Tensor-based Adaptive Techniques: A Deep Diving in Nonlinear Systems

Laura-Maria Dogariu

Faculty of Electronics, Telecommunications and Information Technology
University Politehnica of Bucharest
ldogariu@comm.pub.ro

IARIA NetWare Congress, November 21-26, 2020
Valencia, Spain

Outline

About the Presenter

Laura-Maria Dogariu received a Bachelor degree in telecommunications systems from the Faculty of Electronics and Telecommunications (ETTI), University Politehnica of Bucharest (UPB), Romania, in 2014, and a double Master degree in wireless communications systems from UPB and Centrale Supélec, Université Paris-Saclay (with *Distinction* mention), in 2016. She received a PhD degree with *Excellent* mention (*SUMMA CUM LAUDE*) in 2019 from UPB and is currently a postdoctoral researcher and lecturer at the same university. Her research interests include adaptive filtering algorithms and signal processing. She acts as a reviewer for several important journals and conferences, such as *IEEE Transactions on Signal Processing, Signal Processing, IEEE International Symposium on Signals, Circuits and Systems (ISSCS)*. She was the recipient of several prizes and scholarships, among which the Paris-Saclay scholarship, the excellence scholarship offered by Orange Romania, and an excellence scholarship from UPB. Laura Dogariu is also the winner of the competition for a postdoctoral research grant on adaptive algorithms for multilinear system identification using tensor modelling, financed by the Romanian Government, starting in 2021 (first place, with the maximum score).

Outline

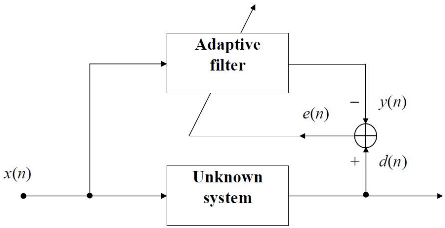

Figure 1: System identification configuration

- **System identification**: estimate a model (unknown system) based on the available and observed data (usually input and output of the system), using an adaptive filter

Introduction

- **Multidimensional** system identification:
  $\rightarrow$ modeled using tensors
  $\rightarrow$ multilinearity is defined with respect to the impulse responses composing the complex system (as opposed to the classical approach, referring to the input-output relation) $\Rightarrow$ multilinear in parameters system

Introduction

- **Multidimensional** system identification:
  $\rightarrow$ modeled using tensors
  $\rightarrow$ multilinearity is defined with respect to the impulse responses composing the complex system (as opposed to the classical approach, referring to the input-output relation) $\Rightarrow$ multilinear in parameters system
- **Purpose**: analyzing and developing adaptive algorithms for multilinear in parameters systems

- **Multidimensional** system identification:
  $\rightarrow$ modeled using tensors
  $\rightarrow$ multilinearity is defined with respect to the impulse responses composing the complex system (as opposed to the classical approach, referring to the input-output relation) $\Rightarrow$ multilinear in parameters system
- **Purpose**: analyzing and developing adaptive algorithms for multilinear in parameters systems
- **Possible applications**:
  $\rightarrow$ identification of Hammerstein systems
  $\rightarrow$ nonlinear acoustic echo cancellation $\Rightarrow$ multi-party voice communications (e.g., videoconference solutions)
  $\rightarrow$ source separation
  $\rightarrow$ tensor algebra - big data
  $\rightarrow$ algorithms for machine learning

Outline

System Model for Bilinear Forms

- **Signal model:** $d(n) = y(n) + v(n) = \mathbf{h}^T(n)\mathbf{X}(n)\mathbf{g}(n) + v(n)$

  $\rightarrow d(n)$ - reference (desired) signal

  $\rightarrow$ output signal $y(n)$ - bilinear form with respect to the impulse responses

  $\rightarrow \mathbf{h}$, $\mathbf{g}$ - unknown system impulse responses of lengths L, M:

  $\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{w_h}(n) \qquad \mathbf{g}(n) = \mathbf{g}(n-1) + \mathbf{w_g}(n)$

  $\mathbf{w_h}(n)$, $\mathbf{w_g}(n)$: zero-mean WGN

  $$\mathbf{R}_{\mathbf{w_h}}(n) = \sigma^2_{w_\mathbf{h}}\mathbf{I}_L \qquad \mathbf{R}_{\mathbf{w_g}}(n) = \sigma^2_{w_\mathbf{g}}\mathbf{I}_M$$

  $\rightarrow \mathbf{X}(n) = [\mathbf{x}_1(n) \quad \mathbf{x}_2(n) \quad \dots \quad \mathbf{x}_M(n)]$ - input signal matrix

  $\rightarrow \mathbf{x}_m(n) = [x_m(n) \;\; x_m(n-1) \;\; \dots \;\; x_m(n-L+1)]^T$,
  $m = 1, 2, \dots, M$

  $\rightarrow v(n)$: zero-mean WGN

System Model for Bilinear Forms

- **Signal model:** $d(n) = y(n) + v(n) = \mathbf{h}^T(n)\mathbf{X}(n)\mathbf{g}(n) + v(n)$
  $\rightarrow d(n)$ - reference (desired) signal
  $\rightarrow$ output signal $y(n)$ - bilinear form with respect to the impulse responses
  $\rightarrow \mathbf{h}$, $\mathbf{g}$ - unknown system impulse responses of lengths L, M:
  $\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{w_h}(n)$ $\quad\quad$ $\mathbf{g}(n) = \mathbf{g}(n-1) + \mathbf{w_g}(n)$
  $\mathbf{w_h}(n)$, $\mathbf{w_g}(n)$: zero-mean WGN
  $$\mathbf{R}_{\mathbf{w_h}}(n) = \sigma^2_{w_\mathbf{h}}\mathbf{I}_L \qquad \mathbf{R}_{\mathbf{w_g}}(n) = \sigma^2_{w_\mathbf{g}}\mathbf{I}_M$$
  $\rightarrow \mathbf{X}(n) = [\mathbf{x}_1(n) \quad \mathbf{x}_2(n) \quad \ldots \quad \mathbf{x}_M(n)]$ - input signal matrix
  $\rightarrow \mathbf{x}_m(n) = [x_m(n) \;\; x_m(n-1) \;\; \ldots \;\; x_m(n-L+1)]^T$,
  $m = 1, 2, \ldots, M$
  $\rightarrow v(n)$: zero-mean WGN

- **Equivalent model:** $d(n) = \mathbf{f}^T(n)\widetilde{\mathbf{x}}(n) + v(n)$
  $\rightarrow \mathbf{f}(n) = \mathbf{g}(n) \otimes \mathbf{h}(n)$ – Kronecker product of length ML
  $\rightarrow \widetilde{\mathbf{x}}(n) = \mathrm{vec}[\mathbf{X}(n)] = [\mathbf{x}_1^T(n) \quad \mathbf{x}_2^T(n) \quad \ldots \quad \mathbf{x}_M^T(n)]^T$

System Model for Bilinear Forms

- Estimated output signal: $\widehat{y}(n) = \widehat{\mathbf{h}}^T(n-1)\mathbf{X}(n)\widehat{\mathbf{g}}(n-1)$

System Model for Bilinear Forms

- Estimated output signal: $\widehat{y}(n) = \widehat{\mathbf{h}}^T(n-1)\mathbf{X}(n)\widehat{\mathbf{g}}(n-1)$
- Error signal:

$$\begin{aligned}
e(n) &= d(n) - \widehat{y}(n) \\
&= d(n) - \widehat{\mathbf{f}}^T(n-1)\widetilde{\mathbf{x}}(n) \\
&= d(n) - \widehat{\mathbf{h}}^T(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n) \leftarrow e_{\widehat{\mathbf{g}}}(n) \\
&= d(n) - \widehat{\mathbf{g}}^T(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n) \leftarrow e_{\widehat{\mathbf{h}}}(n) \\
&= [\mathbf{g}(n) \otimes \mathbf{h}(n)]^T \widetilde{\mathbf{x}}(n) + v(n) - \left[\widehat{\mathbf{g}}(n-1) \otimes \widehat{\mathbf{h}}(n-1)\right]^T \widetilde{\mathbf{x}}(n) \\
&= \mathbf{h}^T(n)\mathbf{x}_{\mathbf{g}}(n) + v(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{g}}}(n) \\
&= \mathbf{g}^T(n)\mathbf{x}_{\mathbf{h}}(n) + v(n) - \widehat{\mathbf{g}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}}(n)
\end{aligned}$$

$$\mathbf{x}_{\mathbf{g}}(n) = [\mathbf{g}(n) \otimes \mathbf{I}_L]^T\widetilde{\mathbf{x}}(n) \qquad \mathbf{x}_{\mathbf{h}}(n) = [\mathbf{I}_M \otimes \mathbf{h}(n)]^T\widetilde{\mathbf{x}}(n)$$

$$\mathbf{x}_{\widehat{\mathbf{g}}}(n) = [\widehat{\mathbf{g}}(n-1) \otimes \mathbf{I}_L]^T\widetilde{\mathbf{x}}(n) \qquad \mathbf{x}_{\widehat{\mathbf{h}}}(n) = [\mathbf{I}_M \otimes \widehat{\mathbf{h}}(n-1)]^T\widetilde{\mathbf{x}}(n)$$

Optimized LMS Algorithm for Bilinear Forms

The desired signal can be written in two equivalent forms:

- $$\begin{aligned} d(n) &= \mathbf{g}^T(n)\mathbf{x}_{\mathbf{h}}(n) + \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) - \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + v(n) \\ &= \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + v_{\mathbf{g}}(n) + v(n) \end{aligned}$$

$v_{\mathbf{g}}(n)$: additional noise term, introduced by the system $\mathbf{g}$

- $$\begin{aligned} d(n) &= \mathbf{h}^T(n)\mathbf{x}_{\mathbf{g}}(n) + \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) - \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + v(n) \\ &= \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + v_{\mathbf{h}}(n) + v(n) \end{aligned}$$

$v_{\mathbf{h}}(n)$: additional noise term, introduced by the system $\mathbf{h}$

Optimized LMS Algorithm for Bilinear Forms

The desired signal can be written in two equivalent forms:

- $$\begin{aligned} d(n) &= \mathbf{g}^T(n)\mathbf{x}_{\mathbf{h}}(n) + \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) - \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + v(n) \\ &= \mathbf{g}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + v_{\mathbf{g}}(n) + v(n) \end{aligned}$$

$v_{\mathbf{g}}(n)$: additional noise term, introduced by the system $\mathbf{g}$

- $$\begin{aligned} d(n) &= \mathbf{h}^T(n)\mathbf{x}_{\mathbf{g}}(n) + \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) - \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + v(n) \\ &= \mathbf{h}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + v_{\mathbf{h}}(n) + v(n) \end{aligned}$$

$v_{\mathbf{h}}(n)$: additional noise term, introduced by the system $\mathbf{h}$

In the context of LMS:

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \mu_{\widehat{\mathbf{g}}}\mathbf{x}_{\widehat{\mathbf{h}}}(n)e(n) \qquad \widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mu_{\widehat{\mathbf{h}}}\mathbf{x}_{\widehat{\mathbf{g}}}(n)e(n)$$

- After computations ⇒ optimal step-size values $\mu_{\widehat{\mathbf{g}},o}, \mu_{\widehat{\mathbf{h}},o}$:

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \frac{x_{\widehat{\mathbf{h}}}(n)e(n)}{M\sigma_{\mathbf{x}}^2 \mathbb{E}\{||\widehat{\mathbf{h}}(n-1)||^2\}} \times \frac{1}{\left[1 + \frac{\mathbb{E}\left\{\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}}(n)\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\mathbf{g}}(n)\right\}+\sigma_v^2+\sigma_{v_{\mathbf{g}}}^2(n)}{\mathbb{E}\left\{\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}}(n)\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\mathbf{g}}(n)\right\}+\sigma_{\mathbf{x}}^2\mathbb{E}\left\{||\widehat{\mathbf{h}}(n-1)||^2\right\}\left[m_{\mathbf{g}}(n-1)+M\sigma_{w_{\mathbf{g}}}^2\right]}\right]}$$

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{x_{\widehat{\mathbf{g}}}(n)e(n)}{L\sigma_{\mathbf{x}}^2 \mathbb{E}\{||\widehat{\mathbf{g}}(n-1)||^2\}} \times \frac{1}{\left[1 + \frac{\mathbb{E}\left\{\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{g}}}(n)\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\mathbf{h}}(n)\right\}+\sigma_v^2+\sigma_{v_{\mathbf{h}}}^2(n)}{\mathbb{E}\left\{\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{g}}}(n)\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\mathbf{h}}(n)\right\}+\sigma_{\mathbf{x}}^2\mathbb{E}\left\{||\widehat{\mathbf{g}}(n-1)||^2\right\}\left[m_{\mathbf{h}}(n-1)+L\sigma_{w_{\mathbf{h}}}^2\right]}\right]}$$

→ $\mathbf{c}_{\mathbf{g}}(n) = \mathbf{g}(n) - \widehat{\mathbf{g}}(n)$, $\mathbf{c}_{\mathbf{h}}(n) = \mathbf{h}(n) - \widehat{\mathbf{h}}(n)$: a posteriori misalignments

→ $m_{\mathbf{g}}(n) = \mathbb{E}\{||\mathbf{c}_{\mathbf{g}}(n)||^2\}$, $m_{\mathbf{h}}(n) = \mathbb{E}\{||\mathbf{c}_{\mathbf{h}}(n)||^2\}$

Scaling Ambiguity

- $\mathbf{f}(n) = \mathbf{g}(n) \otimes \mathbf{h}(n) = [\eta \mathbf{g}(n)] \otimes \left[\frac{1}{\eta}\mathbf{h}(n)\right]$ $\quad \eta \in \mathcal{R}^*$ - scaling factor

$$\left[\frac{1}{\eta}\mathbf{h}(n)\right]^T \mathbf{X}(n) \left[\eta \mathbf{g}(n)\right] = \mathbf{h}^T(n)\mathbf{X}(n)\mathbf{g}(n)$$

Scaling Ambiguity

- $\mathbf{f}(n) = \mathbf{g}(n) \otimes \mathbf{h}(n) = [\eta \mathbf{g}(n)] \otimes \left[\frac{1}{\eta}\mathbf{h}(n)\right] \quad \eta \in \mathcal{R}^*$ - scaling factor

$$\left[\frac{1}{\eta}\mathbf{h}(n)\right]^T \mathbf{X}(n)\,[\eta \mathbf{g}(n)] = \mathbf{h}^T(n)\mathbf{X}(n)\mathbf{g}(n) \Rightarrow \begin{array}{l} \widehat{\mathbf{h}}(n) \to \frac{1}{\eta}\mathbf{h}(n) \\ \widehat{\mathbf{g}}(n) \to \eta \mathbf{g}(n) \\ \widehat{\mathbf{f}}(n) \to \mathbf{f}(n) \end{array}$$

Scaling Ambiguity

- $\mathbf{f}(n) = \mathbf{g}(n) \otimes \mathbf{h}(n) = [\eta \mathbf{g}(n)] \otimes \left[\frac{1}{\eta}\mathbf{h}(n)\right] \quad \eta \in \mathcal{R}^*$ - scaling factor

$$\left[\frac{1}{\eta}\mathbf{h}(n)\right]^T \mathbf{X}(n) \left[\eta \mathbf{g}(n)\right] = \mathbf{h}^T(n)\mathbf{X}(n)\mathbf{g}(n) \Rightarrow \begin{array}{l} \widehat{\mathbf{h}}(n) \to \frac{1}{\eta}\mathbf{h}(n) \\ \widehat{\mathbf{g}}(n) \to \eta \mathbf{g}(n) \\ \widehat{\mathbf{f}}(n) \to \mathbf{f}(n) \end{array}$$

Normalized projection misalignment (NPM):

[Morgan et al., IEEE Signal Processing Letters, July 1998]

$$\text{NPM}[\mathbf{h}(n), \widehat{\mathbf{h}}(n)] = 1 - \left[\frac{\mathbf{h}^T(n)\widehat{\mathbf{h}}(n)}{||\mathbf{h}(n)||\,||\widehat{\mathbf{h}}(n)||}\right]^2$$

$$\text{NPM}[\mathbf{g}(n), \widehat{\mathbf{g}}(n)] = 1 - \left[\frac{\mathbf{g}^T(n)\widehat{\mathbf{g}}(n)}{||\mathbf{g}(n)||\,||\widehat{\mathbf{g}}(n)||}\right]^2$$

Normalized misalignment (NM):

$$\text{NM}[\mathbf{f}(n), \widehat{\mathbf{f}}(n)] = \|\mathbf{f}(n) - \widehat{\mathbf{f}}(n)\|^2 / \|\mathbf{f}(n)\|^2$$

Simulation Setup

- Input signals $x_m(n), m = 1, 2, \ldots, M$ - independent WGN, respectively AR(1) generated by filtering a white Gaussian noise through a first-order system $1/\left(1 - 0.8z^{-1}\right)$
- **h**, **g** - Gaussian, randomly generated, of lengths $L = 64$, $M = 8$
- $v(n)$ - independent WGN of variance $\sigma_v^2 = 0.01$
- Assumptions: $\rightarrow \mathbb{E}\left\{\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{h}}}(n)\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\mathbf{g}}(n)\right\} \stackrel{not.}{=} p_{\mathbf{g}}(n) = 0$
  $\rightarrow \mathbb{E}\left\{\mathbf{c}_{\mathbf{h}}^T(n-1)\mathbf{x}_{\widehat{\mathbf{g}}}(n)\mathbf{c}_{\mathbf{g}}^T(n-1)\mathbf{x}_{\mathbf{h}}(n)\right\} \stackrel{not.}{=} p_{\mathbf{h}}(n) = 0$
- Performance measure - NM for the global filter

Compared algorithms

- **OLMS-BF** and **NLMS-BF** [C. Paleologu et al., "Adaptive filtering for the identification of bilinear forms," Digital Signal Process., Apr. 2018]
- **OLMS-BF** and regular **JO-NLMS** [S. Ciochină et al., "An optimized NLMS algorithm for system identification," Signal Process., 2016]

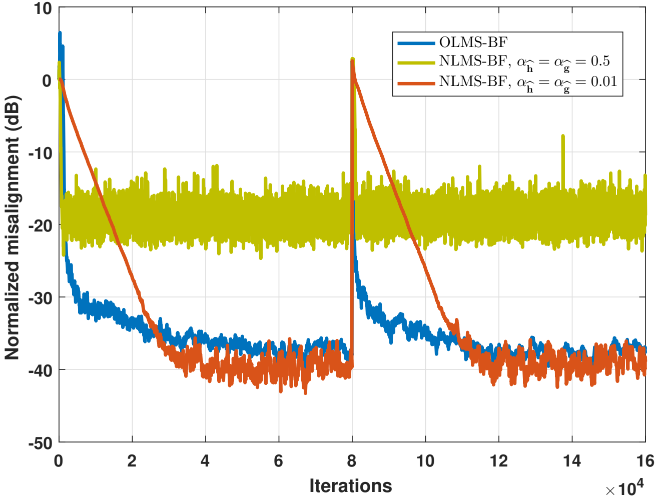

Figure 2: Normalized misalignment for the OLMS-BF and NLMS-BF algorithms, with white Gaussian input signals, $ML = 512$, SNR $= 20$ dB.

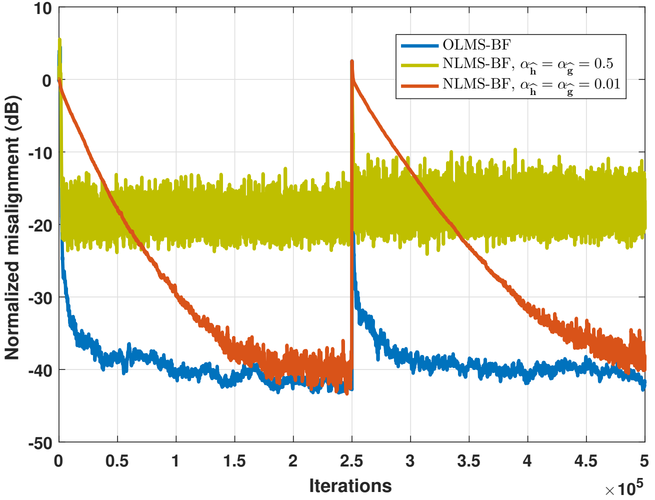

Figure 3: Normalized misalignment for the OLMS-BF and NLMS-BF algorithms, with AR(1) input signals, $ML = 512$, SNR $= 20$ dB.

