Distributed Projection Approximation Subspace Tracking Based on Consensus Propagation

Carolina Reyes, Thibault Hilaire, Christoph F. Mecklenbräuker

creyes@nt.tuwien.ac.at, thibault.hilaire@lip6.fr, cfm@nt.tuwien.ac.at

Institute of Communications and Radio Frequency Engineering

Vienna University of Technology

CAMSAP

December 16, 2009

INSTITUT
FÜR NACHRICHTENTECHNIK
UND HOCHFREQUENZTECHNIK

FORTE:wit
WOMEN IN TECHNOLOGY

Outline

Problem Statement

- Finite capacity of communication channel.
- Bit rate constraints.
- Sensor network architectures are structured in a centralized/small distributed fashion.
- Average data collected from the whole network is more important than individual node data.

Applications

- Industrial, building, home system automation.
- Monitoring (concentrations of chemicals in hydrology, agriculture, pollution control, prediction of avalanches and land slides).
- Healthcare sensor implantation in human bodies.

Projection Approximation Subspace Tracking Algorithm

Mathematical model

Let $\underline{\boldsymbol{x}}(t) \in \mathbb{C}^N$ be the data vector observed at time t, with r narrow-band signal waves impinging on an array of N sensors

$$\underline{\boldsymbol{x}}(t) = \boldsymbol{A}(\underline{\boldsymbol{\omega}}(t))\underline{\boldsymbol{s}}(t)+\underline{\boldsymbol{v}}(t),$$

$$\boldsymbol{A} = \begin{pmatrix} 1 & 1 & 1 \\ e^{j\omega_1} & e^{j\omega_2} & e^{j\omega_r} \\ \vdots & \vdots & \vdots \\ e^{(N-1)j\omega_1} & e^{(n-1)j\omega_2} & e^{(n-1)j\omega_r} \end{pmatrix}, \underline{\boldsymbol{s}}(t) = \begin{pmatrix} s_1(t) \\ \vdots \\ s_r(t) \end{pmatrix}$$

Projection Approximation Subspace Tracking Algorithm

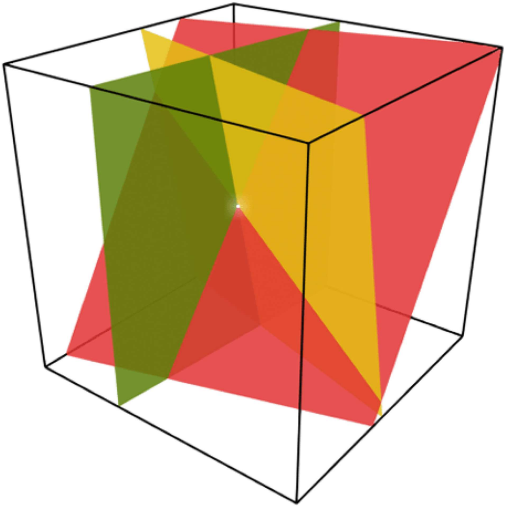


Image source: Euclidean Subspace, Wikipedia

Projection Approximation Subspace Tracking Algorithm

Cost function

Bin Yang [1] proposed to minimize the following cost function

$$J'(\boldsymbol{W}(t)) = \sum_{i=1}^{t} \beta^{t-i} \left\| \underline{\boldsymbol{x}}(i) - \boldsymbol{W}(t)\underline{\boldsymbol{y}}(i) \right\|^2,$$

by the the approximation

$$\underline{\boldsymbol{y}}(i) = \boldsymbol{W}^H(i-1)\underline{\boldsymbol{x}}(i).$$

[1] B. Yang, “Projection Approximation Subspace Tracking”, IEEE Trans. Sig. Proc., vol. 43, no. 1, pp. 95-107, 1995.

What is Projection Approximation Subspace Tracking?

- Requires $O(nr)$ operations per update.
- n: Input vector dimension.
- r: Desired number of eigencomponents.
- t: Number of snapshots.
  Constrained to $r < n < t$.

