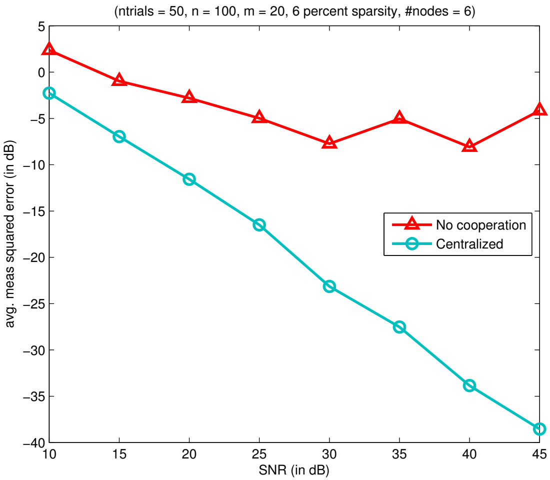


Figure: Average MSE vs SNR for highly undersampled case

Distributed scheme

- Rewrite the objective function $Q(\gamma, \gamma^{(k)})$ in M-step.

$$\begin{aligned}\gamma^{(k+1)} &= \underset{\gamma}{argmax}\ Q(\gamma, \gamma^{(k)}) \\ &= \underset{\gamma}{argmax}\ \frac{1}{2}\sum_{j=1}^{L}\sum_{i=1}^{n}\log\gamma(i) - \gamma(i)(\Sigma_j^{(k)}(i,i) + \boldsymbol{\mu}_j^{(k)}(i)^2)\end{aligned}$$

- $Q(\gamma, \gamma^{(k)})$ is separable in $\mathbf{y}_1, \mathbf{y}_2 \ldots \mathbf{y}_L$!
- $Q(\gamma, \gamma^{(k)})$ can be split as

$$Q(\gamma, \gamma^{(k)}) = \sum_{j=1}^{L} f_j(\gamma, \mathbf{y}_j)$$

$$f_j(\gamma, \mathbf{y}_j) = \sum_{i=1}^{n}\log\gamma(i) - \gamma(i)(\Sigma_j^{(k)}(i,i) + \boldsymbol{\mu}_j^{(k)}(i)^2)$$

Distributed scheme

- We re-write the optimization problem in M-step as

$$\min_{\gamma} \sum_{j=1}^{L} \sum_{i=1}^{n} -\log \gamma_j(i) + \gamma_j(i)(\Sigma_j^{(k)}(i,i) + \mu_j^{(k)}(i)^2)$$

$$\textit{such that } \gamma_j = \gamma_b \quad \forall\, b \in B_j \ , \quad j \in J$$

  - γ_b are auxilliary parameters used to establish consensus among $\gamma_0, \gamma_1 \cdots \gamma_L$.
  - γ_b is maintained by a bridge node s_b.
  - B_j denotes the set of bridge nodes connected to s_j.
  - $B = \bigcup_{j \in J} B_j$ is the set of all bridge nodes in WSN.
  - N_b is the set of sensor node s_j connected to bridge node s_b.

Distributed scheme

- Selection of bridge nodes in WSN

  Rule-1: Each node in the network must be connected to atleast one bridge node in B, i.e., $B_j \neq \phi$.

  Rule-2: If two nodes s_{j_1} and s_{j_2} are single hop connected neighbours, then $B_{j_1} \bigcap B_{j_2} \neq \phi$.

- For a connected WSN, if conditions (1) and (2) hold then $\gamma_j = \gamma_b \quad \forall\, b \in B_j \,, \quad j \in J$ implies that all γ_j are equal.