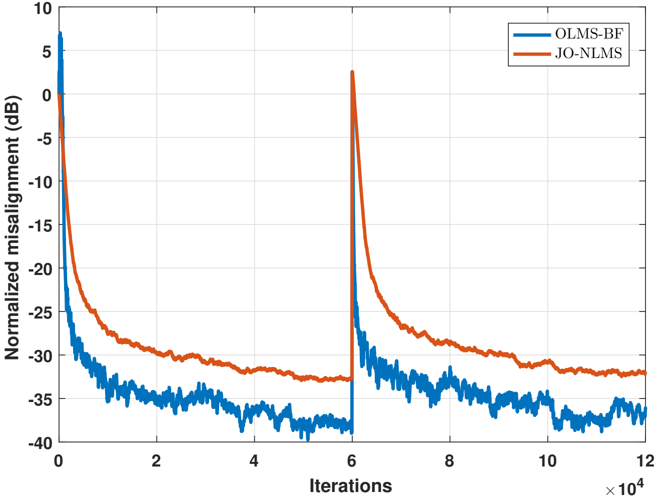

Figure 4: Normalized misalignment for the OLMS-BF and regular JO-NLMS algorithms, with white Gaussian input signals, $ML = 512$, SNR $= 20$ dB.

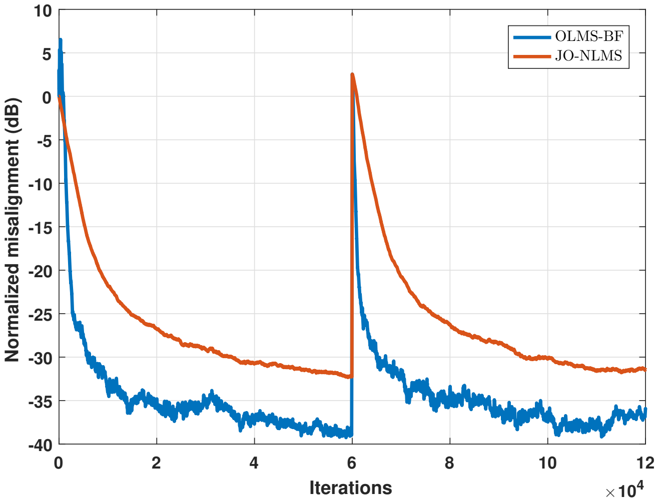

Figure 5: Normalized misalignment for the OLMS-BF and regular JO-NLMS algorithms, with AR(1) input signals, $ML = 512$, SNR $= 20$ dB.

Kalman Filter for Bilinear Forms (KF-BF)

- A posteriori misalignments:

$$\mathbf{c_h}(n) = \frac{1}{\eta}\mathbf{h}(n) - \widehat{\mathbf{h}}(n) \qquad \mathbf{c_g}(n) = \eta\mathbf{g}(n) - \widehat{\mathbf{g}}(n)$$

$\rightarrow$ with correlation matrices:

$$\mathbf{R_{c_h}}(n) = \mathbb{E}[\mathbf{c_h}(n)\mathbf{c_h}^T(n)] \qquad \mathbf{R_{c_g}}(n) = \mathbb{E}[\mathbf{c_g}(n)\mathbf{c_g}^T(n)]$$

- A priori misalignments:

$$\mathbf{c_{h_a}}(n) = \frac{1}{\eta}\mathbf{h}(n) - \widehat{\mathbf{h}}(n-1) = \mathbf{c_h}(n-1) + \frac{1}{\eta}\mathbf{w_h}(n)$$

$$\mathbf{c_{g_a}}(n) = \eta\mathbf{g}(n) - \widehat{\mathbf{g}}(n-1) = \mathbf{c_g}(n-1) + \eta\mathbf{w_g}(n)$$

$\rightarrow$ with correlation matrices:

$$\mathbf{R_{c_{h_a}}}(n) = \mathbb{E}\left[\mathbf{c_{h_a}}(n)\mathbf{c_{h_a}}^T(n)\right] \qquad \mathbf{R_{c_{g_a}}}(n) = \mathbb{E}\left[\mathbf{c_{g_a}}(n)\mathbf{c_{g_a}}^T(n)\right]$$

$$\mathbf{R_{c_{h_a}}}(n) = \mathbf{R_{c_h}}(n-1) + \sigma^2_{w_h}\mathbf{I}_L \qquad \mathbf{R_{c_{g_a}}}(n) = \mathbf{R_{c_g}}(n-1) + \sigma^2_{w_g}\mathbf{I}_M$$

- KF-BF update relations:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mathbf{k}_{\mathbf{h}}(n)e(n) \qquad \widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \mathbf{k}_{\mathbf{g}}(n)e(n)$$

$\mathbf{k}_{\mathbf{h}}(n)$, $\mathbf{k}_{\mathbf{g}}(n)$: Kalman gain vectors

- Minimizing $(1/L)\mathrm{tr}\left[\mathbf{R}_{\mathbf{c}_{\mathbf{h}}}(n)\right]$, $(1/M)\mathrm{tr}\left[\mathbf{R}_{\mathbf{c}_{\mathbf{g}}}(n)\right]$ yields:

$$\mathbf{k}_{\mathbf{h}}(n) = \frac{\mathbf{R}_{\mathbf{c}_{\mathbf{h}_{\mathrm{a}}}}(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n)}{\mathbf{x}_{\widehat{\mathbf{g}}}^T(n)\mathbf{R}_{\mathbf{c}_{\mathbf{h}_{\mathrm{a}}}}(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n)+\sigma_v^2} \qquad \mathbf{k}_{\mathbf{g}}(n) = \frac{\mathbf{R}_{\mathbf{c}_{\mathbf{g}_{\mathrm{a}}}}(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n)}{\mathbf{x}_{\widehat{\mathbf{h}}}^T(n)\mathbf{R}_{\mathbf{c}_{\mathbf{g}_{\mathrm{a}}}}(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n)+\sigma_v^2}$$

- KF-BF update relations:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mathbf{k_h}(n)e(n) \qquad \widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \mathbf{k_g}(n)e(n)$$

$\mathbf{k_h}(n)$, $\mathbf{k_g}(n)$: Kalman gain vectors

- Minimizing $(1/L)\mathrm{tr}\left[\mathbf{R_{c_h}}(n)\right]$, $(1/M)\mathrm{tr}\left[\mathbf{R_{c_g}}(n)\right]$ yields:

$$\mathbf{k_h}(n) = \frac{\mathbf{R}_{\mathbf{c}_{\mathbf{h}_\mathrm{a}}}(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n)}{\mathbf{x}^T_{\widehat{\mathbf{g}}}(n)\mathbf{R}_{\mathbf{c}_{\mathbf{h}_\mathrm{a}}}(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n)+\sigma_v^2} \qquad \mathbf{k_g}(n) = \frac{\mathbf{R}_{\mathbf{c}_{\mathbf{g}_\mathrm{a}}}(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n)}{\mathbf{x}^T_{\widehat{\mathbf{h}}}(n)\mathbf{R}_{\mathbf{c}_{\mathbf{g}_\mathrm{a}}}(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n)+\sigma_v^2}$$

Simplifying assumptions:

- after convergence was reached:

$$\mathbf{R}_{\mathbf{c}_{\mathbf{h}_\mathrm{a}}}(n) \approx \sigma^2_{\mathbf{c}_{\mathbf{h}_\mathrm{a}}}(n)\mathbf{I}_L \qquad \mathbf{R}_{\mathbf{c}_{\mathbf{g}_\mathrm{a}}}(n) \approx \sigma^2_{\mathbf{c}_{\mathbf{g}_\mathrm{a}}}(n)\mathbf{I}_M$$

- misalignments of the individual coefficients: uncorrelated

$\Rightarrow$ we can approximate:

$$\mathbf{I}_L - \mathbf{k_h}(n)\widetilde{\mathbf{x}}^T_{\widehat{\mathbf{g}}}(n) \approx \left[1 - \frac{1}{L}\mathbf{k}^T_{\mathbf{h}}(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)\right]\mathbf{I}_L$$

$$\mathbf{I}_M - \mathbf{k_g}(n)\widetilde{\mathbf{x}}^T_{\widehat{\mathbf{h}}}(n) \approx \left[1 - \frac{1}{M}\mathbf{k}^T_{\mathbf{g}}(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)\right]\mathbf{I}_M$$

$\Rightarrow$ **Simplified Kalman Filter for bilinear forms (SKF - BF)**

- $\mathbf{k_h}(n), \mathbf{k_g}(n)$ - Simplified Kalman gain vectors:

$$\mathbf{k_g}(n) = \mathbf{x}_{\widehat{\mathbf{h}}}(n)\left[\mathbf{x}^T_{\widehat{\mathbf{h}}}(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + \frac{\sigma^2_{v_\mathbf{g}}(n)+\sigma^2_v}{\sigma^2_{\mathbf{c}_{\mathbf{g}_\mathrm{a}}}(n)}\right]^{-1}$$

$$\mathbf{k_h}(n) = \mathbf{x}_{\widehat{\mathbf{g}}}(n)\left[\mathbf{x}^T_{\widehat{\mathbf{g}}}(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + \frac{\sigma^2_{v_\mathbf{h}}(n)+\sigma^2_v}{\sigma^2_{\mathbf{c}_{\mathbf{h}_\mathrm{a}}}(n)}\right]^{-1}$$

- $\mathbf{k_h}(n), \mathbf{k_g}(n)$ - Simplified Kalman gain vectors:

$$\mathbf{k_g}(n) = \mathbf{x}_{\widehat{\mathbf{h}}}(n)\left[\mathbf{x}_{\widehat{\mathbf{h}}}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + \frac{\sigma_{v_\mathbf{g}}^2(n)+\sigma_v^2}{\sigma_{\mathbf{c}_{\mathbf{g}_a}}^2(n)}\right]^{-1}$$

$$\mathbf{k_h}(n) = \mathbf{x}_{\widehat{\mathbf{g}}}(n)\left[\mathbf{x}_{\widehat{\mathbf{g}}}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + \frac{\sigma_{v_\mathbf{h}}^2(n)+\sigma_v^2}{\sigma_{\mathbf{c}_{\mathbf{h}_a}}^2(n)}\right]^{-1}$$

- SKF-BF becomes identical to OLMS-BF if: $p_\mathbf{g} = p_\mathbf{h} = 0$

- $\mathbf{k_h}(n), \mathbf{k_g}(n)$ - Simplified Kalman gain vectors:

$$\mathbf{k_g}(n) = \mathbf{x}_{\widehat{\mathbf{h}}}(n)\left[\mathbf{x}_{\widehat{\mathbf{h}}}^T(n)\mathbf{x}_{\widehat{\mathbf{h}}}(n) + \frac{\sigma_{v_\mathbf{g}}^2(n)+\sigma_v^2}{\sigma_{\mathbf{c}_{\mathbf{g}_a}}^2(n)}\right]^{-1}$$

$$\mathbf{k_h}(n) = \mathbf{x}_{\widehat{\mathbf{g}}}(n)\left[\mathbf{x}_{\widehat{\mathbf{g}}}^T(n)\mathbf{x}_{\widehat{\mathbf{g}}}(n) + \frac{\sigma_{v_\mathbf{h}}^2(n)+\sigma_v^2}{\sigma_{\mathbf{c}_{\mathbf{h}_a}}^2(n)}\right]^{-1}$$

- SKF-BF becomes identical to OLMS-BF if: $p_\mathbf{g} = p_\mathbf{h} = 0$

Practical Considerations

- The parameters related to uncertainties in **h**, **g**: $\sigma_{w_\mathbf{h}}^2$, $\sigma_{w_\mathbf{g}}^2$:

$\rightarrow$ small $\Rightarrow$ good misalignment, poor tracking
$\rightarrow$ large (i.e., high uncertainty in the systems) $\Rightarrow$ good tracking, high misalignment

- In practice $\rightarrow$ some a priori information may be available (e.g., we may consider **g** - time-invariant $\Rightarrow \sigma_{w_\mathbf{g}}^2 = 0$)
- By applying the ℓ_2 norm on the state equation:

$$\widehat{\sigma}_{w_\mathbf{h}}^2(n) = \frac{1}{L}\left\|\widehat{\mathbf{h}}(n) - \widehat{\mathbf{h}}(n-1)\right\|_2^2$$

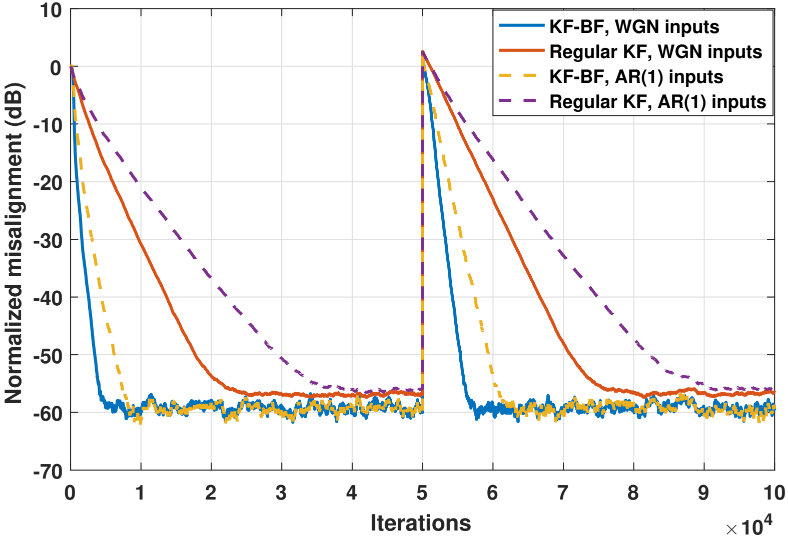

Figure 6: Normalized misalignment of the KF-BF and regular KF for different types of input signals. $ML = 512$, $\sigma_v^2 = 0.01$, $\sigma_{w_{\mathbf{h}}}^2 = \sigma_{w_{\mathbf{g}}}^2 = \sigma_w^2 = 10^{-9}$, and $\epsilon = 10^{-5}$.

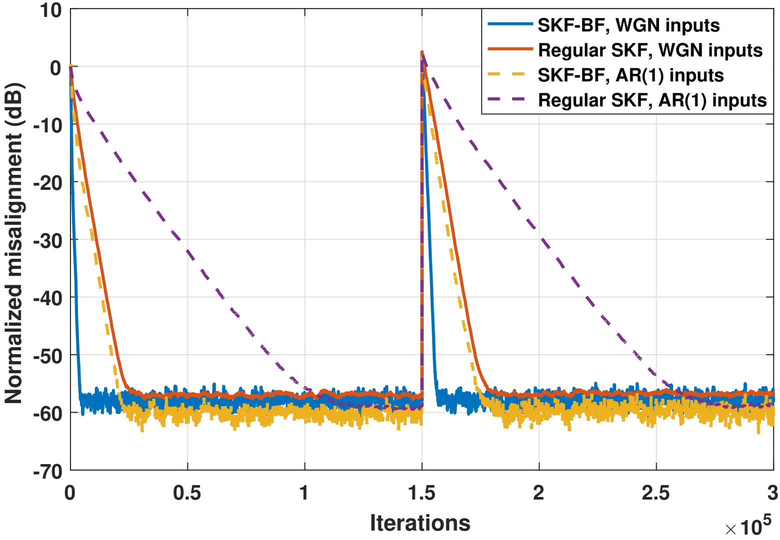

Figure 7: Normalized misalignment of the SKF-BF and regular SKF for different types of input signals. Other conditions are the same as in Fig. 6.

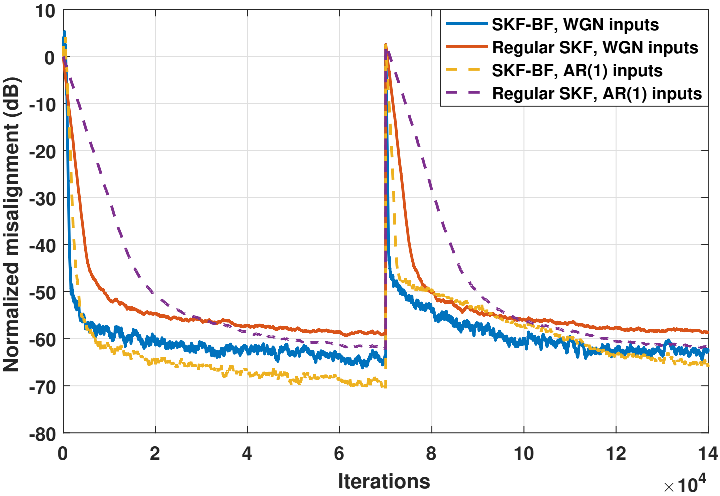

Figure 8: Normalized misalignment of the SKF-BF and regular SKF for different types of input signals, using the recursive estimates $\widehat{\sigma}^2_{w_{\mathbf{h}}}(n)$ and $\widehat{\sigma}^2_{w}(n)$, respectively. $ML = 512$, $\sigma_v^2 = 0.01$, $\sigma^2_{w_{\mathbf{g}}} = 0$, and $\epsilon = 10^{-5}$.

