On the Denoising of Cross-Spectral Matrices for (Aero)Acoustic Applications

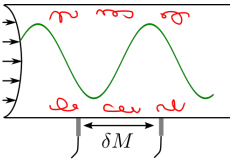
A. Dinsenmeyer[1,2], J. Antoni[1], Q. Leclère[1] and A. Pereira[2]

[1] Laboratoire Vibrations Acoustique
[2] Laboratoire de Mécanique des Fluides et d'Acoustique
Lyon, France

March 5, 2018 – 7th BeBeC

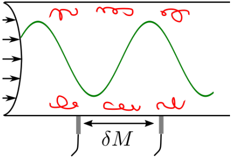
Context

- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



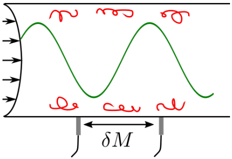
Context

- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



- Existing denoising methods:

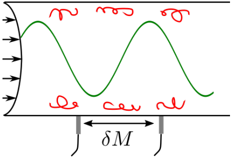
Context



- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths

- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering. . .

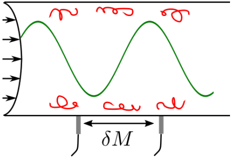
Context



- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths
- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering. . .
  - Use a background noise measurement, $\rightarrow$ not always available or representative

Context

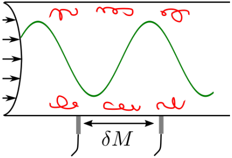
- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering. . .
  - Use a background noise measurement, $\rightarrow$ not always available or representative
  - Wavenumber filtering

Context

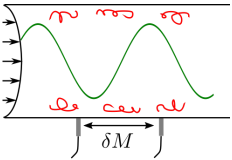
- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering...
  - Use a background noise measurement, $\rightarrow$ not always available or representative
  - Wavenumber filtering
  - Diagonal removal $\rightarrow$ underestimation of source level

Context

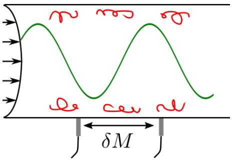
- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering. . .
  - Use a background noise measurement, $\rightarrow$ not always available or representative
  - Wavenumber filtering
  - Diagonal removal $\rightarrow$ underestimation of source level
  - Exploit noise/signal properties & solve an optimization problem

Context

- Unwanted random noise:
  - electronic, ambient, flow-induced,...
  - short correlation lengths



- Existing denoising methods:
  - Physical removal : mic recession, porous treatment, vibrating structure filtering. . .
  - Use a background noise measurement, $\rightarrow$ not always available or representative
  - Wavenumber filtering
  - Diagonal removal $\rightarrow$ underestimation of source level
  - Exploit noise/signal properties & solve an optimization problem

Context – CSM properties

$$\underbrace{\boldsymbol{p}}_{\text{measured spectra}} = \underbrace{\boldsymbol{a}}_{\text{source spectrum}} + \underbrace{\boldsymbol{n}}_{\text{Gaussian noise}}$$

Context – CSM properties

$$\left\langle \underbrace{\boldsymbol{p}}_{\text{measured spectra}} = \underbrace{\boldsymbol{a}}_{\text{source spectrum}} + \underbrace{\boldsymbol{n}}_{\text{Gaussian noise}} \right\rangle_{N_s} \text{ snapshots}$$

Cross-Spectral Matrix (covariance of Fourier component):

$$\boldsymbol{S}_{pp} = \frac{1}{N_s} \sum_i \boldsymbol{p}_i \boldsymbol{p}_i^H$$

Context – CSM properties

$$\left\langle \underbrace{\boldsymbol{p}}_{\text{measured spectra}} = \underbrace{\color{green}{\boldsymbol{a}}}_{\text{source spectrum}} + \underbrace{\color{red}{\boldsymbol{n}}}_{\text{Gaussian noise}} \right\rangle_{N_s} \text{ snapshots}$$

Cross-Spectral Matrix (covariance of Fourier component):

$$\boldsymbol{S}_{pp} = \frac{1}{N_s} \sum_i \boldsymbol{p}_i \boldsymbol{p}_i^H$$

- Hermitian (conjugate symmetric)

Context – CSM properties

$$\left\langle \underbrace{\boldsymbol{p}}_{\text{measured spectra}} = \underbrace{\boldsymbol{a}}_{\text{source spectrum}} + \underbrace{\boldsymbol{n}}_{\text{Gaussian noise}} \right\rangle_{N_s} \text{ snapshots}$$