Alternating directions method of multipliers

- ADMM problem form (with f, g convex)

$$\begin{aligned} \textit{minimize} \quad & f(\mathbf{x}) + g(\mathbf{z}) \\ \textit{subject to} \quad & \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c} \end{aligned}$$

- Augmented Lagrangian

$$L_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda}) = f(\mathbf{x}) + g(\mathbf{z}) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}) + \frac{\rho}{2}||\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}||_2^2$$

- ADMM iterations

$$\begin{aligned} \mathbf{x}^{k+1} &= \underset{\mathbf{x}}{\textit{argmin}}\ L_\rho(\mathbf{x}, \mathbf{z}^k, \boldsymbol{\lambda}^k) && \text{(x-minimization)} \\ \mathbf{z}^{k+1} &= \underset{\mathbf{z}}{\textit{argmin}}\ L_\rho(\mathbf{x}^{k+1}, \mathbf{z}, \boldsymbol{\lambda}^k) && \text{(z-minimization)} \\ \boldsymbol{\lambda}^{k+1} &= \boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}) && \text{(dual update)} \end{aligned}$$

ADMM for distributed scheme

- Augmented Lagrangian $L_\rho(\gamma_{j\in J}, \gamma_{b\in B}, \boldsymbol{\lambda}_{j\in J})$.

$$\sum_{j=1}^{L}\sum_{i=1}^{n} - \log \gamma_j(i) + \gamma_j(i)(\Sigma_j^{(k)}(i,i) + (\mu_j^{(k)}(i))^2)$$
$$+ \quad \sum_{j=1}^{L}\sum_{b\in B_j} (\boldsymbol{\lambda}_j^b)^T(\gamma_{\boldsymbol{j}} - \gamma_{\boldsymbol{b}})$$
$$+ \quad \frac{\rho}{2}\sum_{j=1}^{L}\sum_{b\in B_j} ||\gamma_{\boldsymbol{j}} - \gamma_{\boldsymbol{b}}||_2^2$$

- ADMM iterations

$$(\gamma_{j\in J})^{k+1} = \underset{\gamma_{j\in J}}{argmin}\, L_\rho(\gamma_{j\in J}, \gamma_{b\in B}^k, \boldsymbol{\lambda}_{j\in J}^k)$$
$$(\gamma_{b\in B})^{k+1} = \underset{\gamma_{b\in B}}{argmin}\, L_\rho(\gamma_{j\in J}^{k+1}, \gamma_{b\in B}, \boldsymbol{\lambda}_{j\in J}^k)$$
$$(\boldsymbol{\lambda}_j^b)^{k+1} = (\boldsymbol{\lambda}_j^b)^k + \rho(\gamma_{\boldsymbol{j}}^{k+1} - \gamma_{\boldsymbol{b}}^{k+1})$$

ADMM for distributed scheme

- ADMM iterations

$$
\begin{aligned}
(\boldsymbol{\lambda}_j^b)^{k+1} &= (\boldsymbol{\lambda}_j^b)^k + \rho(\gamma_{\boldsymbol{j}}^k - \gamma_{\boldsymbol{b}}^k) & \forall j \in J,\ b \in B_j \qquad (1)\\
(\gamma_j)^{k+1} &= \underset{\gamma_{j\in J}}{argmin}\ L_\rho(\gamma_{j\in J}, \gamma_{b\in B}^k, \boldsymbol{\lambda}_{j\in J}^{k+1}) & \forall\ j \in J \qquad (2)\\
(\gamma_b)^{k+1} &= \frac{\sum_{j\in N_b}(\rho\gamma_j^{k+1} + (\boldsymbol{\lambda}_j^b)^{k+1})}{\sum_{j\in N_b}\rho} & \forall b \in B \qquad (3)
\end{aligned}
$$

EM iterations in WSN

- Initialization
  - $\boldsymbol{\lambda}_j^b$ and $\boldsymbol{\gamma_b}$ seeded with zero
  - $\boldsymbol{\gamma_j}$ seeded with random values.
- Each iteration of EM algorithm comprises of:
  - *COMM ROUND1*: Each bridge node $b \in B$ transmits its $\boldsymbol{\gamma}_b$ to all nodes in N_b.
  - Each node $j \in J$ updates its set of lagrangian variables according to

$$(\boldsymbol{\lambda}_j^b)^{k+1} = (\boldsymbol{\lambda}_j^b)^k + \rho(\boldsymbol{\gamma}_{\boldsymbol{j}}^k - \boldsymbol{\gamma}_{\boldsymbol{b}}^k) \qquad \forall\ b \in B_j$$