Improved Proportionate APA for the Identification of Sparse Bilinear forms

Motivation:

- **Echo cancellation** - a particular type of system identification problem - estimate a model (echo path) using the available and observed data (usually input and output of the system)
- The echo paths are **sparse** in nature: only a few impulse response components have a significant magnitude, while the rest are zero or small
- **Proportionate** algorithms: adjust the adaptation step-size in proportion to the magnitude of the estimated filter coefficient
- **Affine Projection Algorithm (APA)**: frequently used in echo cancellation, due to its fast convergence

Target: A proportionate APA for the identification of sparse bilinear forms

Improved Proportionate APA for Sparse Bilinear Forms

- **NLMS-BF** [C. Paleologu et al., *Digital Signal Processing*, Apr. 2018]:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\alpha_{\widehat{\mathbf{h}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)e_{\widehat{\mathbf{g}}}(n)}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}^T(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)+\delta_{\widehat{\mathbf{h}}}} \quad \widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \frac{\alpha_{\widehat{\mathbf{g}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)e_{\widehat{\mathbf{h}}}(n)}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}^T(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)+\delta_{\widehat{\mathbf{g}}}}$$

$\rightarrow 0 < \alpha_{\widehat{\mathbf{h}}} < 2$, $0 < \alpha_{\widehat{\mathbf{g}}} < 2$: normalized step-size parameters

$\rightarrow \delta_{\widehat{\mathbf{h}}} > 0$, $\delta_{\widehat{\mathbf{g}}} > 0$: regularization parameters

Improved Proportionate APA for Sparse Bilinear Forms

- **NLMS-BF** [C. Paleologu et al., *Digital Signal Processing*, Apr. 2018]:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\alpha_{\widehat{\mathbf{h}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)e_{\widehat{\mathbf{g}}}(n)}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}^T(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)+\delta_{\widehat{\mathbf{h}}}} \quad \widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \frac{\alpha_{\widehat{\mathbf{g}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)e_{\widehat{\mathbf{h}}}(n)}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}^T(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)+\delta_{\widehat{\mathbf{g}}}}$$

  $\rightarrow 0 < \alpha_{\widehat{\mathbf{h}}} < 2$, $0 < \alpha_{\widehat{\mathbf{g}}} < 2$: normalized step-size parameters
  $\rightarrow \delta_{\widehat{\mathbf{h}}} > 0$, $\delta_{\widehat{\mathbf{g}}} > 0$: regularization parameters

- APA-BF can be seen as a generalization of NLMS-BF

Improved Proportionate APA for Sparse Bilinear Forms

- **NLMS-BF** [C. Paleologu et al., *Digital Signal Processing*, Apr. 2018]:
  $\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\alpha_{\widehat{\mathbf{h}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)e_{\widehat{\mathbf{g}}}(n)}{\widetilde{\mathbf{x}}^T_{\widehat{\mathbf{g}}}(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)+\delta_{\widehat{\mathbf{h}}}}$ $\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \frac{\alpha_{\widehat{\mathbf{g}}}\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)e_{\widehat{\mathbf{h}}}(n)}{\widetilde{\mathbf{x}}^T_{\widehat{\mathbf{h}}}(n)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)+\delta_{\widehat{\mathbf{g}}}}$
  $\rightarrow 0 < \alpha_{\widehat{\mathbf{h}}} < 2$, $0 < \alpha_{\widehat{\mathbf{g}}} < 2$: normalized step-size parameters
  $\rightarrow \delta_{\widehat{\mathbf{h}}} > 0$, $\delta_{\widehat{\mathbf{g}}} > 0$: regularization parameters

- APA-BF can be seen as a generalization of NLMS-BF

- Notations: $\rightarrow \widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}(n) = \begin{bmatrix} \widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n) & \widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n-1) & \cdots & \widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(t-P+1) \end{bmatrix}$
  $\rightarrow \widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}(n) = \begin{bmatrix} \widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n) & \widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n-1) & \cdots & \widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(t-P+1) \end{bmatrix}$
  $\rightarrow \mathbf{d}(n) = \begin{bmatrix} d(n) & d(n-1) & \cdots & d(t-P+1) \end{bmatrix}^T$
  $\rightarrow P$: projection order

- Error signals $\Rightarrow$ error vectors: $\mathbf{e}_{\widehat{\mathbf{g}}}(n) = \mathbf{d}(n) - \widetilde{\mathbf{X}}^T_{\widehat{\mathbf{g}}}(n)\widehat{\mathbf{h}}(n-1)$
  $\mathbf{e}_{\widehat{\mathbf{h}}}(n) = \mathbf{d}(n) - \widetilde{\mathbf{X}}^T_{\widehat{\mathbf{h}}}(n)\widehat{\mathbf{g}}(n-1)$

Improved Proportionate NLMS Algorithm for Bilinear Forms (IPNLMS-BF)

- **IPNLMS-BF:** [C. Paleologu et al., *Proc. IEEE TSP*, 2018]

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \left[\alpha_{\widehat{\mathbf{h}}}\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)e_{\widehat{\mathbf{g}}}(n)\right]\frac{1}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)+\widetilde{\delta}_{\widehat{\mathbf{h}}}}$$

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \left[\alpha_{\widehat{\mathbf{g}}}\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)e_{\widehat{\mathbf{h}}}(n)\right]\frac{1}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)+\widetilde{\delta}_{\widehat{\mathbf{g}}}}$$ where

$\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) = \text{diag}\left[\begin{array}{ccc} q_{\widehat{\mathbf{h}},1}(n-1) & \cdots & q_{\widehat{\mathbf{h}},L}(n-1) \end{array}\right]$ - size $L \times L$

$\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) = \text{diag}\left[\begin{array}{ccc} q_{\widehat{\mathbf{g}},1}(n-1) & \cdots & q_{\widehat{\mathbf{g}},M}(n-1) \end{array}\right]$ - size $M \times M$

Improved Proportionate NLMS Algorithm for Bilinear Forms (IPNLMS-BF)

- **IPNLMS-BF:** [C. Paleologu et al., *Proc. IEEE TSP*, 2018]

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \left[\alpha_{\widehat{\mathbf{h}}}\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)e_{\widehat{\mathbf{g}}}(n)\right]\frac{1}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{g}}}(n)+\widetilde{\delta}_{\widehat{\mathbf{h}}}}$$

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \left[\alpha_{\widehat{\mathbf{g}}}\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)e_{\widehat{\mathbf{h}}}(n)\right]\frac{1}{\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{x}}_{\widehat{\mathbf{h}}}(n)+\widetilde{\delta}_{\widehat{\mathbf{g}}}}$$ where

$\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) = \text{diag}\begin{bmatrix} q_{\widehat{\mathbf{h}},1}(n-1) & \cdots & q_{\widehat{\mathbf{h}},L}(n-1) \end{bmatrix}$ - size $L \times L$

$\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) = \text{diag}\begin{bmatrix} q_{\widehat{\mathbf{g}},1}(n-1) & \cdots & q_{\widehat{\mathbf{g}},M}(n-1) \end{bmatrix}$ - size $M \times M$

$\rightarrow$ Proportionate factors:

$$q_{\widehat{\mathbf{h}},l}(n-1) = \frac{1-\kappa_{\widehat{\mathbf{h}}}}{2L} + (1+\kappa_{\widehat{\mathbf{h}}})\frac{\left|\widehat{h}_l(n-1)\right|}{2\left\|\widehat{\mathbf{h}}(n-1)\right\|_1}, \quad 1 \le l \le L$$

$$q_{\widehat{\mathbf{g}},m}(n-1) = \frac{1-\kappa_{\widehat{\mathbf{g}}}}{2M} + (1+\kappa_{\widehat{\mathbf{g}}})\frac{\left|\widehat{g}_m(n-1)\right|}{2\left\|\widehat{\mathbf{g}}(n-1)\right\|_1}, \quad 1 \le m \le M$$

Improved Proportionate APA for Bilinear Forms

- **IPAPA-BF**:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \alpha_{\widehat{\mathbf{h}}} \mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) \widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}(n) \left[\widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}^{T}(n) \mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) \widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}(n) + \widetilde{\delta}_{\widehat{\mathbf{h}}} \mathbf{I}_P \right]^{-1} \mathbf{e}_{\widehat{\mathbf{g}}}$$

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \alpha_{\widehat{\mathbf{g}}} \mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) \widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}(n) \left[\widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}^{T}(n) \mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) \widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}(n) + \widetilde{\delta}_{\widehat{\mathbf{g}}} \mathbf{I}_P \right]^{-1} \mathbf{e}_{\widehat{\mathbf{h}}}$$

$\rightarrow$ $\mathbf{Q}_{\widehat{\mathbf{h}}}, \mathbf{Q}_{\widehat{\mathbf{g}}}$: matrices containing proportionality factors
$\rightarrow$ if $P = 1 \Rightarrow$ IPNLMS-BF
$\rightarrow$ if $\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) = \mathbf{I}_L$, $\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) = \mathbf{I}_M \Rightarrow$ APA-BF

Improved Proportionate APA for Bilinear Forms

- **IPAPA-BF**:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \alpha_{\widehat{\mathbf{h}}}\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}(n)\left[\widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1)\widetilde{\mathbf{X}}_{\widehat{\mathbf{g}}}(n) + \widetilde{\delta}_{\widehat{\mathbf{h}}}\mathbf{I}_P\right]^{-1}\mathbf{e}_{\widehat{\mathbf{g}}}$$

$$\widehat{\mathbf{g}}(n) = \widehat{\mathbf{g}}(n-1) + \alpha_{\widehat{\mathbf{g}}}\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}(n)\left[\widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}^T(n)\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1)\widetilde{\mathbf{X}}_{\widehat{\mathbf{h}}}(n) + \widetilde{\delta}_{\widehat{\mathbf{g}}}\mathbf{I}_P\right]^{-1}\mathbf{e}_{\widehat{\mathbf{h}}}$$

$\rightarrow$ $\mathbf{Q}_{\widehat{\mathbf{h}}}, \mathbf{Q}_{\widehat{\mathbf{g}}}$: matrices containing proportionality factors
$\rightarrow$ if $P = 1 \Rightarrow$ IPNLMS-BF
$\rightarrow$ if $\mathbf{Q}_{\widehat{\mathbf{h}}}(n-1) = \mathbf{I}_L$, $\mathbf{Q}_{\widehat{\mathbf{g}}}(n-1) = \mathbf{I}_M \Rightarrow$ APA-BF

Experiments - system identification:

- $\mathbf{h}$, of length $L = 512$: the first impulse response from G168 Recommendation, padded with zeros [*Digital Network Echo Cancellers*, ITU-T Recommendations G.168, 2002]
- $\mathbf{g}$, of length $M = 4$: computed as $g_m = 0.5^m, \; m = 1, \ldots, M$

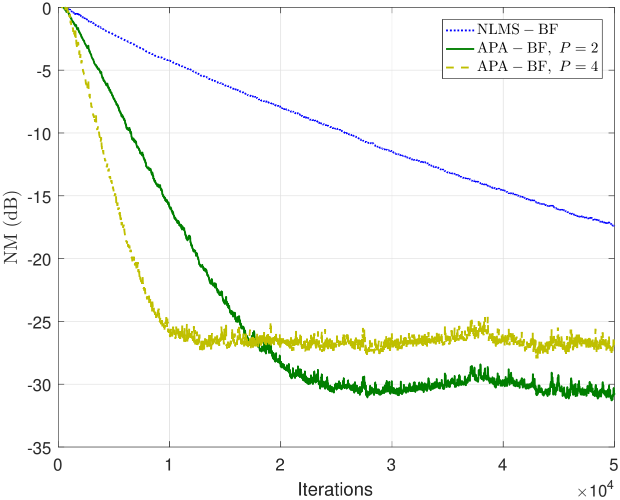

Figure 9: Performance of the NLMS-BF and APA-BF in terms of NM. The input signals are AR(1) processes and $ML = 2048$.

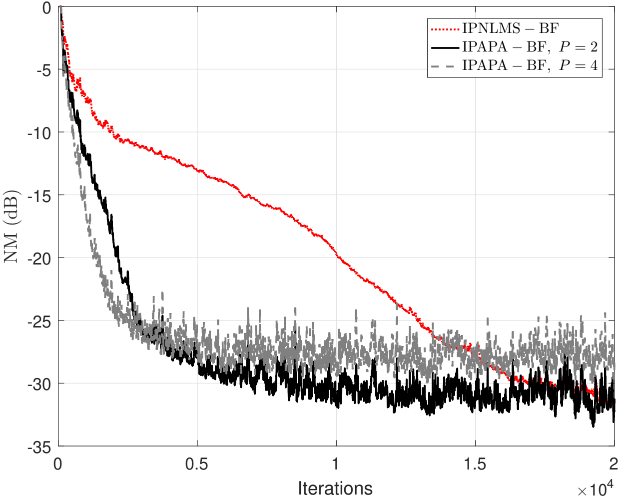

Figure 10: Performance of the IPNLMS-BF and IPAPA-BF in terms of NM. The input signals are AR(1) processes and $ML = 2048$.

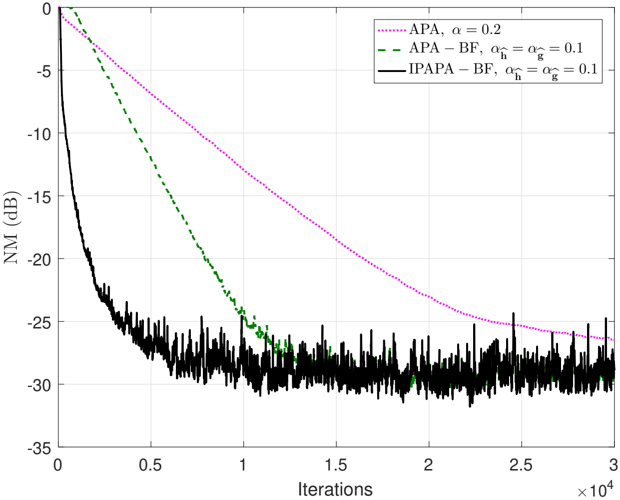

Figure 11: Performance of the APA, APA-BF, and IPAPA-BF in terms of NM. The input signals are white Gaussian noises and $ML = 2048$.

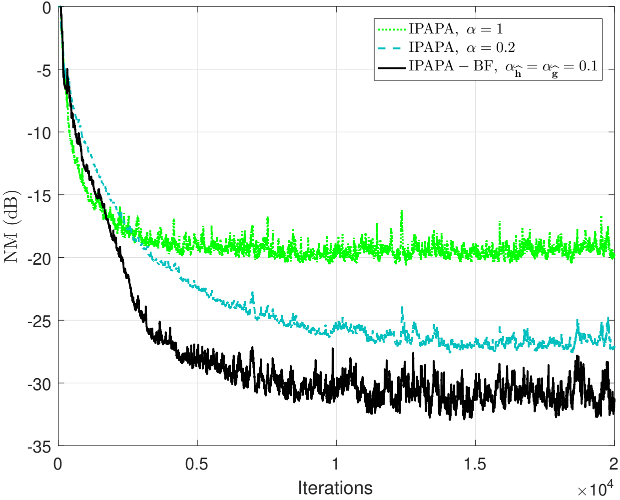

Figure 12: Performance of the IPAPA and IPAPA-BF in terms of NM for different values of the normalized step-size parameters α, $\alpha_{\widehat{\mathbf{h}}}$, and $\alpha_{\widehat{\mathbf{g}}}$. The input signals are AR(1) processes and $ML = 2048$.

Outline

Short Review on Tensors

- **Tensor**: a multidimensional array of data
- Trilinear forms $\Rightarrow$ we only need third-order tensors:

$$\mathcal{A} \in \mathbb{R}^{L_1 \times L_2 \times L_3}, \text{ of dimension } L_1 \times L_2 \times L_3$$

Short Review on Tensors

- **Tensor**: a multidimensional array of data
- Trilinear forms $\Rightarrow$ we only need third-order tensors:

$$\mathcal{A} \in \mathbb{R}^{L_1 \times L_2 \times L_3}, \text{ of dimension } L_1 \times L_2 \times L_3$$

- mode-1 product between tensor $\mathcal{A}$ and matrix $\mathbf{M}_1 \in \mathbb{R}^{M_1 \times L_1}$:

$$\mathcal{U} = \mathcal{A} \times_1 \mathbf{M}_1, \qquad \mathcal{U} \in \mathbb{R}^{M_1 \times L_2 \times L_3},$$

$$u_{m_1 l_2 l_3} = \sum_{l_1=1}^{L_1} a_{l_1 l_2 l_3} m_{m_1 l_1}, \; m_1 = 1, 2, \ldots, M_1$$

- mode-2 product between tensor $\mathcal{A}$ and matrix $\mathbf{M}_2 \in \mathbb{R}^{M_2 \times L_2}$:

$$\mathcal{U} = \mathcal{A} \times_2 \mathbf{M}_2, \qquad \mathcal{U} \in \mathbb{R}^{L_1 \times M_2 \times L_3},$$

$$u_{l_1 m_2 l_3} = \sum_{l_2=1}^{L_2} a_{l_1 l_2 l_3} m_{m_2 l_2}, \; m_2 = 1, 2, \ldots, M_2$$

- mode-3 product between tensor $\mathcal{A}$ and matrix $\mathbf{M}_3 \in \mathbb{R}^{M_3 \times L_3}$:

$$\mathcal{U} = \mathcal{A} \times_3 \mathbf{M}_3, \qquad \mathcal{U} \in \mathbb{R}^{L_1 \times L_2 \times M_3},$$

$$u_{l_1 l_2 m_3} = \sum_{l_3=1}^{L_3} a_{l_1 l_2 l_3} m_{m_3 l_3}, \; m_3 = 1, 2, \ldots, M_3$$

System Model for Trilinear Forms

- **Signal model:**

$$y(t) = \mathcal{X}(t) \times_1 \mathbf{h}_1^T \times_2 \mathbf{h}_2^T \times_3 \mathbf{h}_3^T = \sum_{l_1=1}^{L_1} \sum_{l_2=1}^{L_2} \sum_{l_3=1}^{L_3} x_{l_1 l_2 l_3}(t) h_{1 l_1} h_{2 l_2} h_{3 l_3},$$

where $\mathcal{X}(t) \in \mathbb{R}^{L_1 \times L_2 \times L_3}$: zero-mean input signals,

$$(\mathcal{X})_{l_1 l_2 l_3}(t) = x_{l_1 l_2 l_3}(t),\ l_k = 1, 2, \ldots, L_k,\ k = 1, 2, 3,$$

and $\mathbf{h}_k,\ k = 1, 2, 3$, of lengths L_1, L_2, and L_3: impulse responses

$$\mathbf{h}_k = \begin{bmatrix} h_{k1} & h_{k2} & \cdots & h_{kL_k} \end{bmatrix}^T,\ k = 1, 2, 3.$$

System Model for Trilinear Forms

- **Signal model:**

$$y(t) = \mathcal{X}(t) \times_1 \mathbf{h}_1^T \times_2 \mathbf{h}_2^T \times_3 \mathbf{h}_3^T = \sum_{l_1=1}^{L_1} \sum_{l_2=1}^{L_2} \sum_{l_3=1}^{L_3} x_{l_1 l_2 l_3}(t) h_{1l_1} h_{2l_2} h_{3l_3},$$

where $\mathcal{X}(t) \in \mathbb{R}^{L_1 \times L_2 \times L_3}$: zero-mean input signals,

$$(\mathcal{X})_{l_1 l_2 l_3}(t) = x_{l_1 l_2 l_3}(t),\ l_k = 1, 2, \ldots, L_k,\ k = 1, 2, 3,$$

and $\mathbf{h}_k,\ k = 1, 2, 3$, of lengths L_1, L_2, and L_3: impulse responses

$$\mathbf{h}_k = \begin{bmatrix} h_{k1} & h_{k2} & \cdots & h_{kL_k} \end{bmatrix}^T,\ k = 1, 2, 3.$$

$\rightarrow$ output signal $y(t)$: **trilinear form** with respect to the impulse responses
$\rightarrow$ it can be seen as an extension of the bilinear form [Benesty et al., *IEEE Signal Processing Lett.*, May 2017]

- **Equivalent expression**: $y(t) = \text{vec}^T(\mathcal{H})\,\text{vec}[\mathcal{X}(t)] = \mathbf{h}^T\mathbf{x}(t)$

$$\text{vec}(\mathcal{H}) = \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1 \triangleq \mathbf{h}$$
$$\text{vec}[\mathcal{X}(t)] = \mathbf{x}(t)$$

- **Equivalent expression**: $y(t) = \mathrm{vec}^T(\mathcal{H})\,\mathrm{vec}[\mathcal{X}(t)] = \mathbf{h}^T\mathbf{x}(t)$

$$\mathrm{vec}(\mathcal{H}) = \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1 \triangleq \mathbf{h}$$
$$\mathrm{vec}[\mathcal{X}(t)] = \mathbf{x}(t)$$

- **Goal:** estimation of the global impulse response $\mathbf{h}$
- **Cost function:** $J\left(\widehat{\mathbf{h}}\right) = E\left[e^2(t)\right] = E\left\{\left[d(t) - \widehat{\mathbf{h}}^T\mathbf{x}(t)\right]^2\right\}$

  $\rightarrow \sigma_d^2 = E\left[d^2(t)\right]$: reference signal's variance
  $\rightarrow \mathbf{p} = E[\mathbf{x}(t)d(t)]$: cross-correlation vector between the input and reference signals
  $\rightarrow \mathbf{R} = E\left[\mathbf{x}(t)\mathbf{x}^T(t)\right]$: input signal's covariance matrix

- **Equivalent expression**: $y(t) = \mathrm{vec}^T(\mathcal{H})\,\mathrm{vec}[\mathcal{X}(t)] = \mathbf{h}^T\mathbf{x}(t)$

$$\mathrm{vec}(\mathcal{H}) = \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1 \triangleq \mathbf{h}$$
$$\mathrm{vec}[\mathcal{X}(t)] = \mathbf{x}(t)$$

- **Goal:** estimation of the global impulse response $\mathbf{h}$
- **Cost function:** $J\left(\widehat{\mathbf{h}}\right) = E\left[e^2(t)\right] = E\left\{\left[d(t) - \widehat{\mathbf{h}}^T\mathbf{x}(t)\right]^2\right\}$

  $\rightarrow \sigma_d^2 = E\left[d^2(t)\right]$: reference signal's variance
  $\rightarrow \mathbf{p} = E[\mathbf{x}(t)d(t)]$: cross-correlation vector between the input and reference signals
  $\rightarrow \mathbf{R} = E\left[\mathbf{x}(t)\mathbf{x}^T(t)\right]$: input signal's covariance matrix
- **After computations:** $J\left(\widehat{\mathbf{h}}\right) = \sigma_d^2 - 2\widehat{\mathbf{h}}^T\mathbf{p} + \widehat{\mathbf{h}}^T\mathbf{R}\widehat{\mathbf{h}}$
- **Minimize** $J\left(\widehat{\mathbf{h}}\right)$ $\Rightarrow$ conventional Wiener filter: $\widehat{\mathbf{h}}_{\mathrm{W}} = \mathbf{R}^{-1}\mathbf{p}$