Cross-Spectral Matrix (covariance of Fourier component):

$$\boldsymbol{S}_{pp} = \frac{1}{N_s} \sum_i \boldsymbol{p}_i \boldsymbol{p}_i^H$$

- Hermitian (conjugate symmetric)
- Positive semidefinite (nonnegative eigenvalues)

Context – CSM properties

$$\left\langle \underbrace{\boldsymbol{p}}_{\text{measured spectra}} = \underbrace{\boldsymbol{a}}_{\text{source spectrum}} + \underbrace{\boldsymbol{n}}_{\text{Gaussian noise}} \right\rangle_{N_s} \text{ snapshots}$$

Cross-Spectral Matrix (covariance of Fourier component):

$$\boldsymbol{S}_{pp} = \frac{1}{N_s} \sum_i \boldsymbol{p}_i \boldsymbol{p}_i^H$$

- Hermitian (conjugate symmetric)
- Positive semidefinite (nonnegative eigenvalues)

$$\underbrace{\boldsymbol{S}_{pp}}_{\text{measured CSM}} = \underbrace{\boldsymbol{S}_{aa}}_{\text{signal of interest}} + \underbrace{\boldsymbol{S}_{nn}}_{\text{unwanted noise}} + \underbrace{\boldsymbol{S}_{an} + \boldsymbol{S}_{na}}_{\text{cross-terms}}$$

- Rank of $\boldsymbol{S}_{aa}$ = number of uncorrelated monopoles

Context – CSM properties

$$\underbrace{\boldsymbol{S}_{pp}}_{\text{measured CSM}} = \underbrace{\boldsymbol{S}_{aa}}_{\text{signal of interest}} + \underbrace{\boldsymbol{S}_{nn}}_{\text{unwanted noise}} + \underbrace{\boldsymbol{S}_{an} + \boldsymbol{S}_{na}}_{\text{cross-terms}}$$

For $N_s \to \infty$

Context – CSM properties

$$\underbrace{\boldsymbol{S}_{pp}}_{\text{measured CSM}} = \underbrace{\boldsymbol{S}_{aa}}_{\text{signal of interest}} + \underbrace{\cancel{\boldsymbol{S}_{nn}}}_{\substack{\text{unwanted noise} \\ \approx \operatorname{diag}(\sigma^2)}} + \underbrace{\boldsymbol{S}_{an} + \boldsymbol{S}_{na}}_{\text{cross-terms}}$$

For $N_s \to \infty$

- Short correlation length : off-diagonal elements of $S_{nn} \to 0$

Context – CSM properties

$$\underbrace{\boldsymbol{S}_{pp}}_{\text{measured CSM}} = \underbrace{\color{green}{\boldsymbol{S}_{aa}}}_{\text{signal of interest}} + \underbrace{\cancel{\color{red}{\boldsymbol{S}_{nn}}}}_{\substack{\text{unwanted noise} \\ \approx \operatorname{diag}\left(\color{red}{\sigma^2}\right)}} + \underbrace{\cancel{\boldsymbol{S}_{an} + \boldsymbol{S}_{na}}}_{\substack{\text{cross-terms} \\ \to 0}}$$

For $N_s \to \infty$

- Short correlation length : off-diagonal elements of $S_{nn} \to 0$
- Independent signal/noise : cross-terms $\to 0$

How to separate signal part from noise ?

- Existing methods:
  - 3 diagonal reconstruction methods
  - Robust Principal Component Analysis (RPCA)

How to separate signal part from noise ?

- Existing methods:
  - 3 diagonal reconstruction methods
  - Robust Principal Component Analysis (RPCA)

- Proposed method:
  - PFA

How to separate signal part from noise ?

- Existing methods:
  - 3 diagonal reconstruction methods
  - Robust Principal Component Analysis (RPCA)
- Proposed method:
  - PFA
- What is the influence on denoising performance of :
  - noise level,
  - number of snapshots,
  - number of sources ?

1 Diagonal Reconstruction
Comparison on a test case

2 RPCA

3 Probabilistic Factor Analysis

4 Comparison

Diagonal Reconstruction

"Remove as much noise as possible as long as denoised CSM remains positive"

Convex optimization (Hald, 2017)

$$\text{maximize } \|\boldsymbol{\sigma}_n^2\|_1 \quad \text{subject to} \quad \boldsymbol{S}_{pp} - \mathrm{diag}\left(\boldsymbol{\sigma}_n^2\right) \geq 0$$

Problem solved with CVX Matlab toolbox.