  - *COMM ROUND2:* Each node $j \in J$ transmits $\boldsymbol{\lambda}_j^b$ to all bridge nodes in B_j
  - Each node $j \in J$ updates its estimate of hyperparameters $\boldsymbol{\gamma_j}$ according to

$$\gamma_j^{k+1}(i) = \frac{\sqrt{P^2 + 4\rho|B_j|} - P}{2\rho|B_j|} \qquad \textit{where } P = \Sigma^k(i,i) + \boldsymbol{\mu}^k(i)^2 + \sum_{b \in B_j}(\boldsymbol{\lambda}_j^{b\,k+1}(i) - \rho\boldsymbol{\gamma}_{\boldsymbol{b}}^k(i))$$

  - *COMM ROUND3:* each node $j \in J$ transmits its $\boldsymbol{\gamma_j}$ to all bridge nodes in B_j.
  - Each node $b \in B$ updates its estimate of bridge parameter $\boldsymbol{\gamma}_b$ according to

$$(\boldsymbol{\gamma}_b)^{k+1} = \frac{\sum_{j \in N_b}(\rho\boldsymbol{\gamma}_j^{k+1} + (\boldsymbol{\lambda}_j^b)^{k+1})}{\sum_{j \in N_b}\rho}$$

Distributed scheme

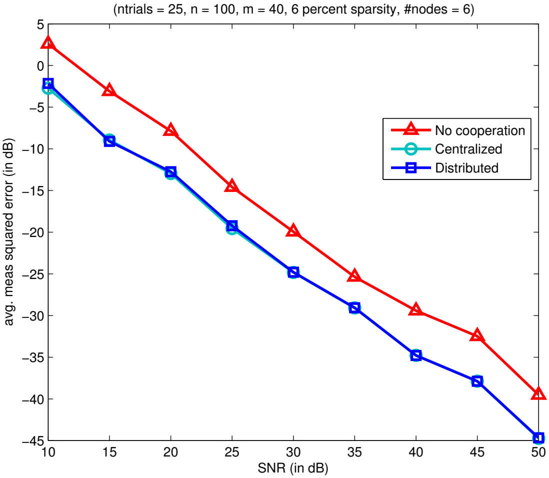


Figure: Average MSE vs SNR

Distributed scheme

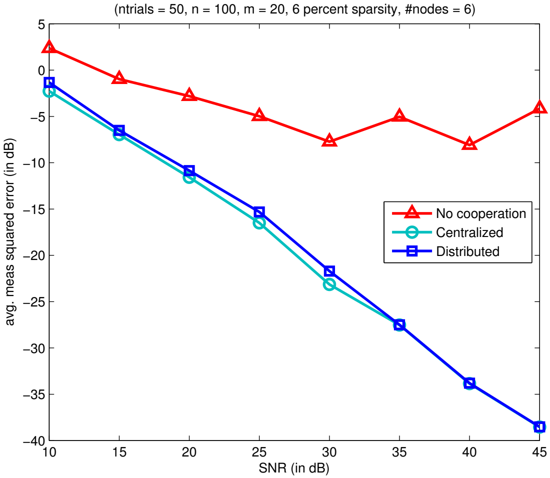


Figure: Average MSE vs SNR for undersampled case

Distributed scheme

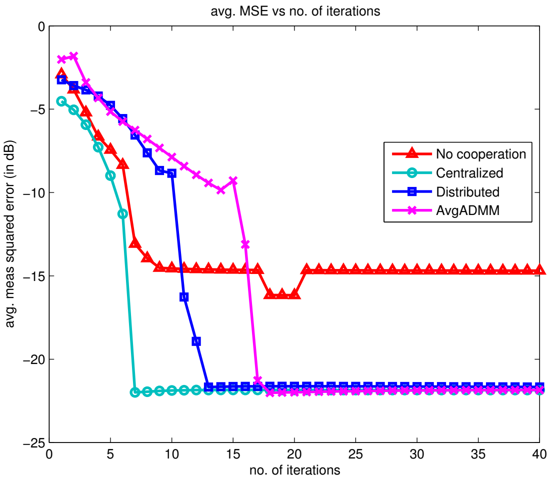


Figure: Comparison of convergence rate at SNR = 20 dB

Convergence rate analysis for ADMM

- Suppose a sequence x_k converges to L.
- x_k is said to be *Q-linearly* convergent to L, if there exists $\mu \in (0, 1)$ such that