Consensus Propagation

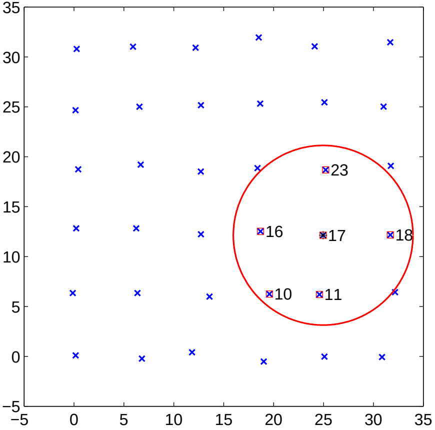


Figure: **Sensor network with neighborhood $\mathcal{N}_{17}$ for radius 9**

Consensus Propagation

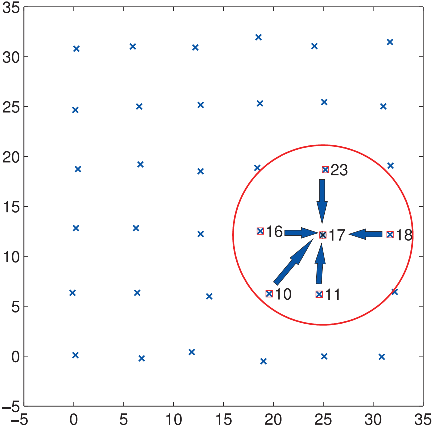


Figure: **Sensor network with neighborhood $\mathcal{N}_{17}$ for radius 9**

Consensus Propagation

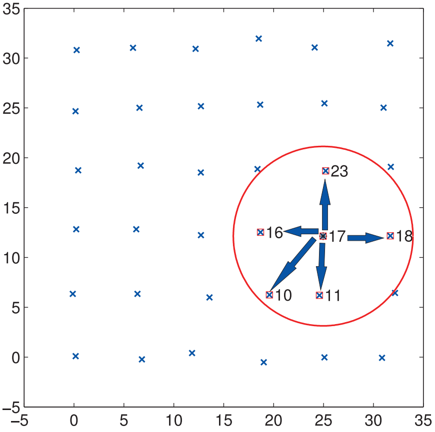


Figure: **Sensor network with neighborhood $\mathcal{N}_{17}$ for radius 9**

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1) w_j \right) / \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

Broadcast the pair $\{\underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$

Output: $\underline{\boldsymbol{y}}_n(t)$ is the estimation of the average in step t at node n

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1) w_j \right) \Big/ \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\mathbf{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\mathbf{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\mathbf{y}}_j(t-1) w_j \right) / \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

Broadcast the pair $\{\underline{\mathbf{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$,

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{y}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{y}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{y}_j(t-1) w_j \right) / \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

Broadcast the pair $\{\underline{y}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$, $w_n = 1/\sqrt{|\mathcal{N}_n|}$

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1) w_j\right) \Big/ \left(\sum_{j \in \mathcal{N}_n} w_j\right)$$

Broadcast the pair $\{\underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$, $w_n = 1 / \sqrt{|\mathcal{N}_n|}$

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1) w_j \right) / \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

Broadcast the pair $\{\underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$, $w_n = 1 / \sqrt{|\mathcal{N}_n|}$

Output: $\underline{\boldsymbol{y}}_n(t)$ is the estimation of the average in step t at node n

endfor

Consensus Propagation Algorithm

for $n := 1, 2, \ldots, N$ do

Input $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j \in \mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j \in \mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1) w_j \right) \Big/ \left(\sum_{j \in \mathcal{N}_n} w_j \right)$$

Broadcast the pair $\{\underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$, $w_n = 1 / \sqrt{|\mathcal{N}_n|}$

Output: $\underline{\boldsymbol{y}}_n(t)$ is the estimation of the average in step t at node n

endfor

PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$
for $t := 1, 2, \ldots$ do
  for $n := 1, 2, \ldots, N$ do
    Input: $x_n(t)$
    aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1)$ from all nodes$\in \mathcal{N}_n$
    $\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$
    locally average $\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{h}}_n(t) = \boldsymbol{P}_n(t-1)\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{g}}_n(t) = \underline{\boldsymbol{h}}_n(t)/[\beta + \underline{\boldsymbol{y}}_n^H(t)\underline{\boldsymbol{h}}_n(t)]$
    $\boldsymbol{P}_n(t) = \frac{1}{\beta}(\boldsymbol{P}_n(t-1) - \underline{\boldsymbol{g}}_n(t)\underline{\boldsymbol{h}}_n^H(t))$
    $\underline{\boldsymbol{e}}_n(t) = \boldsymbol{D}_n(\underline{\boldsymbol{x}}(t) - \boldsymbol{W}_n(t-1)\underline{\boldsymbol{y}}_n(t))$
    $\boldsymbol{W}_n(t) = \boldsymbol{W}_n(t-1) + \underline{\boldsymbol{e}}_n(t)\underline{\boldsymbol{g}}_n^H(t)$
    broadcast $\{x_n(t), \underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes$\in \mathcal{N}_n$
  end
end

PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$
for $t := 1, 2, \ldots$ do
for $n := 1, 2, \ldots, N$ do
Input: $x_n(t)$
end
end

PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$

for $t := 1, 2, \ldots$ do

for $n := 1, 2, \ldots, N$ do

Input: $x_n(t)$

aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1)$ from all nodes $\in \mathcal{N}_n$

$\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$

locally average $\underline{\boldsymbol{y}}_n(t)$

end

end

PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$
for $t := 1, 2, \ldots$ do
  for $n := 1, 2, \ldots, N$ do
    Input: $x_n(t)$
    aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1)$ from all nodes $\in \mathcal{N}_n$
    $\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$
    locally average $\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{h}}_n(t) = \boldsymbol{P}_n(t-1)\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{g}}_n(t) = \underline{\boldsymbol{h}}_n(t)/[\beta + \underline{\boldsymbol{y}}_n^H(t)\underline{\boldsymbol{h}}_n(t)]$
    $\boldsymbol{P}_n(t) = \frac{1}{\beta}(\boldsymbol{P}_n(t-1) - \underline{\boldsymbol{g}}_n(t)\underline{\boldsymbol{h}}_n^H(t))$
    $\underline{\boldsymbol{e}}_n(t) = \boldsymbol{D}_n(\underline{\boldsymbol{x}}(t) - \boldsymbol{W}_n(t-1)\underline{\boldsymbol{y}}_n(t))$
    $\boldsymbol{W}_n(t) = \boldsymbol{W}_n(t-1) + \underline{\boldsymbol{e}}_n(t)\underline{\boldsymbol{g}}_n^H(t)$
  end
end

PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$
for $t := 1, 2, \ldots$ do
  for $n := 1, 2, \ldots, N$ do
    Input: $x_n(t)$
    aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1)$ from all nodes $\in \mathcal{N}_n$
    $\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$
    locally average $\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{h}}_n(t) = \boldsymbol{P}_n(t-1)\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{g}}_n(t) = \underline{\boldsymbol{h}}_n(t)/[\beta + \underline{\boldsymbol{y}}_n^H(t)\underline{\boldsymbol{h}}_n(t)]$
    $\boldsymbol{P}_n(t) = \frac{1}{\beta}(\boldsymbol{P}_n(t-1) - \underline{\boldsymbol{g}}_n(t)\underline{\boldsymbol{h}}_n^H(t))$
    $\underline{\boldsymbol{e}}_n(t) = \boldsymbol{D}_n(\underline{\boldsymbol{x}}(t) - \boldsymbol{W}_n(t-1)\underline{\boldsymbol{y}}_n(t))$
    $\boldsymbol{W}_n(t) = \boldsymbol{W}_n(t-1) + \underline{\boldsymbol{e}}_n(t)\underline{\boldsymbol{g}}_n^H(t)$
    broadcast $\{x_n(t), \underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes $\in \mathcal{N}_n$
  end
end