Diagonal Reconstruction

"Remove as much noise as possible as long as denoised CSM remains positive"

Convex optimization (Hald, 2017)

$$\text{maximize } \|\boldsymbol{\sigma}_n^2\|_1 \text{ subject to } \boldsymbol{S}_{pp} - \operatorname{diag}\left(\boldsymbol{\sigma}_n^2\right) \geq 0$$

Problem solved with CVX Matlab toolbox.

Linear optimization (Dougherty, 2016)

$$\text{maximize } \|\boldsymbol{\sigma}_n^2\|_1 \text{ subject to } \boldsymbol{V}_{(k-1)}^H \left(\boldsymbol{S}_{pp} - \operatorname{diag}\left(\boldsymbol{\sigma}_n^2\right)_{(k)}\right) \boldsymbol{V}_{(k-1)} \geq 0$$

$\boldsymbol{V}_{(k-1)}$: eigenvectors of $\boldsymbol{S}_{pp} - \operatorname{diag}\left(\boldsymbol{\sigma}_n^2\right)_{(1,\ldots,k-1)}$

Solved with *linprog* Matlab function .

Diagonal Reconstruction

"Remove as much noise as possible as long as denoised CSM remains positive"

Convex optimization (Hald, 2017)

$$\text{maximize } \|\boldsymbol{\sigma}_n^2\|_1 \ \text{ subject to } \ \boldsymbol{S}_{pp} - \text{diag}\left(\boldsymbol{\sigma}_n^2\right) \geq 0$$

Problem solved with CVX Matlab toolbox.

Linear optimization (Dougherty, 2016)

$$\text{maximize } \|\boldsymbol{\sigma}_n^2\|_1 \ \text{ subject to } \ \boldsymbol{V}_{(k-1)}^H \left(\boldsymbol{S}_{pp} - \text{diag}\left(\boldsymbol{\sigma}_n^2\right)_{(k)}\right) \boldsymbol{V}_{(k-1)} \geq 0$$

$\boldsymbol{V}_{(k-1)}$: eigenvectors of $\boldsymbol{S}_{pp} - \text{diag}\left(\boldsymbol{\sigma}_n^2\right)_{(1,\ldots,k-1)}$
Solved with *linprog* Matlab function .

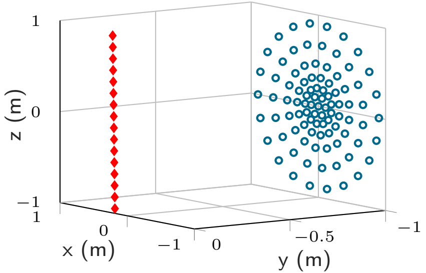
Alternating Projections (Leclère et al., 2015)

$$\boldsymbol{S}_{pp_{(k+1)}} := \bar{\boldsymbol{S}}_{pp_{(0)}} + \boldsymbol{V}_{(k)}^H \boldsymbol{s}_{(k)}^{+} \boldsymbol{V}_{(k)}$$

$\boldsymbol{V}_{(k)}^H$ and $\boldsymbol{s}_{(k)}$: eigenvectors/values of $\boldsymbol{S}_{pp_{(k)}}$.

Diagonal Reconstruction

- Default parameters:
  - 20 uncorrelated free field monopoles: ♦
  - 93 receivers: ○
  - SNR: 10 dB
  - 10^4 snapshots
  - frequency: 15 kHz



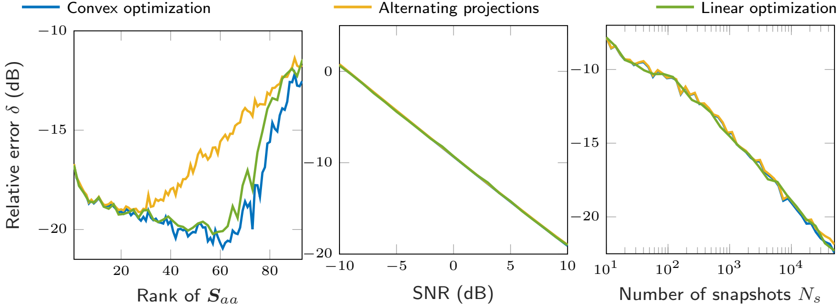
- Varying parameters:
  - number of ♦ (rank of $\boldsymbol{S}_{aa}$) : from 1 to 93
  - SNR from -10 to 10 dB
  - Number of snapshots (level of extra-diagonal terms): from 10 to 5.10^4
- Error on the signal CSM:

$$\delta = \frac{\| \operatorname{diag}(\boldsymbol{S}_{aa}) - \operatorname{diag}\left(\hat{\boldsymbol{S}}_{aa}\right) \|_2}{\| \operatorname{diag}(\boldsymbol{S}_{aa}) \|_2}$$