Iterative Wiener Filter for Trilinear Forms

- **Problems** of the conventional Wiener filter:

$\rightarrow$ $\mathbf{R}$: size $L_1L_2L_3 \times L_1L_2L_3 \Rightarrow$ huge amount of data for its estimation
$\rightarrow$ $\mathbf{R}$ could be very ill-conditioned, due to its huge size
$\rightarrow$ the solution $\widehat{\mathbf{h}}_{\mathrm{W}}$ could be very inaccurate in practice

- **Idea**: $\mathbf{h}$ ($L_1L_2L_3$ coefficients) is obtained through a combination of $\mathbf{h}_k,\ k = 1, 2, 3$, with L_1, L_2, and L_3 coefficients
  $\rightarrow L_1 + L_2 + L_3$ different elements are enough to form $\mathbf{h}$, not $L_1L_2L_3$
- **Solution**: an iterative version of the Wiener filter

Iterative Wiener Filter for Trilinear Forms

- **Problems** of the conventional Wiener filter:

→ $\mathbf{R}$: size $L_1L_2L_3 \times L_1L_2L_3 \Rightarrow$ huge amount of data for its estimation
→ $\mathbf{R}$ could be very ill-conditioned, due to its huge size
→ the solution $\widehat{\mathbf{h}}_{\mathrm{W}}$ could be very inaccurate in practice

- **Idea**: $\mathbf{h}$ ($L_1L_2L_3$ coefficients) is obtained through a combination of $\mathbf{h}_k,\ k = 1, 2, 3$, with L_1, L_2, and L_3 coefficients
  → $L_1 + L_2 + L_3$ different elements are enough to form $\mathbf{h}$, not $L_1L_2L_3$
- **Solution**: an iterative version of the Wiener filter

−$\widehat{\mathbf{h}}$ can be decomposed as:

$$\begin{aligned}
\widehat{\mathbf{h}} &= \widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1, \\
&= \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)\widehat{\mathbf{h}}_1 \\
&= \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)\widehat{\mathbf{h}}_2 \\
&= \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)\widehat{\mathbf{h}}_3
\end{aligned}$$

Iterative Wiener Filter for Trilinear Forms

- **Problems** of the conventional Wiener filter:

$\rightarrow$ $\mathbf{R}$: size $L_1L_2L_3 \times L_1L_2L_3 \Rightarrow$ huge amount of data for its estimation
$\rightarrow$ $\mathbf{R}$ could be very ill-conditioned, due to its huge size
$\rightarrow$ the solution $\widehat{\mathbf{h}}_{\mathrm{W}}$ could be very inaccurate in practice

- **Idea**: $\mathbf{h}$ ($L_1L_2L_3$ coefficients) is obtained through a combination of $\mathbf{h}_k,\ k=1,2,3$, with L_1, L_2, and L_3 coefficients
  $\rightarrow L_1+L_2+L_3$ different elements are enough to form $\mathbf{h}$, not $L_1L_2L_3$
- **Solution**: an iterative version of the Wiener filter

$-\widehat{\mathbf{h}}$ can be decomposed as:

$$
\begin{aligned}
\widehat{\mathbf{h}} &= \widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1, \\
&= \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)\widehat{\mathbf{h}}_1 \\
&= \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)\widehat{\mathbf{h}}_2 \\
&= \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)\widehat{\mathbf{h}}_3
\end{aligned}
$$

$-$ in a corresponding manner, $J\left(\widehat{\mathbf{h}}\right)$ can be written as:

$$
\begin{aligned}
J_{\widehat{\mathbf{h}}_2,\widehat{\mathbf{h}}_3}\left(\widehat{\mathbf{h}}_1\right) &= \sigma_d^2 - 2\widehat{\mathbf{h}}_1^T\mathbf{p}_1 + \widehat{\mathbf{h}}_1^T\mathbf{R}_1\widehat{\mathbf{h}}_1 \\
J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_3}\left(\widehat{\mathbf{h}}_2\right) &= \sigma_d^2 - 2\widehat{\mathbf{h}}_2^T\mathbf{p}_2 + \widehat{\mathbf{h}}_2^T\mathbf{R}_2\widehat{\mathbf{h}}_2 \\
J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_2}\left(\widehat{\mathbf{h}}_3\right) &= \sigma_d^2 - 2\widehat{\mathbf{h}}_3^T\mathbf{p}_3 + \widehat{\mathbf{h}}_3^T\mathbf{R}_3\widehat{\mathbf{h}}_3
\end{aligned}
$$

Iterative Wiener Filter for Trilinear Forms

where

$$\mathbf{p}_1 = \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)^T \mathbf{p},$$

$$\mathbf{R}_1 = \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right),$$

$$\mathbf{p}_2 = \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{p},$$

$$\mathbf{R}_2 = \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right),$$

$$\mathbf{p}_3 = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{p},$$

$$\mathbf{R}_3 = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right).$$

Iterative Wiener Filter for Trilinear Forms

where

$$\mathbf{p}_1 = \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)^T \mathbf{p},$$

$$\mathbf{R}_1 = \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3 \otimes \widehat{\mathbf{h}}_2 \otimes \mathbf{I}_{L_1}\right),$$

$$\mathbf{p}_2 = \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{p},$$

$$\mathbf{R}_2 = \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3 \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1\right),$$

$$\mathbf{p}_3 = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{p},$$

$$\mathbf{R}_3 = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2 \otimes \widehat{\mathbf{h}}_1\right).$$

- **Initialize**:

$$\widehat{\mathbf{h}}_2^{(0)} = (1/L_2) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T$$
$$\widehat{\mathbf{h}}_3^{(0)} = (1/L_3) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T$$

- **Compute**: $\mathbf{p}_1^{(0)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)^T \mathbf{p}$

$$\mathbf{R}_1^{(0)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_2,\widehat{\mathbf{h}}_3}\left(\widehat{\mathbf{h}}_1^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}_1^{(0)} + \left(\widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R}_1^{(0)}\widehat{\mathbf{h}}_1^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_1^{(1)} = \left(\mathbf{R}_1^{(0)}\right)^{-1} \mathbf{p}_1^{(0)}$$

- **Compute**: $\mathbf{p}_1^{(0)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)^T \mathbf{p}$

$$\mathbf{R}_1^{(0)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \mathbf{I}_{L_1}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_2,\widehat{\mathbf{h}}_3}\left(\widehat{\mathbf{h}}_1^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}_1^{(0)} + \left(\widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R}_1^{(0)} \widehat{\mathbf{h}}_1^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_1^{(1)} = \left(\mathbf{R}_1^{(0)}\right)^{-1} \mathbf{p}_1^{(0)}$$

- **Compute**: $\mathbf{p}_2^{(1)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}$

$$\mathbf{R}_2^{(1)} = \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3^{(0)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_3}\left(\widehat{\mathbf{h}}_2^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_2^{(1)}\right)^T \mathbf{p}_2^{(1)} + \left(\widehat{\mathbf{h}}_2^{(1)}\right)^T \mathbf{R}_2^{(1)} \widehat{\mathbf{h}}_2^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_2^{(1)} = \left(\mathbf{R}_2^{(1)}\right)^{-1} \mathbf{p}_2^{(1)}$$

- **Compute**: $\mathbf{p}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}$

$$\mathbf{R}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_2}\left(\widehat{\mathbf{h}}_3^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{p}_3^{(1)} + \left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{R}_3^{(1)} \widehat{\mathbf{h}}_3^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_3^{(1)} = \left(\mathbf{R}_3^{(1)}\right)^{-1} \mathbf{p}_3^{(1)}$$

- **Compute:** $\mathbf{p}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}$

$$\mathbf{R}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_2}\left(\widehat{\mathbf{h}}_3^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{p}_3^{(1)} + \left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{R}_3^{(1)}\widehat{\mathbf{h}}_3^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_3^{(1)} = \left(\mathbf{R}_3^{(1)}\right)^{-1} \mathbf{p}_3^{(1)}$$

- **At iteration** n:

$$\widehat{\mathbf{h}}_1^{(n)} = \left(\mathbf{R}_1^{(n-1)}\right)^{-1} \mathbf{p}_1^{(n-1)}, \quad \mathbf{p}_2^{(n)} = \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{p},$$

$$\mathbf{R}_2^{(n)} = \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right),$$

$$\widehat{\mathbf{h}}_2^{(n)} = \left(\mathbf{R}_2^{(n)}\right)^{-1} \mathbf{p}_2^{(n)}, \quad \mathbf{p}_3^{(n)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{p},$$

$$\mathbf{R}_3^{(n)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right),$$

$$\widehat{\mathbf{h}}_3^{(n)} = \left(\mathbf{R}_3^{(n)}\right)^{-1} \mathbf{p}_3^{(n)}.$$

- **Compute**: $\mathbf{p}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}$

$$\mathbf{R}_3^{(1)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(1)} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)$$

- **Minimize** $J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_2}\left(\widehat{\mathbf{h}}_3^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{p}_3^{(1)} + \left(\widehat{\mathbf{h}}_3^{(1)}\right)^T \mathbf{R}_3^{(1)} \widehat{\mathbf{h}}_3^{(1)}$

$$\Rightarrow \widehat{\mathbf{h}}_3^{(1)} = \left(\mathbf{R}_3^{(1)}\right)^{-1} \mathbf{p}_3^{(1)}$$

- **At iteration** n:

$$\widehat{\mathbf{h}}_1^{(n)} = \left(\mathbf{R}_1^{(n-1)}\right)^{-1} \mathbf{p}_1^{(n-1)}, \quad \mathbf{p}_2^{(n)} = \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{p},$$

$$\mathbf{R}_2^{(n)} = \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{R} \left(\widehat{\mathbf{h}}_3^{(n-1)} \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(n)}\right),$$

$$\widehat{\mathbf{h}}_2^{(n)} = \left(\mathbf{R}_2^{(n)}\right)^{-1} \mathbf{p}_2^{(n)}, \quad \mathbf{p}_3^{(n)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{p},$$

$$\mathbf{R}_3^{(n)} = \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right)^T \mathbf{R} \left(\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}\right),$$

$$\widehat{\mathbf{h}}_3^{(n)} = \left(\mathbf{R}_3^{(n)}\right)^{-1} \mathbf{p}_3^{(n)}.$$

- **Finally**: $\widehat{\mathbf{h}}^{(n)} = \widehat{\mathbf{h}}_3^{(n)} \otimes \widehat{\mathbf{h}}_2^{(n)} \otimes \widehat{\mathbf{h}}_1^{(n)}$

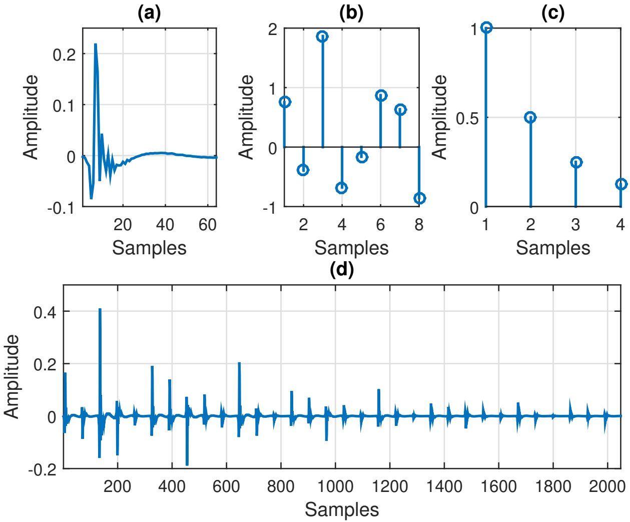

Figure 13: Impulse responses used in simulations: (a) $\mathbf{h}_1$ of length $L_1 = 64$ [*Digital Network Echo Cancellers*, ITU-T Recommendations G.168, 2002.], (b) $\mathbf{h}_2$ of length $L_2 = 8$ (randomly generated), (c) $\mathbf{h}_3$ of length $L_3 = 4$ (evaluated as $h_{3l_3} = 0.5^{l_3-1}$, $l_3 = 1, \ldots, L_3$), (d) global impulse response $\mathbf{h} = \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1$ of length $L = L_1 L_2 L_3 = 2048$.

- N data samples available to estimate **R** and **p**

$$\widehat{\mathbf{R}} = \frac{1}{N}\sum_{t=1}^{N}\mathbf{x}(t)\mathbf{x}^T(t) \qquad \widehat{\mathbf{p}} = \frac{1}{N}\sum_{t=1}^{N}\mathbf{x}(t)d(t)$$

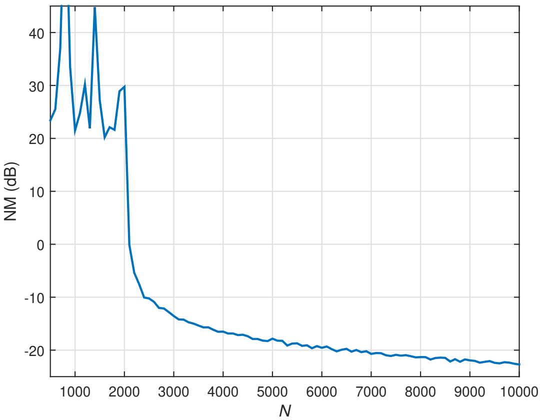

Figure 14: Normalized misalignment of the conventional Wiener filter as a function of N (available data samples to estimate the statistics), for the identification of **h**.

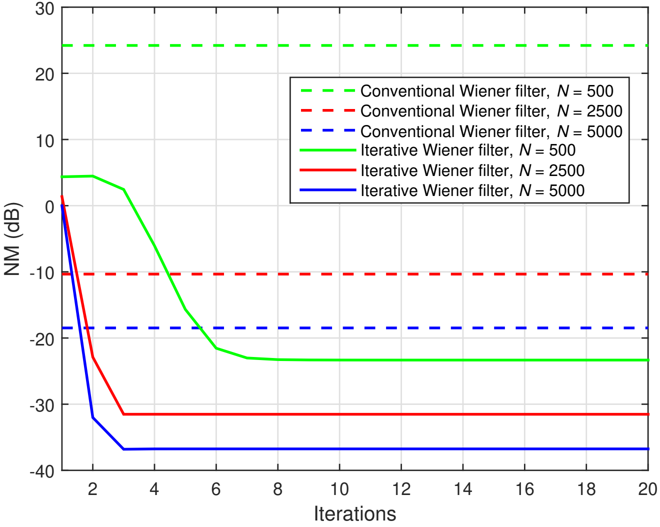

Figure 15: Normalized misalignment of the conventional and iterative Wiener filters, for different values of N (available data samples to estimate the statistics), for the identification of $\mathbf{h}$.

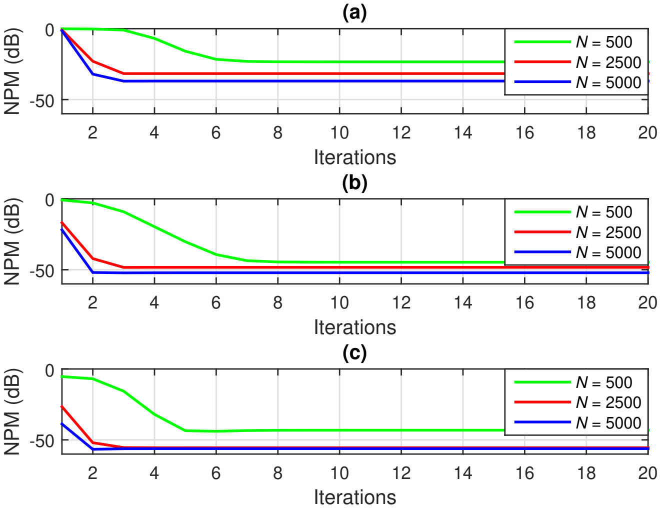

Figure 16: Normalized projection misalignment of the iterative Wiener filter, for different values of N (available data samples to estimate the statistics), for the identification of $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$

Iterative Wiener Filter for Trilinear Forms

- The proposed approach offers:
  - **Lower computational complexity**: a high-dimension system identification problem of size $L_1 L_2 L_3$ is translated in low-dimension problems of sizes L_1, L_2, and L_3, tensorized together
  - **A more accurate solution**, especially when a small amount of data is available to estimate the statistics $\Rightarrow$ advantage in case of incomplete data sets, under-modeling cases, and very ill-conditioned problems

Iterative Wiener Filter for Trilinear Forms

- The proposed approach offers:
  - **Lower computational complexity**: a high-dimension system identification problem of size $L_1 L_2 L_3$ is translated in low-dimension problems of sizes L_1, L_2, and L_3, tensorized together
  - **A more accurate solution**, especially when a small amount of data is available to estimate the statistics $\Rightarrow$ advantage in case of incomplete data sets, under-modeling cases, and very ill-conditioned problems
- Limitations of the Wiener filter:
  - matrix inversion operation
  - correlation matrix estimation
  - unsuitable in real-world scenarios (e.g., nonstationary environments and/or requiring real-time processing)
- **Solution**: LMS-based algorithms for the identification of trilinear forms

Least-Mean-Square Algorithm for Trilinear Forms (LMS-TF)

- A priori error signal can be written (similar to BF) as:

$$
\begin{aligned}
e(t) = d(t) - \widehat{y}(t) &= d(t) - \widehat{\mathbf{h}}(t-1)^T \mathbf{x}(t) \\
&= d(t) - \widehat{\mathbf{h}}_1^T(t-1)\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) \quad \leftarrow e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) \\
&= d(t) - \widehat{\mathbf{h}}_2^T(t-1)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) \quad \leftarrow e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) \\
&= d(t) - \widehat{\mathbf{h}}_3^T(t-1)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) \quad \leftarrow e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) &= \left[\widehat{\mathbf{h}}_3(t-1) \otimes \widehat{\mathbf{h}}_2(t-1) \otimes \mathbf{I}_{L_1}\right] \mathbf{x}(t) \\
\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) &= \left[\widehat{\mathbf{h}}_3(t-1) \otimes \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_1(t-1)\right] \mathbf{x}(t) \\
\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) &= \left[\mathbf{I}_{L_3} \otimes \widehat{\mathbf{h}}_2(t-1) \otimes \widehat{\mathbf{h}}_1(t-1)\right] \mathbf{x}(t)
\end{aligned}
$$

Least-Mean-Square Algorithm for Trilinear Forms (LMS-TF)

- LMS-TF updates:

$$\widehat{\mathbf{h}}_1(t) = \widehat{\mathbf{h}}_1(t-1) + \mu_{\widehat{\mathbf{h}}_1} \mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)$$
$$\widehat{\mathbf{h}}_2(t) = \widehat{\mathbf{h}}_2(t-1) + \mu_{\widehat{\mathbf{h}}_2} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)$$
$$\widehat{\mathbf{h}}_3(t) = \widehat{\mathbf{h}}_3(t-1) + \mu_{\widehat{\mathbf{h}}_3} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)$$

$\rightarrow \mu_{\widehat{\mathbf{h}}_1} > 0, \mu_{\widehat{\mathbf{h}}_2} > 0, \mu_{\widehat{\mathbf{h}}_3} > 0$: step-size parameters

Least-Mean-Square Algorithm for Trilinear Forms (LMS-TF)

- LMS-TF updates:

$$\widehat{\mathbf{h}}_1(t) = \widehat{\mathbf{h}}_1(t-1) + \mu_{\widehat{\mathbf{h}}_1} \mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)$$
$$\widehat{\mathbf{h}}_2(t) = \widehat{\mathbf{h}}_2(t-1) + \mu_{\widehat{\mathbf{h}}_2} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)$$
$$\widehat{\mathbf{h}}_3(t) = \widehat{\mathbf{h}}_3(t-1) + \mu_{\widehat{\mathbf{h}}_3} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)$$