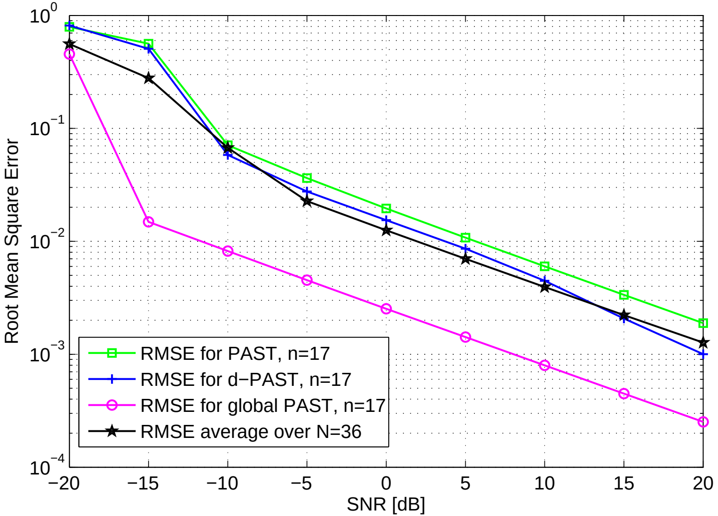
PAST-Consensus Propagation Algorithm

Initialize: $\beta, \boldsymbol{P}_1(0), \ldots, \boldsymbol{P}_N(0), \boldsymbol{W}_1(0), \ldots, \boldsymbol{W}_N(0)$
for $t := 1, 2, \ldots$ do
  for $n := 1, 2, \ldots, N$ do
    Input: $x_n(t)$
    aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1)$ from all nodes $\in \mathcal{N}_n$
    $\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$
    locally average $\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{h}}_n(t) = \boldsymbol{P}_n(t-1)\underline{\boldsymbol{y}}_n(t)$
    $\underline{\boldsymbol{g}}_n(t) = \underline{\boldsymbol{h}}_n(t)/[\beta + \underline{\boldsymbol{y}}_n^H(t)\underline{\boldsymbol{h}}_n(t)]$
    $\boldsymbol{P}_n(t) = \frac{1}{\beta}(\boldsymbol{P}_n(t-1) - \underline{\boldsymbol{g}}_n(t)\underline{\boldsymbol{h}}_n^H(t))$
    $\underline{\boldsymbol{e}}_n(t) = \boldsymbol{D}_n(\underline{\boldsymbol{x}}(t) - \boldsymbol{W}_n(t-1)\underline{\boldsymbol{y}}_n(t))$
    $\boldsymbol{W}_n(t) = \boldsymbol{W}_n(t-1) + \underline{\boldsymbol{e}}_n(t)\underline{\boldsymbol{g}}_n^H(t)$
    broadcast $\{x_n(t), \underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes $\in \mathcal{N}_n$
  end
end

Simulation Parameters

Parameter	Variable	Value
Number of nodes	N	36
Number of incoming signals	r	1
Frequency= cos(DOA)	$\omega_r(t)$	0.1
Max. number of snapshots	$tmax$	1000
Forgetting factor	β	0.97
Transmission radius		9
Topology		Planar array
SNR		-20dB to 20dB

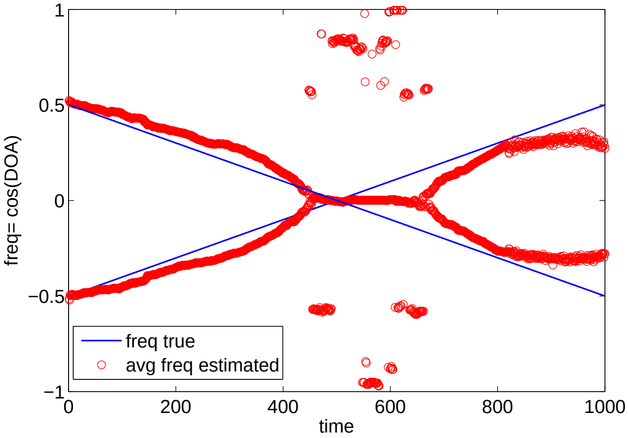
Performance Evaluation of the RMSE for $(r = 1)$, constant



Simulation Parameters

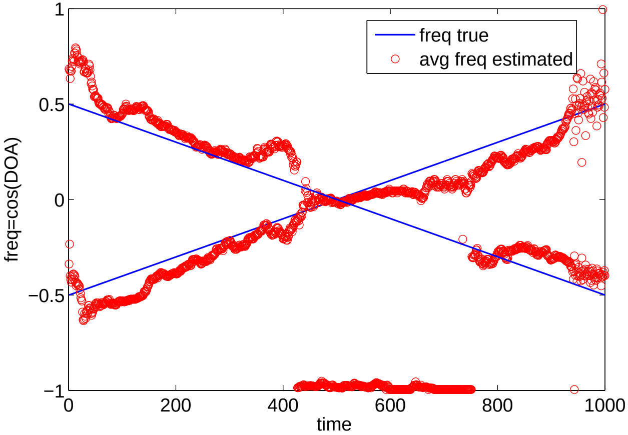
Parameter	Variable	Value
Number of nodes	N	36
Number of incoming signals	r	2
Frequencies= cos(DOA)	$\omega_r(t)$	0.5:-0.5, -0.5:0.5
Max. number of snapshots	$tmax$	1000
Forgetting factor	β	0.97
Transmission radius		9
Topology		Planar array
SNR		3dB