Diagonal Reconstruction

Default values:

20 sources	93 receivers	SNR: 10 dB	10^4 snapshots	frequency: 15 kHz



Select Convex Optimization (DRec) for further comparison

✓ Fast, simple code
✓ Better performance

✗ Local optimization
✗ Denoises only auto-spectra

RPCA

"Search $\boldsymbol{S}_{aa}$ as a low rank matrix and $\boldsymbol{S}_{nn}$ as a sparse matrix"

$$\text{minimize } \|\boldsymbol{S}_{aa}\|_* + \lambda \|\boldsymbol{S}_{nn}\|_1 \quad \text{subject to} \quad \boldsymbol{S}_{aa} + \boldsymbol{S}_{nn} = \boldsymbol{S}_{pp}$$

$\|\cdot\|_*$: nuclear norm (related to rank)
$\|\cdot\|_1$: ℓ_1-norm (related to sparsity)

Solved with a proximal gradient algorithm.

RPCA (Wright et al., 2009)

✓ Modifies the whole CSM

✗ Local optimization

✗ Choose regularization parameter:

- L-curve criterion,
- Generalized cross validation method,
- Bayesian criterion, ...

↪ For comparison : - optimal λ (unknown on real case)
- "universal" constant parameter $\lambda = M^{-\frac{1}{2}} = 0.1$

Probabilistic Factor Analysis

- **Latent variable model**

$$\underbrace{\boldsymbol{p}}_{\text{observable variables}} = \underbrace{\boldsymbol{L}}_{\text{matrix of loading factors}}\underbrace{\boldsymbol{c}}_{\text{latent factors}} + \underbrace{\boldsymbol{n}}_{\text{error (noise)}}$$

Probabilistic Factor Analysis

- **Latent variable model**

observable variables $\longrightarrow$ $\boxed{\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}}$

matrix of loading factors

latent factors

error (noise)

  - Capture dominant correlation with fewer parameters (close to PCA)

Probabilistic Factor Analysis

- **Latent variable model**

observable variables

$$\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}$$

matrix of loading factors

latent factors

error (noise)

- Capture dominant correlation with fewer parameters (close to PCA)
- Extract anisotropic noise

Probabilistic Factor Analysis

- **Latent variable model**

observable variables

$$\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}$$

matrix of loading factors

latent factors

error (noise)

- Capture dominant correlation with fewer parameters (close to PCA)
- Extract anisotropic noise

- **Statistical inference:** See parameters as random variables

$$\boldsymbol{L} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{\gamma}^2) \qquad \boldsymbol{c} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\alpha}^2) \qquad \boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\sigma}^2)$$

Probabilistic Factor Analysis

- **Latent variable model**

observable variables

$$\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}$$

matrix of loading factors

latent factors

error (noise)

  - Capture dominant correlation with fewer parameters (close to PCA)
  - Extract anisotropic noise

- **Statistical inference:** See parameters as random variables

$$\boldsymbol{L} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{\gamma}^2) \qquad \boldsymbol{c} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\alpha}^2) \qquad \boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\sigma}^2)$$

$+$ non-informative priors : $\boldsymbol{\gamma}^2, \boldsymbol{\alpha}^2, \boldsymbol{\sigma}^2 \sim \mathcal{IG}(a_{\gamma,\alpha,\sigma}, b_{\gamma,\alpha,\sigma})$

Probabilistic Factor Analysis

- **Latent variable model**

observable variables

$$\boxed{\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}}$$

matrix of loading factors

latent factors

error (noise)

  - Capture dominant correlation with fewer parameters (close to PCA)
  - Extract anisotropic noise

- **Statistical inference:** See parameters as random variables

$$\boldsymbol{L} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{\gamma}^2) \qquad \boldsymbol{c} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\alpha}^2) \qquad \boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\sigma}^2)$$