$\rightarrow \mu_{\widehat{\mathbf{h}}_1} > 0, \mu_{\widehat{\mathbf{h}}_2} > 0, \mu_{\widehat{\mathbf{h}}_3} > 0$: step-size parameters

- LMS-TF uses three short filters, of lengths L_1, L_2, L_3, instead of a long filter, of length $L_1 L_2 L_3 \Rightarrow$ lower complexity
- Faster convergence rate expected
- For non-stationary signals: it may be more appropriate to use **time-dependent step-sizes** $\mu_{\widehat{\mathbf{h}}_1}(t), \mu_{\widehat{\mathbf{h}}_2}(t), \mu_{\widehat{\mathbf{h}}_3}(t)$

Normalized LMS Algorithm for Trilinear Forms (NLMS-TF)

- A posteriori error signals:

$$
\begin{aligned}
\varepsilon_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) &= d(t) - \widehat{\mathbf{h}}_1^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) \\
\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) &= d(t) - \widehat{\mathbf{h}}_2^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) \\
\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) &= d(t) - \widehat{\mathbf{h}}_3^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)
\end{aligned}
$$

Normalized LMS Algorithm for Trilinear Forms (NLMS-TF)

- A posteriori error signals:

$$\varepsilon_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) = d(t) - \widehat{\mathbf{h}}_1^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)$$
$$\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) = d(t) - \widehat{\mathbf{h}}_2^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)$$
$$\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) = d(t) - \widehat{\mathbf{h}}_3^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)$$

- By cancelling the a posteriori error signals $\Rightarrow$ NLMS-TF:

$$\widehat{\mathbf{h}}_1(t) = \widehat{\mathbf{h}}_1(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_1}\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)}{\mathbf{x}^T_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t)\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3}(t) + \delta_{\widehat{\mathbf{h}}_1}}$$
$$\widehat{\mathbf{h}}_2(t) = \widehat{\mathbf{h}}_2(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_2}\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)}{\mathbf{x}^T_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3}(t) + \delta_{\widehat{\mathbf{h}}_2}}$$
$$\widehat{\mathbf{h}}_3(t) = \widehat{\mathbf{h}}_3(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_3}\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)}{\mathbf{x}^T_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2}(t) + \delta_{\widehat{\mathbf{h}}_3}}$$

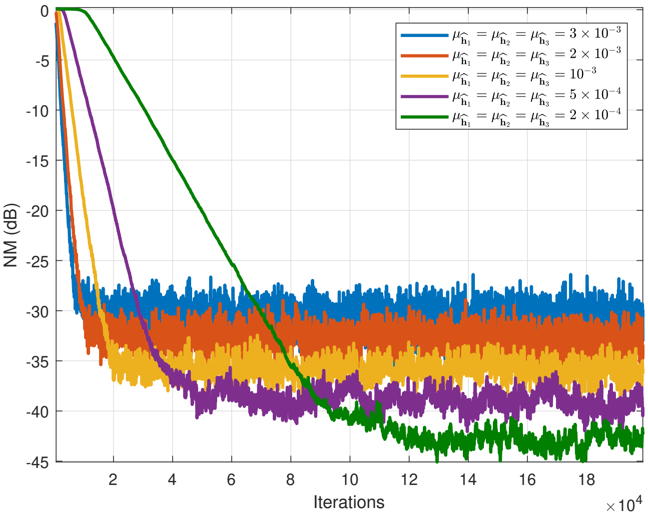

Figure 17: Normalized misalignment of the LMS-TF algorithm using different values of the step-size parameters.

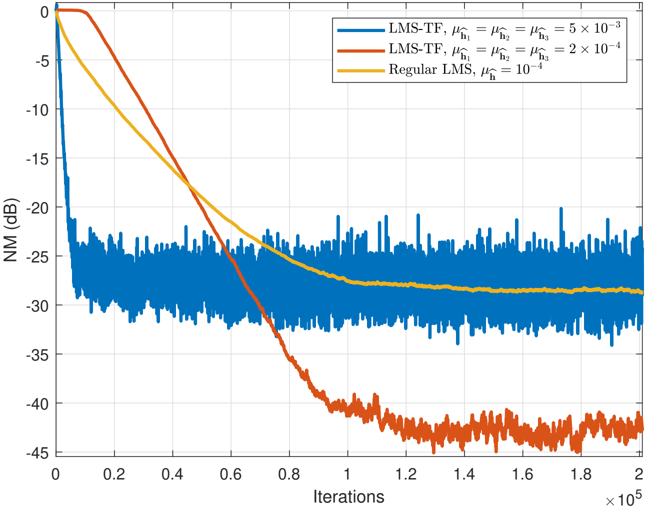

Figure 18: Normalized misalignment of the LMS-TF and regular LMS algorithms.

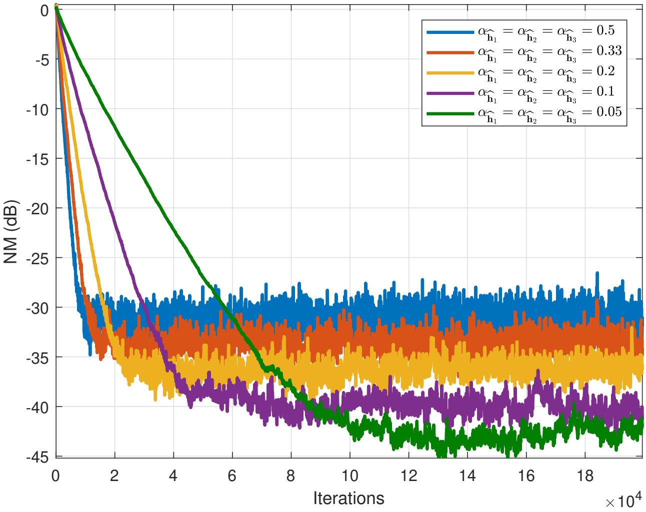

Figure 19: Normalized misalignment of the NLMS-TF algorithm using different values of the step-size parameters.

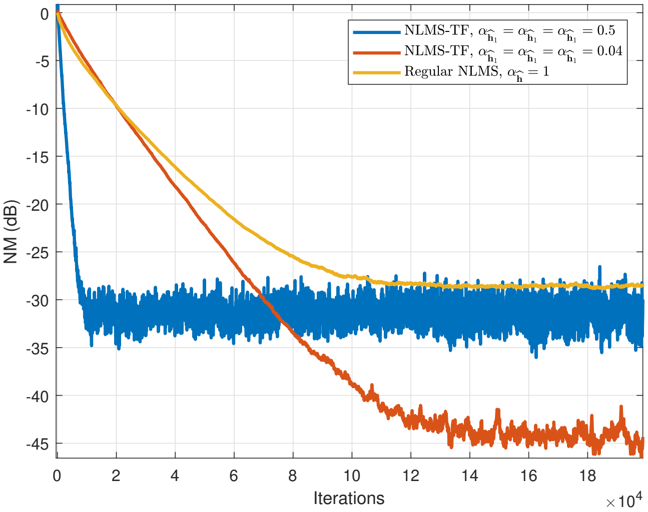

Figure 20: Normalized misalignment of the NLMS-TF and regular NLMS algorithms.

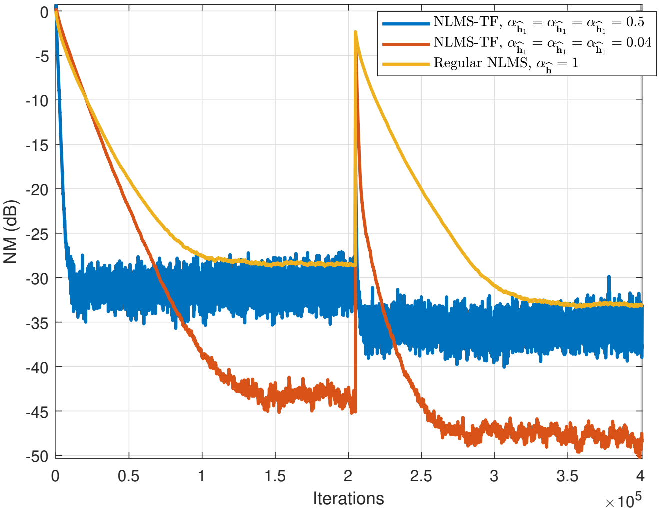

Figure 21: Normalized misalignment of the NLMS-TF and regular NLMS algorithms. The impulse response $\mathbf{h}_2$ changes in the middle of the experiment.

Outline

Iterative Wiener Filter for Multilinear Forms

- **Idea**: $\mathbf{f}$ (with $L_1 L_2 \times \cdots \times L_N$ coefficients) is obtained through a combination of $\mathbf{h}_k, \ k = 1, 2, \ldots, N$, with L_1, $L_2, \ldots, L_N$ coefficients $\rightarrow L_1 + L_2 + \cdots + L_N$ different elements are enough to form $\mathbf{f}$
- **Solution**: an iterative version of the Wiener filter

Iterative Wiener Filter for Multilinear Forms

- **Idea**: $\mathbf{f}$ (with $L_1 L_2 \times \cdots \times L_N$ coefficients) is obtained through a combination of $\mathbf{h}_k,\ k = 1, 2, \ldots, N$, with $L_1, L_2, \ldots, L_N$ coefficients $\rightarrow L_1 + L_2 + \cdots + L_N$ different elements are enough to form $\mathbf{f}$
- **Solution**: an iterative version of the Wiener filter

  $\rightarrow$ It can be verified that:

$$
\begin{aligned}
\mathbf{f} &= \mathbf{h}_N \otimes \mathbf{h}_{N-1} \otimes \cdots \otimes \mathbf{h}_1 \\
&= \left(\mathbf{h}_N \otimes \mathbf{h}_{N-1} \otimes \cdots \otimes \mathbf{I}_{L_1}\right) \mathbf{h}_1 \\
&= \left(\mathbf{h}_N \otimes \mathbf{h}_{N-1} \otimes \cdots \otimes \mathbf{h}_3 \otimes \mathbf{I}_{L_2} \otimes \mathbf{h}_1\right) \mathbf{h}_2 \\
&\ \vdots \\
&= \left(\mathbf{h}_N \otimes \mathbf{h}_{N-1} \otimes \cdots \otimes \mathbf{I}_{L_i} \otimes \mathbf{h}_{L_i - 1} \otimes \cdots \otimes \mathbf{h}_1\right) \mathbf{h}_i \\
&\ \vdots \\
&= \left(\mathbf{I}_{L_N} \otimes \mathbf{h}_{N-1} \otimes \cdots \otimes \mathbf{h}_1\right) \mathbf{h}_N
\end{aligned}
$$

Iterative Wiener Filter for Multilinear Forms

- Consequently, $J\left(\widehat{\mathbf{f}}\right)$ can be written in N equivalent forms
- When all coefficients except $\widehat{\mathbf{h}}_i$ are fixed:

$$J_{\widehat{\mathbf{h}}_1,\widehat{\mathbf{h}}_2,\ldots,\widehat{\mathbf{h}}_{i-1},\widehat{\mathbf{h}}_{i+1},\ldots,\widehat{\mathbf{h}}_N}\left(\widehat{\mathbf{h}}_i\right) = \sigma_d^2 - 2\widehat{\mathbf{h}}_i^T\mathbf{p}_i + \widehat{\mathbf{h}}_i^T\mathbf{R}_i\widehat{\mathbf{h}}_i, \quad i = 1, 2, \ldots, N$$

where

$$\rightarrow \mathbf{p}_i = \left(\widehat{\mathbf{h}}_N \otimes \widehat{\mathbf{h}}_{N-1} \otimes \cdots \otimes \mathbf{I}_{L_i} \otimes \widehat{\mathbf{h}}_{L_i-1} \otimes \cdots \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{p}$$

$$\rightarrow \mathbf{R}_i = \left(\widehat{\mathbf{h}}_N \otimes \widehat{\mathbf{h}}_{N-1} \otimes \cdots \otimes \mathbf{I}_{L_i} \otimes \widehat{\mathbf{h}}_{L_i-1} \otimes \cdots \otimes \widehat{\mathbf{h}}_1\right)^T \mathbf{R} \times \left(\widehat{\mathbf{h}}_N \otimes \widehat{\mathbf{h}}_{N-1} \otimes \cdots \otimes \mathbf{I}_{L_i} \otimes \widehat{\mathbf{h}}_{L_i-1} \otimes \cdots \otimes \widehat{\mathbf{h}}_1\right)$$

- $\widehat{\mathbf{h}}_i = \mathbf{R}_i^{-1}\mathbf{p}_i, \; i = 1, 2, \ldots, N$

Iterative Wiener Filter for Multilinear Forms

→ Initialization: a set of initial values $\widehat{\mathbf{h}}_i^{(0)},\ i = 1, 2, \ldots, N$

→ Computations:

$$\mathbf{p}_1^{(0)} = \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \widehat{\mathbf{I}}_{L_1} \right)^T \mathbf{p}$$

$$\mathbf{R}_1^{(0)} = \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \widehat{\mathbf{I}}_{L_1} \right)^T \mathbf{R} \times \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_2^{(0)} \otimes \widehat{\mathbf{I}}_{L_1} \right)$$

→ Cost function:

$$J_{\widehat{\mathbf{h}}_2, \widehat{\mathbf{h}}_3, \ldots, \widehat{\mathbf{h}}_N} \left(\widehat{\mathbf{h}}_1^{(1)} \right) = \sigma_d^2 - 2 \left(\widehat{\mathbf{h}}_1^{(1)} \right)^T \mathbf{p}_1^{(0)} + \left(\widehat{\mathbf{h}}_1^{(1)} \right)^T \mathbf{R}_1^{(0)} \left(\widehat{\mathbf{h}}_1^{(1)} \right)$$

→ After minimization of the cost function:

$$\widehat{\mathbf{h}}_1^{(1)} = \left(\mathbf{R}_1^{(0)} \right)^{-1} \mathbf{p}_1^{(0)}$$

→ Computations:

$$\mathbf{p}_2^{(1)} = \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{I}}_{L_2} \otimes \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{p}$$

$$\mathbf{R}_2^{(1)} = \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{I}}_{L_2} \widehat{\mathbf{h}}_1^{(1)}\right)^T \mathbf{R} \times \left(\widehat{\mathbf{h}}_N^{(0)} \otimes \widehat{\mathbf{h}}_{N-1}^{(0)} \otimes \cdots \otimes \widehat{\mathbf{h}}_3^{(0)} \otimes \widehat{\mathbf{I}}_{L_2} \widehat{\mathbf{h}}_1^{(1)}\right)$$

→ Cost function:

$$J_{\widehat{\mathbf{h}}_1, \widehat{\mathbf{h}}_3, \ldots, \widehat{\mathbf{h}}_N}\left(\widehat{\mathbf{h}}_2^{(1)}\right) = \sigma_d^2 - 2\left(\widehat{\mathbf{h}}_2^{(1)}\right)^T \mathbf{p}_2^{(1)} + \left(\widehat{\mathbf{h}}_2^{(1)}\right)^T \mathbf{R}_2^{(1)} \left(\widehat{\mathbf{h}}_2^{(1)}\right)$$

→ After minimization of the cost function:

$$\widehat{\mathbf{h}}_2^{(1)} = \left(\mathbf{R}_2^{(1)}\right)^{-1} \mathbf{p}_2^{(1)}$$

→ Similarly, we compute all $\widehat{\mathbf{h}}_i^{(1)}, \; i = 1, 2, \ldots, N$

→ Continuing up to iteration n, we get the estimates of the N vectors

Simulation Setup

- input signals - independent AR(1), obtained by filtering WGN signals through a first-order system $1/\left(1-0.9z^{-1}\right)$
- $w(n)$ - AWGN, with variance $\sigma_w^2 = 0.01$

Simulation Setup

- input signals - independent AR(1), obtained by filtering WGN signals through a first-order system $1/\left(1 - 0.9z^{-1}\right)$
- $w(n)$ - AWGN, with variance $\sigma_w^2 = 0.01$

- Performance measures:

→ Normalized projection misalignment (NPM) [Morgan et al., *IEEE Signal Processing Letters*, July 1998]:

$$\text{NPM}[\mathbf{h}_i, \widehat{\mathbf{h}}_i] = 1 - \left[\frac{\mathbf{h}_i^T \widehat{\mathbf{h}}_i}{\|\mathbf{h}_i(n)\| \|\widehat{\mathbf{h}}_i\|}\right]^2, \; i = 1, 2, \ldots, N$$

→ Normalized misalignment (NM):

$$\text{NM}[\mathbf{f}, \widehat{\mathbf{f}}] = \frac{\|\mathbf{f} - \widehat{\mathbf{f}}\|^2}{\|\mathbf{f}\|^2}$$

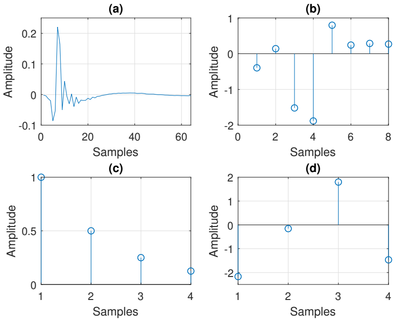

Figure 22: Impulse responses used in simulations: (a) $\mathbf{h}_1$ of length $L_1 = 32$ [*Digital Network Echo Cancellers*, ITU-T Recommendations G.168, 2002.], (b) $\mathbf{h}_2$ of length $L_2 = 8$ (randomly generated), (c) $\mathbf{h}_3$ of length $L_3 = 4$ (evaluated as $h_{3,l_3} = 0.5^{l_3 - 1}, \; l_3 = 1, 2, \ldots, L_3$), (d) $\mathbf{h}_4$ of length $L_4 = 4$, (e) $\mathbf{h}_5$ of length $L_5 = 4$, and (f) $\mathbf{h}_6$ of length $L_6 = 4$ (randomly generated).

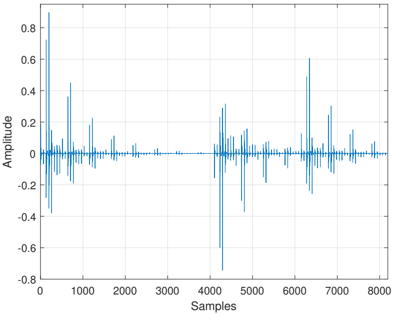

Figure 23: The global impulse response $\mathbf{h} = \mathbf{h}_4 \otimes \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1$, of length $L = L_1 L_2 L_3 L_4 = 8192$.

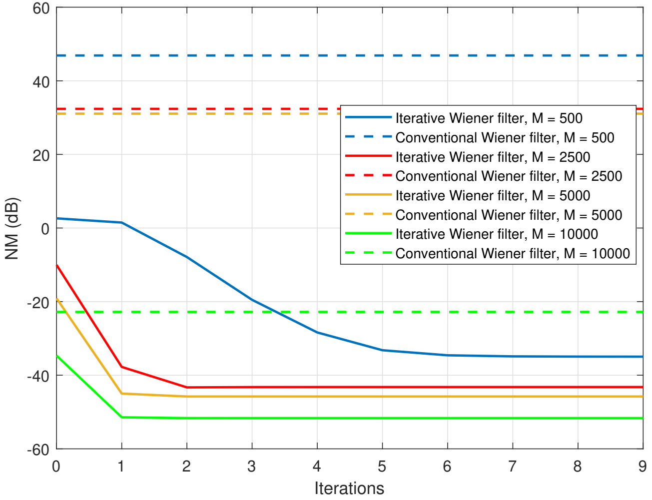

Figure 24: Normalized misalignment of the iterative Wiener filter, for different values of M (available data samples to estimate the statistics), for the identification of the global impulse response from Fig. 23. The input signals are of type AR(1).