$$\lim_{k \to \infty} \frac{|x_{k+1} - L|}{|x_k - L|} = \mu$$

- x_k is said to be *R-linearly* convergent to L, if there exists Q-linearly convergent sequence y_k which converges to zero such that

$$\lim_{k \to \infty} |x_k - L| \leq y_k$$

Main convergence result

- For the convex minimization problem (f is convex),

$$\begin{array}{ll} \textit{minimize} & f(\mathbf{x}) \\ \textit{subject to} & \mathbf{E}_1\mathbf{x} + \mathbf{E}_2\mathbf{z} = 0 \end{array}$$

let $\mathbf{x}^*$ and $\mathbf{z}^*$ denote the unique optimal values which minimize the primal problem. Also let $\boldsymbol{\lambda}^*$ denote the unique maximizer of dual problem with dual variable $\boldsymbol{\lambda}$. If we perform the following iterations:

$$\begin{aligned} \mathbf{x}^{k+1} &= \underset{\mathbf{x}}{\textit{argmin}}\ L_\rho(\mathbf{x}, \mathbf{z}^k, \boldsymbol{\lambda}^k) \\ \mathbf{z}^{k+1} &= \underset{\mathbf{z}}{\textit{argmin}}\ L_\rho(\mathbf{x}^{k+1}, \mathbf{z}, \boldsymbol{\lambda}^k) \\ \boldsymbol{\lambda}^{k+1} &= \boldsymbol{\lambda}^k + \rho(\mathbf{E}_1\mathbf{x} + \mathbf{E}_2\mathbf{z}) \end{aligned}$$

then, we have Q-linear convergence of $\mathbf{u}^k = [\mathbf{E}_2\mathbf{z}^K \quad \boldsymbol{\lambda}^k]^T$ to $\mathbf{u}^* = [\mathbf{E}_2\mathbf{z}^* \quad \boldsymbol{\lambda}^*]^T$ and R-linear convergence of $\mathbf{x}^k$ to $\mathbf{x}^*$.

$$||\mathbf{u}^{k+1} - \mathbf{u}^*||_G^2 \leq \frac{1}{1+\delta}||\mathbf{u}^k - \mathbf{u}^*||_G^2$$

$$||\mathbf{x}^{k+1} - \mathbf{x}^*||^2 \leq \frac{1}{2m_f}||\mathbf{u}^k - \mathbf{u}^*||_G^2$$

Main convergence result

- Where m_f is such that

$$\langle \nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq m_f ||\mathbf{x}_1 - \mathbf{x}_2||_2^2 \qquad \forall\ \mathbf{x}_1, \mathbf{x}_2$$

- $||.||_G$ is a vector norm defined as

$$||\mathbf{w}||_G^2 = \mathbf{w}^T \begin{bmatrix} \rho I & 0 \\ 0 & I \end{bmatrix} \mathbf{w}$$

- $\delta > 0$ and is given by

$$\delta = min\{ \frac{2m_f}{\mu\rho\sigma_{max}^2(\mathbf{E}_1) + \frac{\nu M_f^2}{(\nu-1)\sigma_{min}^2(\mathbf{E}_1)}}, \quad \frac{\sigma_{min}^2(\mathbf{E}_1)}{\rho\nu\sigma_{max}^2(\mathbf{E}_1)}, \quad \frac{(\mu-1)\rho}{\mu} \}$$

  where μ and ν are auxilliary variables greater than one.
- M_f is Lipshitz constant of ∇f.

Optimizing the convergence rate

- For fast convergence, we select ρ which maximizes δ,

$$\delta = min\{\underbrace{\frac{2m_f}{\mu\rho\sigma^2_{max}(\mathbf{E}_1) + \frac{\nu M_f^2}{(\nu-1)\sigma^2_{min}(\mathbf{E}_1)}}}_{\beta_1}, \underbrace{\frac{\sigma^2_{min}(\mathbf{E}_1)}{\rho\nu\sigma^2_{max}(\mathbf{E}_1)}}_{\beta_2}, \underbrace{\frac{(\mu-1)\rho}{\mu}}_{\beta_3}\}$$