$+$ non-informative priors : $\boldsymbol{\gamma}^2, \boldsymbol{\alpha}^2, \boldsymbol{\sigma}^2 \sim \mathcal{IG}(a_{\gamma,\alpha,\sigma}, b_{\gamma,\alpha,\sigma})$

- **Solved using MCMC algorithm** (Gibb's sampling)
  Iteratively draws in the marginal conditional distribution of each parameter

Probabilistic Factor Analysis

- **Latent variable model**

observable variables $\longrightarrow$ $\boxed{\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}}$

matrix of loading factors — latent factors — error (noise)

  - Capture dominant correlation with fewer parameters (close to PCA)
  - Extract anisotropic noise

- **Statistical inference:** See parameters as random variables

$$\boldsymbol{L} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{\gamma}^2) \qquad \boldsymbol{c} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\alpha}^2) \qquad \boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(0, \boldsymbol{I}\boldsymbol{\sigma}^2)$$

$+$ non-informative priors : $\boldsymbol{\gamma}^2, \boldsymbol{\alpha}^2, \boldsymbol{\sigma}^2 \sim \mathcal{IG}(a_{\gamma,\alpha,\sigma}, b_{\gamma,\alpha,\sigma})$

- **Solved using MCMC algorithm** (Gibb's sampling)
  Iteratively draws in the marginal conditional distribution of each parameter

- **Finally**, signal CSM:

$$\hat{\boldsymbol{S}}_{aa} = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{L}\boldsymbol{c}_i\boldsymbol{c}_i^H \boldsymbol{L}^H$$

Probabilistic Factor Analysis

- **Latent variable model**

observable variables

$$\boldsymbol{p} = \boldsymbol{L}\boldsymbol{c} + \boldsymbol{n}$$

matrix of loading factors

latent factors

error (noise)

- Capture dominant correlation with fewer parameters (close to PCA)
- Extract anisotropic noise

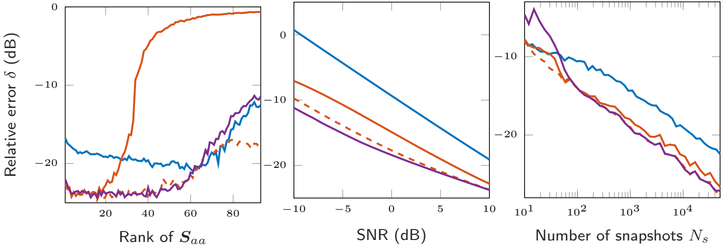
PFA

✓ Global optimization
✓ Flexible model

✗ Computationally expensive

Comparison

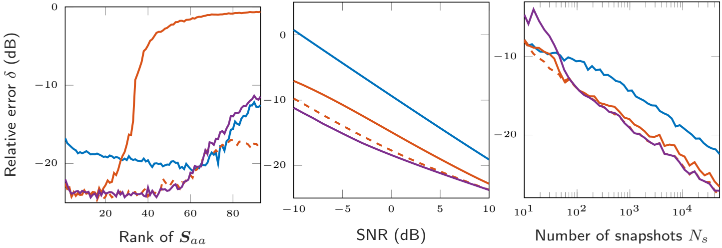
- **Homogeneous noise**



Comparison

DRec — RPCA, λ_{opt} — RPCA, $\lambda = M^{-\frac{1}{2}}$ — PFA

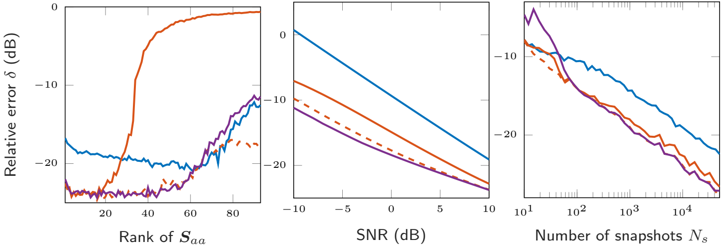
- **Homogeneous noise**



$\hookrightarrow$ For $N_{src} \geq 0.75M$: denoising problem becomes poorly conditioned