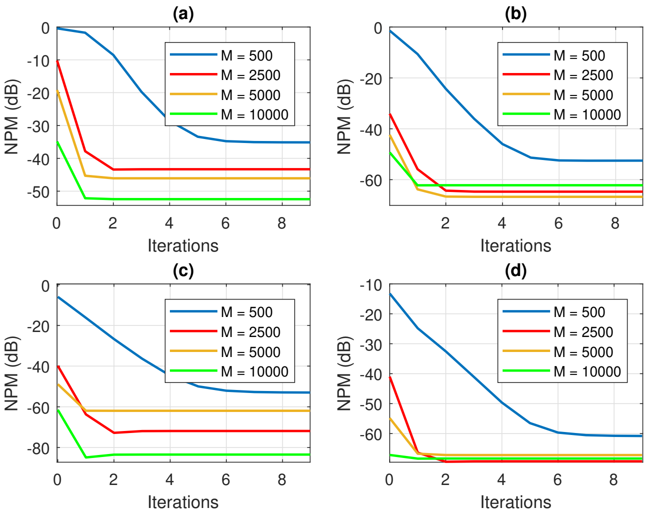

Figure 25: Normalized projection misalignment of the iterative Wiener filter, for different values of M (available data samples to estimate the statistics), for the identification of the individual impulse responses from Fig. 22. The input signals are of type AR(1).

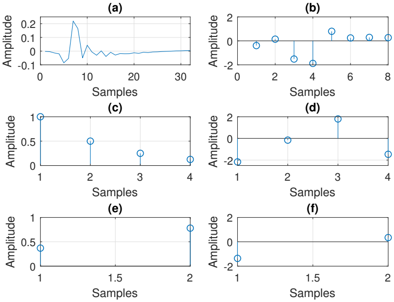

Figure 26: Impulse responses used in simulations: (a) $\mathbf{h}_1$ of length $L_1 = 32$ [*Digital Network Echo Cancellers*, ITU-T Recommendations G.168, 2002.], (b) $\mathbf{h}_2$ of length $L_2 = 8$ (randomly generated), (c) $\mathbf{h}_3$ of length $L_3 = 4$ (evaluated as $h_{3,l_3} = 0.5^{l_3-1}$, $l_3 = 1, 2, \ldots, L_3$), (d) $\mathbf{h}_4$ of length $L_4 = 4$, (e) $\mathbf{h}_5$ of length $L_5 = 4$, and (f) $\mathbf{h}_6$ of length $L_6 = 4$ (randomly generated).

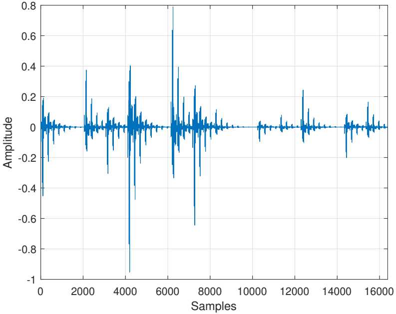

Figure 27: The global impulse response $\mathbf{h} = \mathbf{h}_6 \otimes \mathbf{h}_5 \otimes \mathbf{h}_4 \otimes \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1$, of length $L = L_1 L_2 L_3 L_4 L_5 L_6 = 16384$.

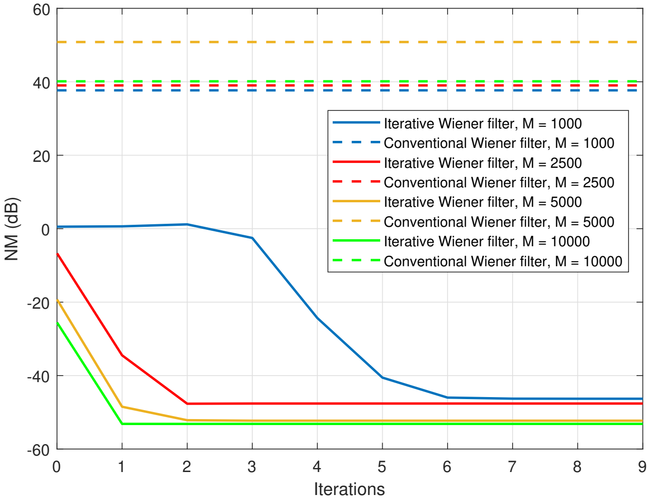

Figure 28: Normalized misalignment of the iterative Wiener filter, for different values of M (available data samples to estimate the statistics), for the identification of the global impulse response from Fig. 27. The input signals are of type AR(1).

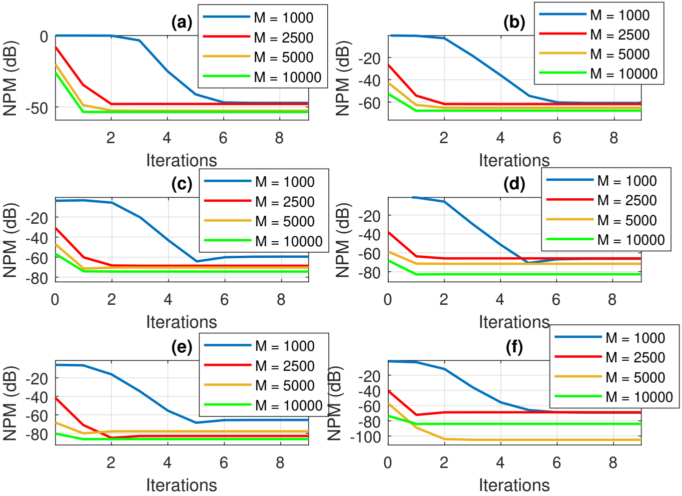

Figure 29: Normalized projection misalignment of the iterative Wiener filter, for different values of M (available data samples to estimate the statistics), for the identification of the individual impulse responses from Fig. 26. The input signals are of type AR(1).

LMS algorithm for the identification of multilinear forms

→ It can be verified that

$$e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = \cdots = e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)$$

LMS algorithm for the identification of multilinear forms

→ It can be verified that

$$e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = \cdots = e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)$$

- **LMS-MF** updates:

$$\begin{aligned}
\widehat{\mathbf{h}}_1(t) &= \widehat{\mathbf{h}}_1(t-1) + \mu_{\widehat{\mathbf{h}}_1}\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)\\
\widehat{\mathbf{h}}_2(t) &= \widehat{\mathbf{h}}_2(t-1) + \mu_{\widehat{\mathbf{h}}_2}\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)\\
&\vdots\\
\widehat{\mathbf{h}}_N(t) &= \widehat{\mathbf{h}}_N(t-1) + \mu_{\widehat{\mathbf{h}}_N}\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)
\end{aligned}$$

→ $\mu_{\widehat{\mathbf{h}}_i} > 0,\ i = 1, 2, \ldots, N$: step-size parameters

- For non-stationary signals: it may be more appropriate to use **time-dependent step-sizes** $\mu_{\widehat{\mathbf{h}}_i}(t)$

NLMS-MF

- For non-stationary signals: it may be more appropriate to use **time-dependent step-sizes** $\mu_{\widehat{\mathbf{h}}_i}(t)$
- A posteriori error signals:

$$\varepsilon_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = d(t) - \widehat{\mathbf{h}}_1^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)$$
$$\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) = d(t) - \widehat{\mathbf{h}}_2^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)$$
$$\vdots$$
$$\varepsilon_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t) = d(t) - \widehat{\mathbf{h}}_N^T(t)\mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)$$

NLMS-MF

- By cancelling the a posteriori error signals $\Rightarrow$ **NLMS-MF**:

$$\widehat{\mathbf{h}}_1(t) = \widehat{\mathbf{h}}_1(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_1} \mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) e_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)}{\delta_{\widehat{\mathbf{h}}_1} + \mathbf{x}^T_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) \mathbf{x}_{\widehat{\mathbf{h}}_2\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)}$$

$$\widehat{\mathbf{h}}_2(t) = \widehat{\mathbf{h}}_2(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_2} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)}{\delta_{\widehat{\mathbf{h}}_2} + \mathbf{x}^T_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t) \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_3\ldots\widehat{\mathbf{h}}_N}(t)}$$

$$\vdots$$

$$\widehat{\mathbf{h}}_N(t) = \widehat{\mathbf{h}}_N(t-1) + \frac{\alpha_{\widehat{\mathbf{h}}_N} \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t) e_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)}{\delta_{\widehat{\mathbf{h}}_N} + \mathbf{x}^T_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t) \mathbf{x}_{\widehat{\mathbf{h}}_1\widehat{\mathbf{h}}_2\ldots\widehat{\mathbf{h}}_{N-1}}(t)}$$

Simulation Setup

- MISO system of order $N = 4$
- $\mathbf{h}_l$, $l = 1, 2, 3, 4$: randomly generated (with Gaussian distribution)
- $L_1 = 32, L_2 = 8, L_3 = 4, L_4 = 2$
- input signals - independent AR(1), obtained by filtering WGN signals through a first-order system $1/\left(1 - 0.8z^{-1}\right)$
- $w(t)$ - AWGN, with variance $\sigma_w^2 = 0.01$

Simulation Setup

- MISO system of order $N = 4$
- $\mathbf{h}_l$, $l = 1, 2, 3, 4$: randomly generated (with Gaussian distribution)
- $L_1 = 32, L_2 = 8, L_3 = 4, L_4 = 2$
- input signals - independent AR(1), obtained by filtering WGN signals through a first-order system $1/\left(1 - 0.8z^{-1}\right)$
- $w(t)$ - AWGN, with variance $\sigma_w^2 = 0.01$

- Performance measure: Normalized misalignment (NM)

$$\text{NM}[\mathbf{f}, \widehat{\mathbf{f}}](\text{dB}) = 20 \log_{10} \left[\frac{\|\mathbf{f} - \widehat{\mathbf{f}}\|^2}{\|\mathbf{f}\|^2} \right]$$

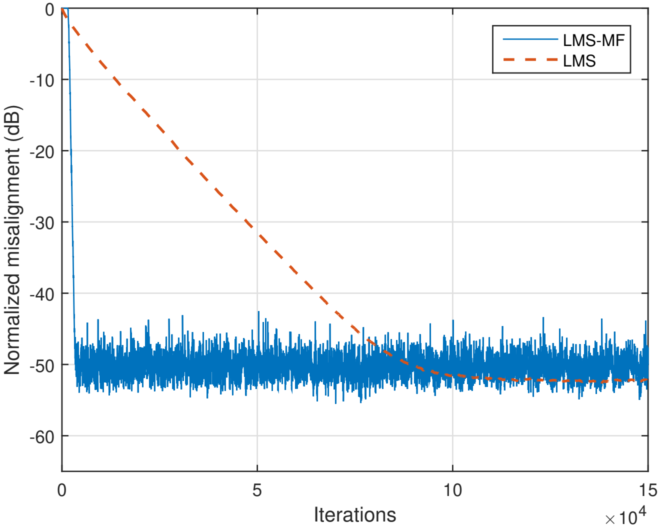

Figure 30: Normalized misalignment of the LMS-MF and LMS algorithms. The inputs are AR(1) processes, $L_1 L_2 L_3 L_4 = 2048$ and $\sigma_w^2 = 0.01$.

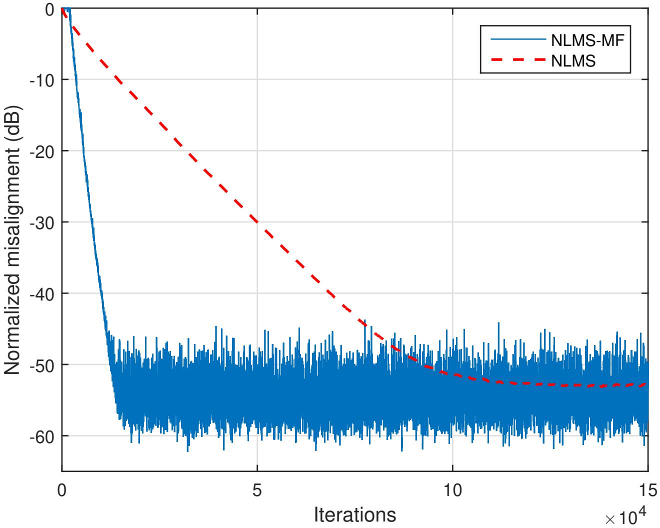

Figure 31: Normalized misalignment of the NLMS-MF and NLMS algorithms. The inputs are AR(1) processes, $L_1 L_2 L_3 L_4 = 2048$ and $\sigma_w^2 = 0.01$.

Outline

Nearest Kronecker Product Decomposition and Low-Rank Approximation

- **Motivation**:
  - System identification is very difficult in case of long length impulse responses (slow convergence, high complexity, low accuracy of the solution)
  - Bilinear and trilinear forms are only applicable to perfectly separable systems
  - Many echo paths are sparse in nature $\Rightarrow$ low-rank systems

Nearest Kronecker Product Decomposition and Low-Rank Approximation

- **Motivation**:
  - System identification is very difficult in case of long length impulse responses (slow convergence, high complexity, low accuracy of the solution)
  - Bilinear and trilinear forms are only applicable to perfectly separable systems
  - Many echo paths are sparse in nature $\Rightarrow$ low-rank systems
- **Idea**: decompose such high-dimension system identification problems into low-dimension problems combined together
- **Solution**:
  - Nearest Kronecker product decomposition
  - Low-rank approximation, to decrease computational complexity

Kalman filter based on the NKP decomposition

- $\mathbf{h}$: unknown system of length $L = L_1 L_2$, $L_1 \geq L_2$
- Reshape $\mathbf{h}$ into an $L_1 \times L_2$ matrix: $\mathbf{H} = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \ldots & \mathbf{s}_{L_2} \end{bmatrix}$
  $\rightarrow \mathbf{s}_l,\ l = 1, 2, \ldots, L_2$: short impulse responses of length L_1 each
- Approximate $\mathbf{h}$ by $\mathbf{h}_2 \otimes \mathbf{h}_1$, where $\mathbf{h}_1$: length L_1, $\mathbf{h}_2$: length L_2
- Performance measure: $\mathcal{M}(\mathbf{h}_1, \mathbf{h}_2) = \frac{\|\mathbf{h} - \mathbf{h}_2 \otimes \mathbf{h}_1\|_2}{\|\mathbf{h}\|_2} = \frac{\left\|\mathbf{H} - \mathbf{h}_1 \mathbf{h}_2^T\right\|_\mathrm{F}}{\|\mathbf{H}\|_\mathrm{F}}$
- Minimize $\mathcal{M} \Longleftrightarrow$ find the nearest rank-1 matrix to $\mathbf{H}$: SVD
- After computations, the NKP decomposition of $\mathbf{h}$ is:

$$\mathbf{h}(t) = \sum_{p=1}^{P} \mathbf{h}_{2,p}(t) \otimes \mathbf{h}_{1,p}(t)$$

- Equivalent forms of the error signal:

$$e_1(t) = d(t) - \sum_{p=1}^{P} \widehat{\mathbf{h}}_{1,p}^T(t-1)\mathbf{x}_{2,p}(t) = d(t) - \underline{\widehat{\mathbf{h}}}_1^T(t-1)\underline{\mathbf{x}}_2(t)$$
$$e_2(t) = d(t) - \sum_{p=1}^{P} \widehat{\mathbf{h}}_{2,p}^T(t-1)\mathbf{x}_{1,p}(t) = d(t) - \underline{\widehat{\mathbf{h}}}_2^T(t-1)\underline{\mathbf{x}}_1(t)$$

- Original system (length $L_1 L_2$) $\Rightarrow$ 2 shorter filters (lengths PL_1, PL_2)

$\Rightarrow$ **Kalman filter based on the NKP decomposition (KF-NKP)**

- SVD: $\mathbf{H} = \mathbf{U}_1 \Sigma \mathbf{U}_2^T = \sum_{l=1}^{L_2} \sigma_l \mathbf{u}_{1,l} \mathbf{u}_{2,l}^T$

$\rightarrow$ $\mathbf{U}_1$, $\mathbf{U}_2$: orthogonal matrices of sizes $L_1 \times L_1$, $L_2 \times L_2$
$\rightarrow$ Σ - $L_1 \times L_2$ rectangular diagonal matrix with nonnegative real numbers on its main diagonal
$\rightarrow$ $\mathbf{u}_{1,l}$, $\mathbf{u}_{2,l}$, with $l = 1, 2, \ldots, L_2$: the columns of $\mathbf{U}_1$, $\mathbf{U}_2$ (they are the left-singular, respectively right-singular vectors of $\mathbf{H}$)
$\rightarrow$ diagonal entries σ_l, $l = 1, 2, \ldots, L_2$ of Σ: the singular values of $\mathbf{H}$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L_2} \geq 0$

- SVD: $\mathbf{H} = \mathbf{U}_1 \Sigma \mathbf{U}_2^T = \sum_{l=1}^{L_2} \sigma_l \mathbf{u}_{1,l} \mathbf{u}_{2,l}^T$

$\rightarrow$ $\mathbf{U}_1$, $\mathbf{U}_2$: orthogonal matrices of sizes $L_1 \times L_1$, $L_2 \times L_2$
$\rightarrow$ Σ - $L_1 \times L_2$ rectangular diagonal matrix with nonnegative real numbers on its main diagonal
$\rightarrow$ $\mathbf{u}_{1,l}$, $\mathbf{u}_{2,l}$, with $l = 1, 2, \ldots, L_2$: the columns of $\mathbf{U}_1$, $\mathbf{U}_2$ (they are the left-singular, respectively right-singular vectors of $\mathbf{H}$)
$\rightarrow$ diagonal entries σ_l, $l = 1, 2, \ldots, L_2$ of Σ: the singular values of $\mathbf{H}$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L_2} \geq 0$

- Optimal approximation of $\mathbf{h}$: $\overline{\mathbf{h}} = \overline{\mathbf{h}}_2 \otimes \overline{\mathbf{h}}_1$

$\rightarrow$ $\overline{\mathbf{h}}_1 = \sqrt{\sigma_1}\mathbf{u}_{1,1}$, $\overline{\mathbf{h}}_2 = \sqrt{\sigma_1}\mathbf{u}_{2,1}$ ($\mathbf{u}_{1,1}$, $\mathbf{u}_{2,1}$: the first columns of $\mathbf{U}_1$, $\mathbf{U}_2$)

- SVD: $\mathbf{H} = \mathbf{U}_1 \Sigma \mathbf{U}_2^T = \sum_{l=1}^{L_2} \sigma_l \mathbf{u}_{1,l} \mathbf{u}_{2,l}^T$

$\rightarrow \mathbf{U}_1$, $\mathbf{U}_2$: orthogonal matrices of sizes $L_1 \times L_1$, $L_2 \times L_2$

$\rightarrow \Sigma$ - $L_1 \times L_2$ rectangular diagonal matrix with nonnegative real numbers on its main diagonal

$\rightarrow \mathbf{u}_{1,l}$, $\mathbf{u}_{2,l}$, with $l = 1, 2, \ldots, L_2$: the columns of $\mathbf{U}_1$, $\mathbf{U}_2$ (they are the left-singular, respectively right-singular vectors of $\mathbf{H}$)

$\rightarrow$ diagonal entries $\sigma_l,\ l = 1, 2, \ldots, L_2$ of Σ: the singular values of $\mathbf{H}$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L_2} \geq 0$

- Optimal approximation of $\mathbf{h}$: $\overline{\mathbf{h}} = \overline{\mathbf{h}}_2 \otimes \overline{\mathbf{h}}_1$

$\rightarrow \overline{\mathbf{h}}_1 = \sqrt{\sigma_1}\mathbf{u}_{1,1}$, $\overline{\mathbf{h}}_2 = \sqrt{\sigma_1}\mathbf{u}_{2,1}$ ($\mathbf{u}_{1,1}$, $\mathbf{u}_{2,1}$: the first columns of $\mathbf{U}_1$, $\mathbf{U}_2$)

- In the general case: the impulse responses that compose $\mathbf{h}$ ($\mathbf{s}_l,\ l = 1, 2, \ldots, L_2$) may not be that linearly dependent

- SVD: $\mathbf{H} = \mathbf{U}_1 \Sigma \mathbf{U}_2^T = \sum_{l=1}^{L_2} \sigma_l \mathbf{u}_{1,l} \mathbf{u}_{2,l}^T$

$\rightarrow \mathbf{U}_1$, $\mathbf{U}_2$: orthogonal matrices of sizes $L_1 \times L_1$, $L_2 \times L_2$
$\rightarrow \Sigma$ - $L_1 \times L_2$ rectangular diagonal matrix with nonnegative real numbers on its main diagonal
$\rightarrow \mathbf{u}_{1,l}$, $\mathbf{u}_{2,l}$, with $l = 1, 2, \ldots, L_2$: the columns of $\mathbf{U}_1$, $\mathbf{U}_2$ (they are the left-singular, respectively right-singular vectors of $\mathbf{H}$)
$\rightarrow$ diagonal entries $\sigma_l,\ l = 1, 2, \ldots, L_2$ of Σ: the singular values of $\mathbf{H}$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L_2} \geq 0$