- For fixed ρ, δ is maximized when $\beta_1 = \beta_2 = \beta_3$
- Let $\beta_1 = \beta_2 = \beta_3$ hold when $\mu = \mu^*$ and $\nu = \nu^*$.
- Define $\nu_o = \frac{\rho\nu^*\sigma^2_{max}(\mathbf{E}_1)}{\sigma^2_{min}(\mathbf{E}_1)}$, then ν_o must satisfy:

$$(2m_f\rho)\nu_o^2 - (2m_f + 2m_f\rho^2 K + \rho^2 K\sigma^2_{min} + \frac{\rho M_f^2}{\sigma^2_{min}})\nu_o + (2m_f\rho K + \rho^3 K^2\sigma^2_{min} + \frac{M_f^2}{\sigma^2_{min}}) = 0$$

$$K = \frac{\sigma^2_{min}(\mathbf{E}_1)}{\sigma^2_{max}(\mathbf{E}_1)} = \frac{Max\ connections\ per\ node}{Min\ connections\ per\ node}$$

- Since $\nu_o = \frac{1}{\delta}$, we want the larger root of this quadratic equation to be as small.

Optimizing the convergence rate

- Root analysis of quadratic equation in ρ

$$(2m_f\rho)\nu_o^2-(2m_f+2m_f\rho^2K+\rho^2K\sigma_{min}^2+\frac{\rho M_f^2}{\sigma_{min}^2})\nu_o+(2m_f\rho K+\rho^3K^2\sigma_{min}^2+\frac{M_f^2}{\sigma_{min}^2})=0$$

- Both roots are positive.
- In order to minimize larger root (maximize δ), we select ρ which minimizes the sum of roots.

$$\rho_{optimal}=\sqrt{\frac{2m_f}{2m_fK+K\sigma_{min}^2(\mathbf{E}_1)}}$$

Optimizing the convergence rate

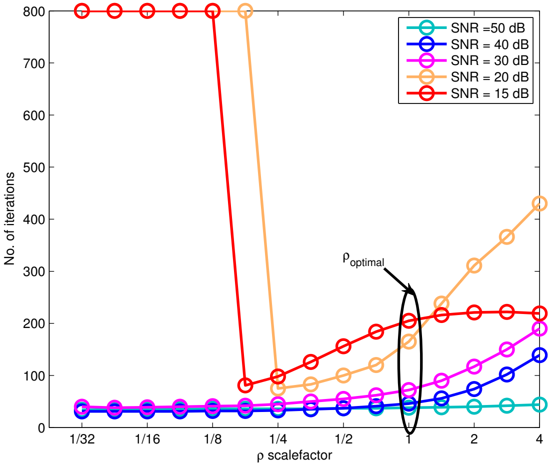


Figure: Optimal ρ selection with respect to no. of iterations

Optimizing the convergence rate

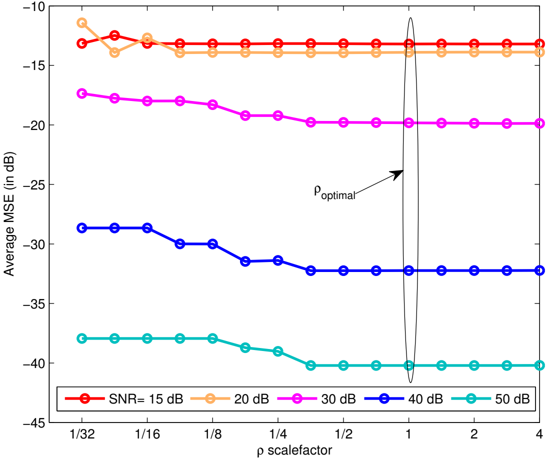


Figure: Optimal ρ selection with respect to average MSE

Future work

- Check the robustness of algorithm under following cases:
  - Messages exchanged between sensor nodes is quantized
  - Different types of connected graphs
  - Non-zero entries of $\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_L$ are distributed according to multi-mode pdfs and other non-Gaussian pdfs.
  - Large variations in SNR at each sensor node
  - Bias in noise variance estimates
- Possible extensions:
  - Exploit inter vector correlation in JSM-2 model
  - Reduction in messages exchanged between sensor nodes
  - Tracking time varying sparse vectors (under JSM-2 paradigm)
  - Study convergence rate for noisy and fading channel/links.

Thank You !!!