Comparison

- **Homogeneous noise**



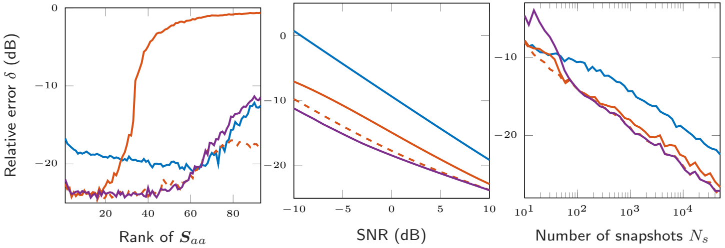
$\hookrightarrow$ For $N_{src} \geq 0.75M$: denoising problem becomes poorly conditioned

$\hookrightarrow$ Error linearly decreases with increasing SNR

Comparison

DRec — RPCA, λ_{opt} — RPCA, $\lambda = M^{-\frac{1}{2}}$ — PFA

- **Homogeneous noise**



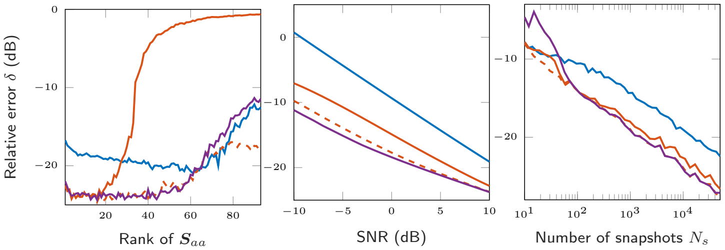
↪ For $N_{src} \geq 0.75M$: denoising problem becomes poorly conditioned

↪ Error linearly decreases with increasing SNR

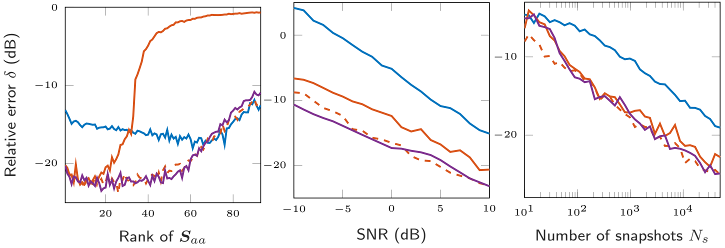
↪ Error linearly decreases with logarithmically increasing N_s

Comparison

- **Homogeneous noise**



- **Heterogeneous noise**: SNR 10 dB lower on 10 random receivers



Conclusion

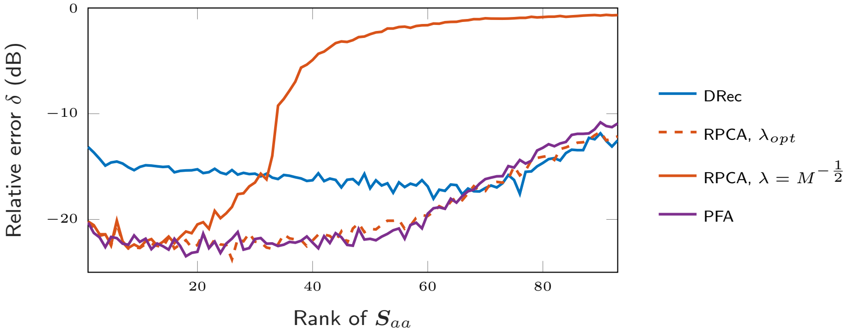
- DRec: fast and simple but error at least 5 dB higher for rank under 70

Conclusion

- DRec: fast and simple but error at least 5 dB higher for rank under 70
- PFA
  - performance similar to RPCA using λ_{opt}
  - PFA and RPCA more robust to heterogeneous noise
  - flexible model $\rightarrow$ to be adapted for correlated noise

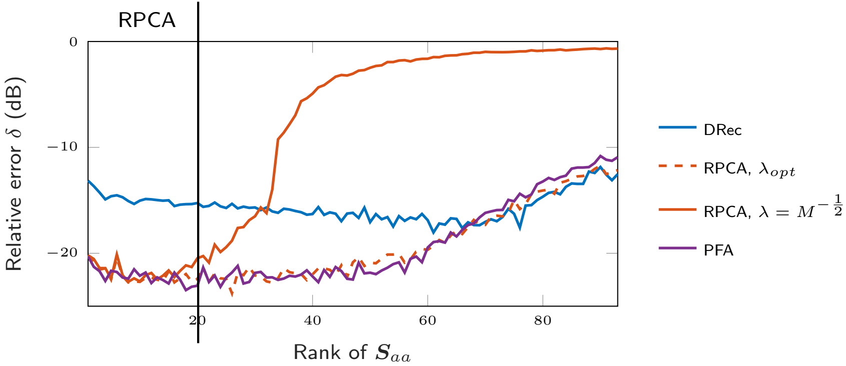
Conclusion

- DRec: fast and simple but error at least 5 dB higher for rank under 70
- PFA
  - performance similar to RPCA using λ_{opt}
  - PFA and RPCA more robust to heterogeneous noise
  - flexible model $\rightarrow$ to be adapted for correlated noise
- Choose your denoising method according to the expected number of sources