Consensus Propagation



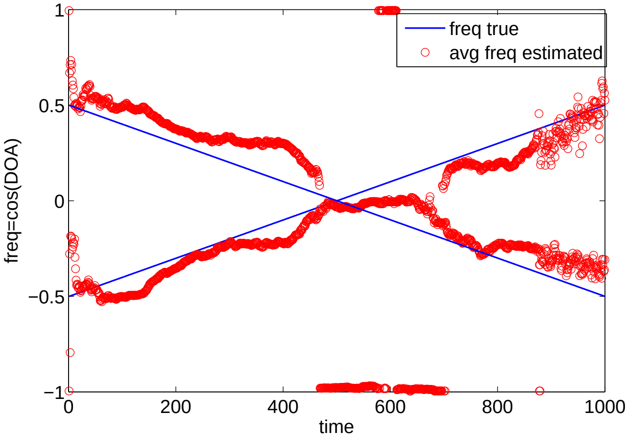
Centralized PAST result for whole sensor array ($N = 36$, $r = 2$)

Consensus Propagation



PAST result neighborhood $\mathcal{N}_{17}$ ($N = 6$, $r = 2$)

Consensus Propagation



Distributed PAST-Consensus result for sensor No. 17 ($r = 2$)

Summary

- Locally average the vector $\underline{\boldsymbol{y}}_n(t)$ in n with information from $\mathcal{N}_n$.
- n broadcasts its local observation $x_n(t)$, the locally filtered r-dimensional vector $\underline{\boldsymbol{y}}_n(t)$, and a weight w_n.
- $\underline{\boldsymbol{y}}_n(t)$ contains information from the updated signal subspace at $t-1$ as well as new observation data $\underline{\boldsymbol{x}}_n(t)$.

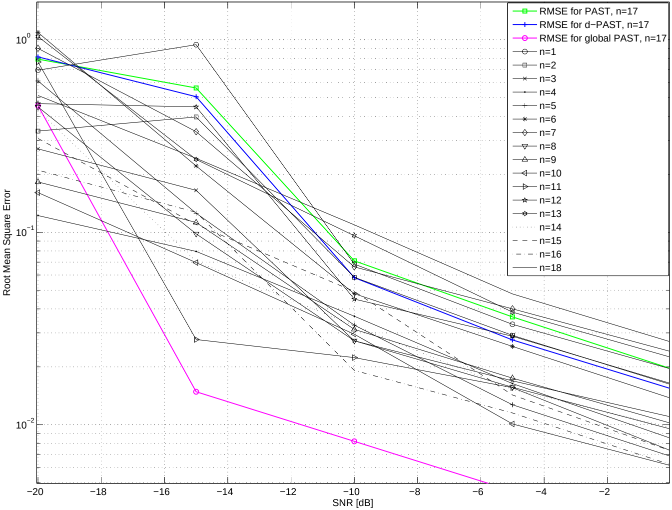
Preliminary Conclusions

- Signal subspace tracking can be implemented without a centralised fusion center.
- Current RMSE performance shows benefits, but also plenty of room for improvement.

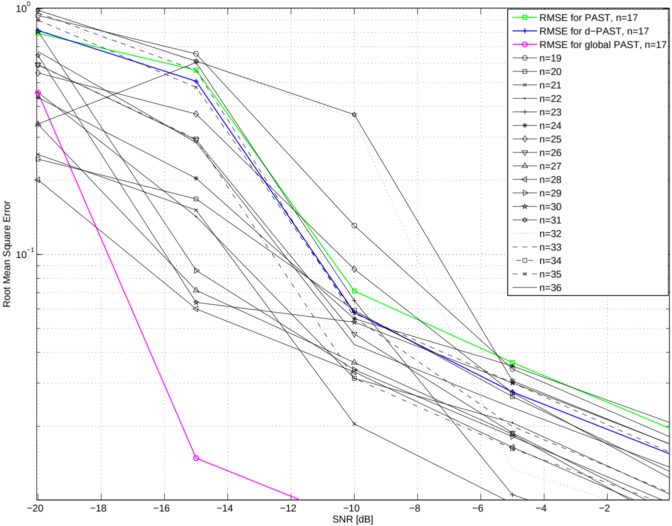
Next Steps

- How to select suitable weights?
- Only do consensus propagation on $\underline{\boldsymbol{y}}_n(t)$? Or also on $\boldsymbol{W}_n(t)$?
- Alternative approach based on distributed RLS

Root Mean Square Error for $N = 1to18$



Root Mean Square Error for $N = 19to36$



RMSE Definition

$$\text{RMSE} = \sqrt{\frac{1}{901}\sum_{t=100}^{1000} |\omega_1(t) - \hat{\omega}_1(t)|^2} \tag{1}$$

$$\text{Average RMSE} = \frac{1}{36}\sqrt{\frac{1}{901}\sum_{t=100}^{1000} |\omega_1(t) - \hat{\omega}_1(t)|^2} \tag{2}$$