- Optimal approximation of $\mathbf{h}$: $\overline{\mathbf{h}} = \overline{\mathbf{h}}_2 \otimes \overline{\mathbf{h}}_1$

$\rightarrow \overline{\mathbf{h}}_1 = \sqrt{\sigma_1}\mathbf{u}_{1,1}$, $\overline{\mathbf{h}}_2 = \sqrt{\sigma_1}\mathbf{u}_{2,1}$ ($\mathbf{u}_{1,1}$, $\mathbf{u}_{2,1}$: the first columns of $\mathbf{U}_1$, $\mathbf{U}_2$)

- In the general case: the impulse responses that compose $\mathbf{h}$ ($\mathbf{s}_l,\ l = 1, 2, \ldots, L_2$) may not be that linearly dependent
- Solution: use the approximation
  $\mathbf{h} \approx \sum_{p=1}^{P} \mathbf{h}_{2,p} \otimes \mathbf{h}_{1,p} = \mathrm{vec}\left(\mathbf{H}_1 \mathbf{H}_2^T\right)$, $P \leq L_2$
  $\rightarrow \mathbf{h}_{1,p}$, $\mathbf{h}_{2,p}$: impulse responses of lengths L_1 and L_2
  $\rightarrow \mathbf{H}_1 = \begin{bmatrix} \mathbf{h}_{1,1} & \mathbf{h}_{1,2} & \ldots & \mathbf{h}_{1,P} \end{bmatrix}$, $\mathbf{H}_2 = \begin{bmatrix} \mathbf{h}_{2,1} & \mathbf{h}_{2,2} & \ldots & \mathbf{h}_{2,P} \end{bmatrix}$: matrices of sizes $L_1 \times P$ and $L_2 \times P$

- Performance measure: $\mathcal{M}\left(\mathbf{H}_1, \mathbf{H}_2\right) = \frac{\left\|\mathbf{H}-\mathbf{H}_1\mathbf{H}_2^T\right\|_{\mathrm{F}}}{\|\mathbf{H}\|_{\mathrm{F}}}$
- Optimal solutions:
  $\overline{\mathbf{H}}_1 = \left[\begin{array}{cccc} \overline{\mathbf{h}}_{1,1} & \overline{\mathbf{h}}_{1,2} & \ldots & \overline{\mathbf{h}}_{1,P} \end{array}\right] = \left[\sqrt{\sigma_1}\mathbf{u}_{1,1} \quad \sqrt{\sigma_2}\mathbf{u}_{1,2} \ldots \sqrt{\sigma_P}\mathbf{u}_{1,P}\right]$
  $\overline{\mathbf{H}}_2 = \left[\begin{array}{cccc} \overline{\mathbf{h}}_{2,1} & \overline{\mathbf{h}}_{2,2} & \ldots & \overline{\mathbf{h}}_{2,P} \end{array}\right] = \left[\sqrt{\sigma_1}\mathbf{u}_{2,1} \quad \sqrt{\sigma_2}\mathbf{u}_{2,2} \ldots \sqrt{\sigma_P}\mathbf{u}_{2,P}\right]$
  $\rightarrow \mathbf{u}_{1,p}, \mathbf{u}_{2,p},\ p = 1, 2, \ldots, P$: the first P columns of $\mathbf{U}_1$, $\mathbf{U}_2$
- Optimal approximation of $\mathbf{h}$:

  $$\overline{\mathbf{h}}(P) = \sum_{p=1}^{P} \overline{\mathbf{h}}_{2,p} \otimes \overline{\mathbf{h}}_{1,p} = \sum_{p=1}^{P} \sigma_p \mathbf{u}_{2,p} \otimes \mathbf{u}_{1,p}$$

  $\rightarrow$ the exact decomposition is obtained for $P = L_2$
  $\rightarrow$ if $\operatorname{rank}(\mathbf{H}) = P < L_2$ (i.e., $\sigma_i = 0$, for $P < i \leq L_2$) $\Rightarrow$ $\mathbf{h}$ can be estimated at least as well as in the conventional approach
  $\rightarrow$ if P is reasonably low as compared to $L_2 \Rightarrow$ important decrease in complexity

System Model

- **Signal model**: $d(t) = \mathbf{h}^T(t)\mathbf{x}(t) + v(t) = y(t) + v(t)$
  $\rightarrow d(t)$: reference (desired) signal
  $\rightarrow \mathbf{h}(t)$: unknown system of length $L = L_1 L_2$, $L_1 \geq L_2$
  $\rightarrow \mathbf{x}(t) = \begin{bmatrix} x(t) & x(t-1) & \cdots & x(t-L+1) \end{bmatrix}^T$: the most recent L time samples of the zero-mean input signal $x(t)$
  $\rightarrow v(t)$: zero-mean additive noise, uncorrelated with $\mathbf{x}(t)$
- **Goal**: Estimate $\mathbf{h}(t)$ using an adaptive filter $\widehat{\mathbf{h}}(t)$
- After computations, the NKP decomposition of $\mathbf{h}$ is:

$$\mathbf{h}(t) = \sum_{p=1}^{P} \mathbf{h}_{2,p}(t) \otimes \mathbf{h}_{1,p}(t)$$

  $\rightarrow$ we can group the vectors as:

$$\underline{\mathbf{h}}_1(t) = \begin{bmatrix} \mathbf{h}_{1,1}^T(t) & \mathbf{h}_{1,2}^T(t) & \cdots & \mathbf{h}_{1,P}^T(t) \end{bmatrix}^T, \text{ of length } PL_1$$

$$\underline{\mathbf{h}}_2(t) = \begin{bmatrix} \mathbf{h}_{2,1}^T(t) & \mathbf{h}_{2,2}^T(t) & \cdots & \mathbf{h}_{2,P}^T(t) \end{bmatrix}^T, \text{ of length } PL_2$$

- Error signal: $e(t) = d(t) - \widehat{y}(t) = d(t) - \widehat{\mathbf{h}}^T(t-1)\mathbf{x}(t)$
- NKP decomposition of the estimated filter:

$$\widehat{\mathbf{h}}(t) = \sum_{p=1}^{P} \widehat{\mathbf{h}}_{2,p}(t) \otimes \widehat{\mathbf{h}}_{1,p}(t)$$

$$\rightarrow \widehat{\mathbf{h}}_{2,p}(t) \otimes \widehat{\mathbf{h}}_{1,p}(t) = \left[\widehat{\mathbf{h}}_{2,p}(t) \otimes \mathbf{I}_{L_1}\right] \widehat{\mathbf{h}}_{1,p}(t) = \left[\mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_{1,p}(t)\right] \widehat{\mathbf{h}}_{2,p}(t)$$

$\rightarrow$ notations: $\left[\widehat{\mathbf{h}}_{2,p}(t-1) \otimes \mathbf{I}_{L_1}\right]^T \mathbf{x}(t) \overset{\text{not.}}{=} \mathbf{x}_{2,p}(t)$

$$\left[\mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_{1,p}(t-1)\right]^T \mathbf{x}(t) \overset{\text{not.}}{=} \mathbf{x}_{1,p}(t)$$

$\rightarrow$ we can group the vectors as:

$$\underline{\widehat{\mathbf{h}}}_1(t) = \begin{bmatrix} \widehat{\mathbf{h}}_{1,1}^T(t) & \widehat{\mathbf{h}}_{1,2}^T(t) & \cdots & \widehat{\mathbf{h}}_{1,P}^T(t) \end{bmatrix}^T$$

$$\underline{\mathbf{x}}_2(t) = \begin{bmatrix} \mathbf{x}_{2,1}^T(t) & \mathbf{x}_{2,2}^T(t) & \cdots & \mathbf{x}_{2,P}^T(t) \end{bmatrix}^T$$

$$\underline{\widehat{\mathbf{h}}}_2(t) = \begin{bmatrix} \widehat{\mathbf{h}}_{2,1}^T(t) & \widehat{\mathbf{h}}_{2,2}^T(t) & \cdots & \widehat{\mathbf{h}}_{2,P}^T(t) \end{bmatrix}^T$$

$$\underline{\mathbf{x}}_1(t) = \begin{bmatrix} \mathbf{x}_{1,1}^T(t) & \mathbf{x}_{1,2}^T(t) & \cdots & \mathbf{x}_{1,P}^T(t) \end{bmatrix}^T$$

Kalman filter based on the NKP decomposition

- Equivalent forms of the error signal:

$$e_1(t) = d(t) - \sum_{p=1}^{P} \widehat{\mathbf{h}}_{1,p}^T(t-1)\mathbf{x}_{2,p}(t) = d(t) - \underline{\widehat{\mathbf{h}}}_1^T(t-1)\underline{\mathbf{x}}_2(t)$$

$$e_2(t) = d(t) - \sum_{p=1}^{P} \widehat{\mathbf{h}}_{2,p}^T(t-1)\mathbf{x}_{1,p}(t) = d(t) - \underline{\widehat{\mathbf{h}}}_2^T(t-1)\underline{\mathbf{x}}_1(t)$$

- Original system (length $L_1 L_2$) $\Rightarrow$ 2 shorter filters (lengths PL_1, PL_2)

- **Kalman filter based on the NKP decomposition**:

$$\underline{\widehat{\mathbf{h}}}_1(t) = \underline{\widehat{\mathbf{h}}}_1(t-1) + \mathbf{k}_1(t)e_1(t) \qquad \underline{\widehat{\mathbf{h}}}_2(t) = \underline{\widehat{\mathbf{h}}}_2(t-1) + \mathbf{k}_2(t)e_2(t)$$

$\rightarrow$ $\mathbf{k}_1(t), \mathbf{k}_2(t)$: Kalman gain vectors:

$$\mathbf{k}_1(t) = \mathbf{R}_{\mathbf{m}_1}(t)\underline{\mathbf{x}}_2(t)\left[\underline{\mathbf{x}}_2^T(t)\mathbf{R}_{\mathbf{m}_1}(t)\underline{\mathbf{x}}_2(t) + \sigma_v^2\right]^{-1}$$

$$\mathbf{k}_2(t) = \mathbf{R}_{\mathbf{m}_2}(t)\underline{\mathbf{x}}_1(t)\left[\underline{\mathbf{x}}_1^T(t)\mathbf{R}_{\mathbf{m}_2}(t)\underline{\mathbf{x}}_1(t) + \sigma_v^2\right]^{-1}$$

Kalman filter based on the NKP decomposition

- A posteriori misalignments:

$\boldsymbol{\mu}_1(t) = \underline{\mathbf{h}}_1(t) - \widehat{\underline{\mathbf{h}}}_1(t)$, with correlation matrix $\mathbf{R}_{\boldsymbol{\mu}_1}(t) = E\left[\boldsymbol{\mu}_1(t)\boldsymbol{\mu}_1^T(t)\right]$

$\boldsymbol{\mu}_2(t) = \underline{\mathbf{h}}_2(t) - \widehat{\underline{\mathbf{h}}}_2(t)$, with correlation matrix $\mathbf{R}_{\boldsymbol{\mu}_2}(t) = E\left[\boldsymbol{\mu}_2(t)\boldsymbol{\mu}_2^T(t)\right]$

- A priori misalignments:

$\mathbf{m}_1(t) = \underline{\mathbf{h}}_1(t) - \widehat{\underline{\mathbf{h}}}_1(t-1) = \boldsymbol{\mu}_1(t-1) + \mathbf{w}_1(t), \quad \mathbf{R}_{\mathbf{m}_1}(t) = E\left[\mathbf{m}_1(t)\mathbf{m}_1^T(t)\right]$

$\mathbf{m}_2(t) = \underline{\mathbf{h}}_2(t) - \widehat{\underline{\mathbf{h}}}_2(t-1) = \boldsymbol{\mu}_2(t-1) + \mathbf{w}_2(t), \quad \mathbf{R}_{\mathbf{m}_2}(t) = E\left[\mathbf{m}_2(t)\mathbf{m}_2^T(t)\right]$

- It is clear that:

  $\mathbf{R}_{\mathbf{m}_1}(t) = \mathbf{R}_{\boldsymbol{\mu}_1}(t-1) + \mathbf{R}_{\mathbf{w}_1}$ $\qquad$ $\mathbf{R}_{\mathbf{m}_2}(t) = \mathbf{R}_{\boldsymbol{\mu}_2}(t-1) + \mathbf{R}_{\mathbf{w}_2}$

- The Kalman gain vectors are:

$$\mathbf{k}_1(t) = \mathbf{R}_{\mathbf{m}_1}(t)\underline{\mathbf{x}}_2(t)\left[\underline{\mathbf{x}}_2^T(t)\mathbf{R}_{\mathbf{m}_1}(t)\underline{\mathbf{x}}_2(t) + \sigma_v^2\right]^{-1}$$

$$\mathbf{k}_2(t) = \mathbf{R}_{\mathbf{m}_2}(t)\underline{\mathbf{x}}_1(t)\left[\underline{\mathbf{x}}_1^T(t)\mathbf{R}_{\mathbf{m}_2}(t)\underline{\mathbf{x}}_1(t) + \sigma_v^2\right]^{-1}$$

Computational Complexity

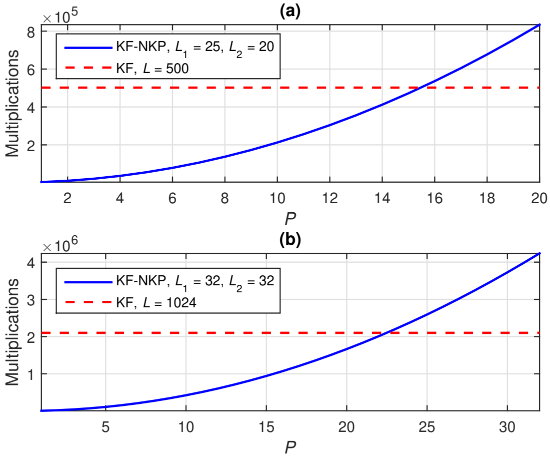

Figure 32: Number of multiplications (per iteration) required by the KF-NKP and KF, as a function of P. The KF-NKP uses two shorter filters of lengths PL_1 and PL_2 (with $P \leq L_2$), while the length of the KF is $L = L_1 L_2$: (a) $L_1 = 25$, $L_2 = 20$, and (b) $L_1 = L_2 = 32$.

Estimation of KF-NKP parameters

- So far, $\mathbf{w}_1(t)$ and $\mathbf{w}_2(t)$ were considered zero-mean WGN signals
- We could consider a more realistic case:

$$\underline{\mathbf{h}}_1(t) = \underline{\mathbf{h}}_1(t-1) + \widetilde{\mathbf{w}}_1(t) \qquad \underline{\mathbf{h}}_2(t) = \underline{\mathbf{h}}_2(t-1) + \widetilde{\mathbf{w}}_2(t)$$

$\rightarrow$ independent fluctuations of each coefficient:

$$\widetilde{\mathbf{w}}_1(t) = \begin{bmatrix} \widetilde{w}_{1,0}(t) & \widetilde{w}_{1,1}(t) & \cdots & \widetilde{w}_{1,PL_1-1}(t) \end{bmatrix}^T$$
$$\widetilde{\mathbf{w}}_2(t) = \begin{bmatrix} \widetilde{w}_{2,0}(t) & \widetilde{w}_{2,1}(t) & \cdots & \widetilde{w}_{2,PL_2-1}(t) \end{bmatrix}^T$$

- Thus, we can express:

$\widetilde{w}_{1,l}(t) = \underline{h}_{1,l}(t) - \underline{h}_{1,l}(t-1),\ l = 0, 1, \ldots, PL_1 - 1$
$\widetilde{w}_{2,j}(t) = \underline{h}_{2,j}(t) - \underline{h}_{2,j}(t-1),\ j = 0, 1, \ldots, PL_2 - 1$

with

$$E\left[\widetilde{w}_{1,k}(t)\widetilde{w}_{1,l}(t)\right] = \begin{cases} \sigma^2_{\widetilde{w}_{1,l}}, & k = l \\ 0, & k \neq l \end{cases}, \ k, l = 0, 1, \ldots, PL_1 - 1$$

$$E\left[\widetilde{w}_{2,i}(t)\widetilde{w}_{2,j}(t)\right] = \begin{cases} \sigma^2_{\widetilde{w}_{2,j}}, & i = j \\ 0, & i \neq j \end{cases}, \ i, j = 0, 1, \ldots, PL_2 - 1$$

Estimation of KF-NKP parameters

- After computations, we obtain:

$$\widehat{\sigma}^2_{\widetilde{w}_{1,l}}(t) = \alpha_1 \widehat{\sigma}^2_{\widetilde{w}_{1,l}}(t-1) + (1-\alpha_1)\left[\underline{\widehat{h}}_{1,l}(t-1) - \underline{\widehat{h}}_{1,l}(t-2)\right]^2$$

$$\widehat{\sigma}^2_{\widetilde{w}_{2,j}}(t) = \alpha_2 \widehat{\sigma}^2_{\widetilde{w}_{2,j}}(t-1) + (1-\alpha_2)\left[\underline{\widehat{h}}_{2,j}(t-1) - \underline{\widehat{h}}_{2,j}(t-2)\right]^2$$

$\rightarrow \alpha_1 = 1 - 1/(\kappa_1 PL_1),\ \kappa_1 \geq 1; \qquad \alpha_2 = 1 - 1/(\kappa_2 PL_2),\ \kappa_2 \geq 1$

$\rightarrow$ when $\alpha_1 = \alpha_2 = 0$ (i.e., without temporal averaging):

$$\widehat{\sigma}^2_{\widetilde{w}_1}(t) = \frac{1}{PL_1}\left\|\underline{\widehat{\mathbf{h}}}_1(t-1) - \underline{\widehat{\mathbf{h}}}_1(t-2)\right\|_2^2$$

$$\widehat{\sigma}^2_{\widetilde{w}_2}(t) = \frac{1}{PL_2}\left\|\underline{\widehat{\mathbf{h}}}_2(t-1) - \underline{\widehat{\mathbf{h}}}_2(t-2)\right\|_2^2$$

- $\underline{\widehat{\sigma}}^2_{\widetilde{w}_{1,l}}(t), \underline{\widehat{\sigma}}^2_{\widetilde{w}_{2,j}}(t)$ are then chosen as:

$$\underline{\widehat{\sigma}}^2_{\widetilde{w}_{1,l}}(t) = \min\left\{\widehat{\sigma}^2_{\widetilde{w}_{1,l}}(t), \widehat{\sigma}^2_{\widetilde{w}_1}(t)\right\},\ l = 0, 1, \ldots, PL_1 - 1$$

$$\underline{\widehat{\sigma}}^2_{\widetilde{w}_{2,j}}(t) = \min\left\{\widehat{\sigma}^2_{\widetilde{w}_{2,j}}(t), \widehat{\sigma}^2_{\widetilde{w}_2}(t)\right\},\ j = 0, 1, \ldots, PL_2 - 1$$

Simulation Setup

Practical Considerations

- So far, $\mathbf{w}_1(t)$ and $\mathbf{w}_2(t)$ were considered zero-mean WGN signals
- In simulations, we consider a more realistic case, with independent fluctuations of each coefficient
- The individual uncertainty parameters are approximated in a similar way as for KF-BF

First set of experiments - toy example

- Input signals - independent AR(1), obtained by filtering WGN signals through a first-order system $1/\left(1 - 0.9z^{-1}\right)$
- $v(t)$ - WGN, SNR$=$ 30 dB

Second set of experiments - more realistic scenario

- Input signals - impulse responses from the G168 Recommendation
- $v(t)$ - WGN, SNR$=$ 20 dB

First Set of Experiments

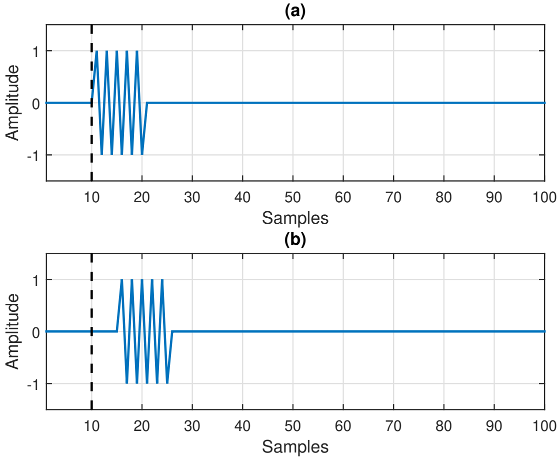

Figure 33: Impulse responses of length $L = 100$, which are decomposed using $L_1 = L_2 = 10$: (a) a cluster of 10 samples (alternating the amplitudes 1 and -1) padded with zero, with $\text{rank}(\mathbf{H}) = 1$; and (b) the same cluster shifted to the right by 5 samples, so that $\text{rank}(\mathbf{H}) = 2$.