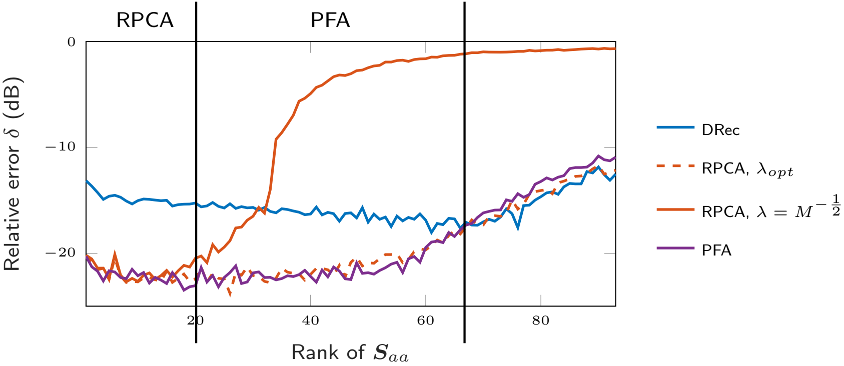
Conclusion

- DRec: fast and simple but error at least 5 dB higher for rank under 70
- PFA
  - performance similar to RPCA using λ_{opt}
  - PFA and RPCA more robust to heterogeneous noise
  - flexible model $\rightarrow$ to be adapted for correlated noise
- Choose your denoising method according to the expected number of sources



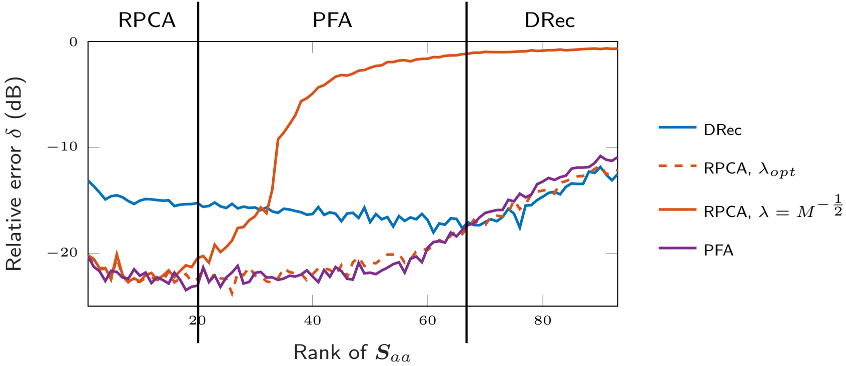
Conclusion

- DRec: fast and simple but error at least 5 dB higher for rank under 70
- PFA
  - performance similar to RPCA using λ_{opt}
  - PFA and RPCA more robust to heterogeneous noise
  - flexible model $\rightarrow$ to be adapted for correlated noise
- Choose your denoising method according to the expected number of sources



Conclusion

- DRec: fast and simple but error at least 5 dB higher for rank under 70
- PFA
  - performance similar to RPCA using λ_{opt}
  - PFA and RPCA more robust to heterogeneous noise
  - flexible model $\rightarrow$ to be adapted for correlated noise
- Choose your denoising method according to the expected number of sources



References

R. Dougherty. Cross spectral matrix diagonal optimization. In *6th Berlin Beamforming Conference*, 02 2016.

J. Hald. Removal of incoherent noise from an averaged cross-spectral matrix. *The Journal of the Acoustical Society of America*, 142(2):846–854, 2017.

Q. Leclère, N. Totaro, C. Pézerat, F. Chevillotte, and P. Souchotte. Extraction of the acoustic part of a turbulent boundary layer from wall pressure and vibration measurements. In *Novem 2015 - Noise and vibration - Emerging technologies*, Proceedings of Novem 2015, page 49046, Dubrovnik, Croatia, Apr. 2015.

J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *Advances in neural information processing systems*, pages 2080–2088, 2009.