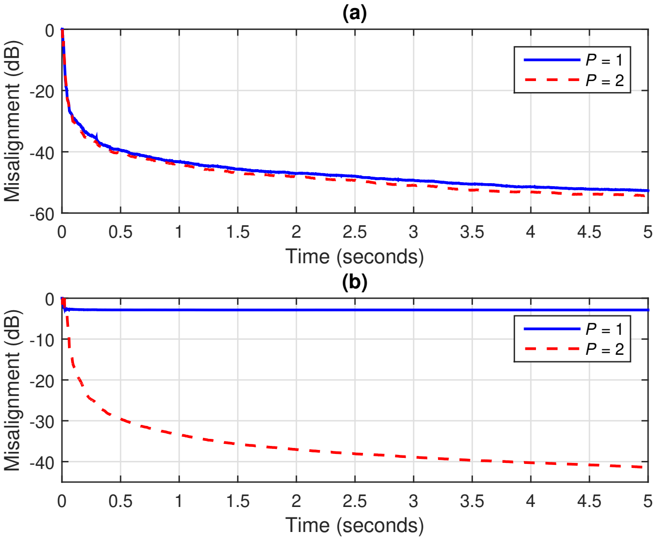

Figure 34: Normalized misalignment of the KF-NKP using $\sigma^2_{w_1} = \sigma^2_{w_2} = 0$, $L_1 = L_2 = 10$, and $P = 1$ or 2, corresponding to the impulse responses from Figs. 33(a) and (b). The input signal is an AR(1) process and $\mathrm{SNR} = 30$ dB.

Second Set of Experiments

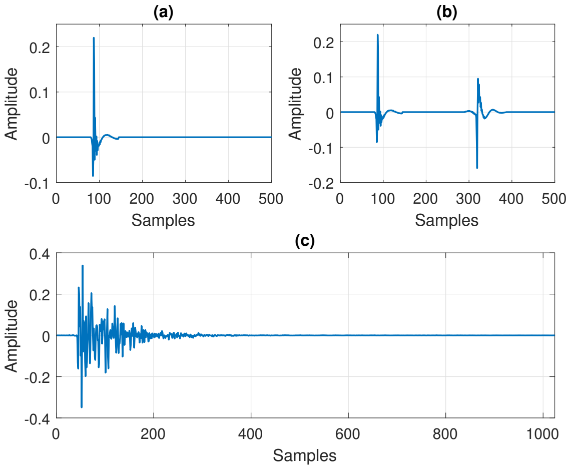

Figure 35: Impulse responses used in simulations: (a) the first impulse response from G168 Recommendation, with $L = 500$; (b) the first and the fifth impulse responses (concatenated) from G168 Recommendation, with $L = 500$; and (c) acoustic impulse response, with $L = 1024$.

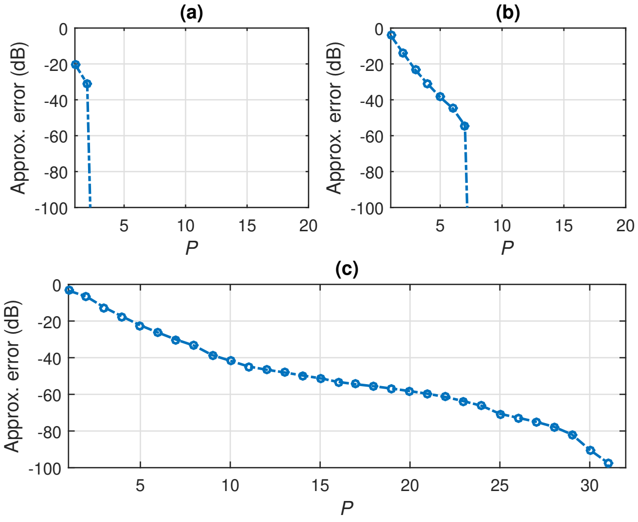

Figure 36: Approximation error (in terms of the normalized misalignment), for the identification of the impulse responses from Fig. 35: (a) impulse response from Fig. 35(a), of length $L = 500$, with $L_1 = 25$ and $L_2 = 20$; (b) impulse response from Fig. 35(b), of length $L = 500$, with $L_1 = 25$ and $L_2 = 20$; and (c) impulse response from Fig. 35(c), of length $L = 1024$, with $L_1 = L_2 = 32$.

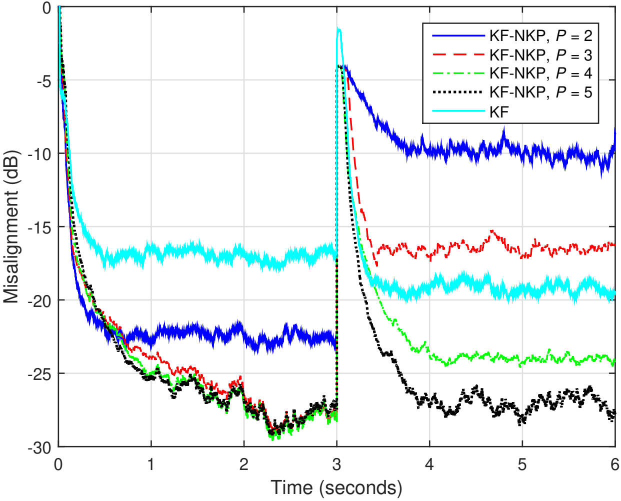

Figure 37: NM of the KF-NKP (using different values of P) and KF, for the identification of the impulse response which changes after 3 seconds from Fig. 35(a) to (b). The input signal is an AR(1) process, $L = 500$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $L_1 = 25$, $L_2 = 20$, and $\sigma^2_{w_1} = \sigma^2_{w_2} = 10^{-8}$; the KF uses the same value of σ^2_w.

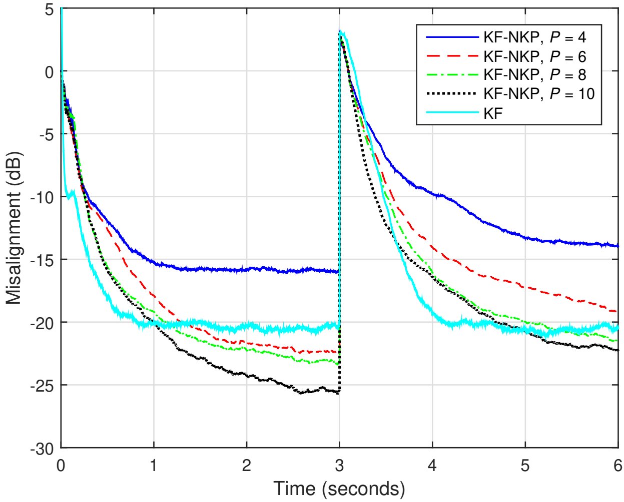

Figure 38: NM of the KF-NKP (using different values of P) and KF, for the identification of the impulse response from Fig. 35(c), which is changed after 3 seconds, by shifting to the right by 12 samples. The input signal is an AR(1) process, $L = 1024$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $L_1 = L_2 = 32$ and $\sigma^2_{w_1} = \sigma^2_{w_2} = 10^{-8}$; the KF uses the same value of its uncertainty parameter.

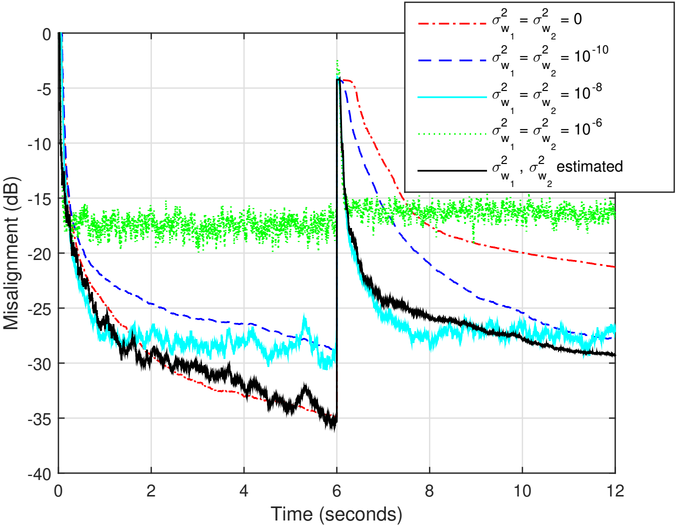

Figure 39: NM of the KF-NKP, for the identification of the impulse responses which changes after 6 seconds from Fig. 35(a) to (b). The input signal is an AR(1) process, $L = 500$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $L_1 = 25$, $L_2 = 20$, $P = 5$, and different values of $\sigma^2_{w_1}$ and $\sigma^2_{w_2}$, including the estimated one.

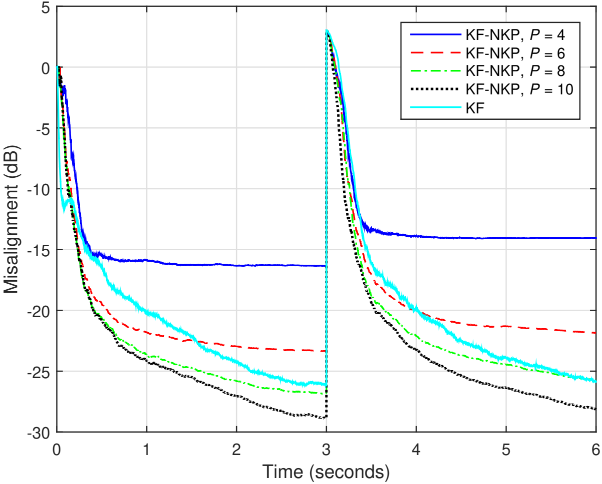

Figure 40: NM of the KF-NKP (using different values of P) and KF, for the identification of the impulse response from Fig. 35(c), which is changed after 3 seconds, by shifting to the right by 12 samples. The input signal is an AR(1) process, $L = 1024$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $L_1 = L_2 = 32$, while the specific parameters $\sigma^2_{w_1}$ and $\sigma^2_{w_2}$ are estimated; the KF uses the uncertainty parameter estimated as in [Paleologu et al., *Proc. IEEE ICASSP*, 2014].

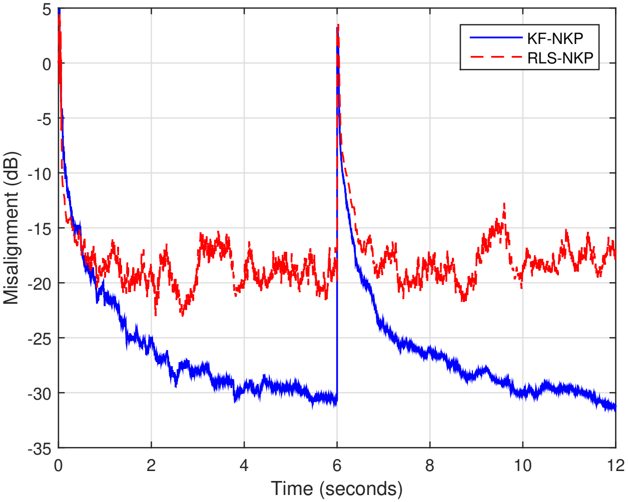

Figure 41: Normalized misalignment of the KF-NKP and RLS-NKP algorithm (using $L_1 = 25$, $L_2 = 20$, and $P = 5$), for the identification of the impulse response from Fig. 35(a). The impulse response changes after 6 seconds. The input signal is a speech sequence, $L = 500$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $\sigma^2_{w_1}$ and $\sigma^2_{w_2}$ estimated.

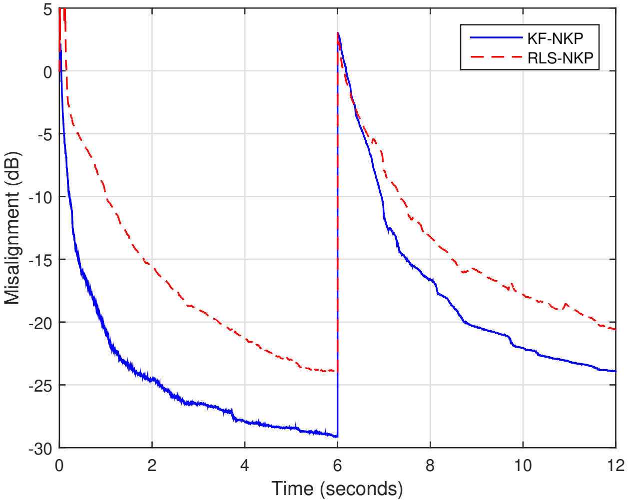

Figure 42: Normalized misalignment of the KF-NKP and RLS-NKP algorithm (using $L_1 = L_2 = 32$ and $P = 10$), for the identification of the impulse response from Fig. 35(c). The impulse response changes after 6 seconds. The input signal is a speech sequence, $L = 1024$, and $\mathrm{SNR} = 20$ dB. The KF-NKP uses $\sigma^2_{w_1}$ and $\sigma^2_{w_2}$ estimated.

Outline

Motivation

- Previous methods for nonlinearities identification:
  $\rightarrow$ Volterra-based approach
  $\rightarrow$ Neural networks

- **Main problem**: very high computational complexity

Motivation

- Previous methods for nonlinearities identification:
  $\rightarrow$ Volterra-based approach
  $\rightarrow$ Neural networks

- **Main problem**: very high computational complexity

- **Our solution**:
  $\rightarrow$ Compute the Taylor series expansion
  $\rightarrow$ Approximate the function using its first significant Taylor series coefficients, neglecting the other ones
  $\rightarrow$ Find the coefficients using an adaptive algorithm

The Nonlinearities Identification Problem

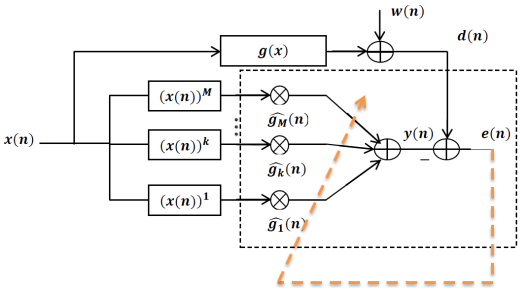

Figure 43: System model.

→ x: zero mean real valued input signal

→ $g(x)$: nonlinear, bijective, odd-type function with the Taylor series expansion of the form $g(x) \cong \sum_{k=1}^{M}(g_k x^k)$

→ $d(n) = \mathbf{g}^T\mathbf{x}(n) + w(n)$: system output, corrupted by AWGN

→ $e(n) = d(n) - \hat{\mathbf{g}}^T\mathbf{x}(n) = (\mathbf{g}^T - \hat{\mathbf{g}}^T)\mathbf{x}(n) + w(n)$: output error

The Nonlinearities Identification Problem

- **Goal** - obtain an estimation of the coefficient vector:

$$\hat{\mathbf{g}}(n) = [\hat{g}_1(n), \hat{g}_2(n), ..., \hat{g}_M(n)]^T$$

- Criterion to minimize - **mean-square error (MSE)**:

$$J(n) = E[e^2(n)] = \sigma_d^2 - 2\hat{\mathbf{g}}^T\mathbf{p} + \hat{\mathbf{g}}^T\mathbf{R}\hat{\mathbf{g}}, \text{ where}$$

$\rightarrow \sigma_d^2 = E[d^2(n)]$ - desired signal variance
$\rightarrow \mathbf{p} = E[\mathbf{x}(n)d(n)]$ - cross-covariance between the input signal $x(n)$ and the desired signal $d(n)$
$\rightarrow \mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ - covariance matrix of the vector $\mathbf{x}(n)$

- Wiener-Hopf solution: $\mathbf{g}_o = \mathbf{R}^{-1}\mathbf{p}$
- **Problems**: $\rightarrow$ the system should be time-invariant
  $\rightarrow$ statistical expectations need to be known

The Adaptive Approach

Least-Mean-Square (LMS) solution:

$$\hat{\mathbf{g}}(n) = \hat{\mathbf{g}}(n-1) + \mu \mathbf{x}(n) e_a(n)$$

$\rightarrow$ $e_a(n) = d(n) - \hat{\mathbf{g}}^T(n-1)\mathbf{x}(n)$: a priori error
$\rightarrow$ μ: step-size parameter

Normalized LMS (NLMS) solution:

$$\mu(n) = \frac{\alpha}{||\mathbf{x}(n)||^2 + \delta}$$

$\rightarrow$ α: normalized step-size ($0 < \alpha < 2$)
$\rightarrow$ δ: regularization parameter

- Covariance matrix of the input signal:

$$\mathbf{R} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\} = [r_{i,j}], i,j = 1,...,M, \quad r_{i,j} = E\{x^{i+j}(n)\}$$

$\rightarrow$ **R** must be non-singular and have a small condition number

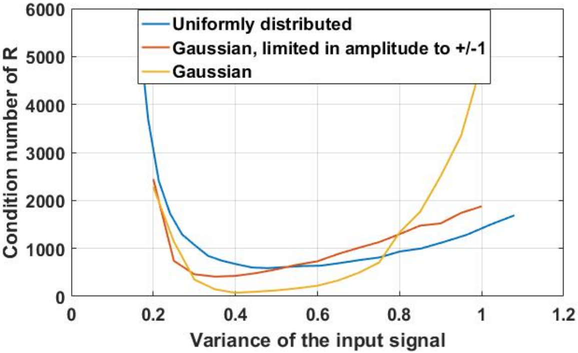

Figure 44: Condition number of **R** as a function of the signal's variance for three types of input signal.

Experimental Results

Simulation Setup:

- NLMS filter of length $M = 6$
- Input: the first M powers of a zero mean Gaussian signal, limited in amplitude to ± 1
- Functions to be identified: $\rightarrow g(x) = x + 0.3x^3 + 0.2x^5$
  $\rightarrow g(x) = \arctan(ax), 0 < a < 1$

Analyze:

- Coefficients' values
- Function's reconstruction

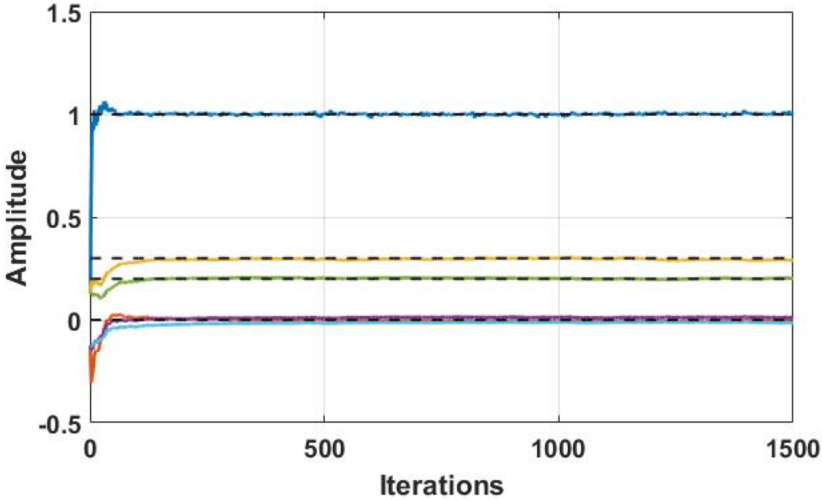

Figure 45: Evolution of the coefficients g_k computed using the NLMS algorithm for the polynomial function $g(x) = x + 0.3x^3 + 0.2x^5$. The black dotted lines are the actual coefficients.

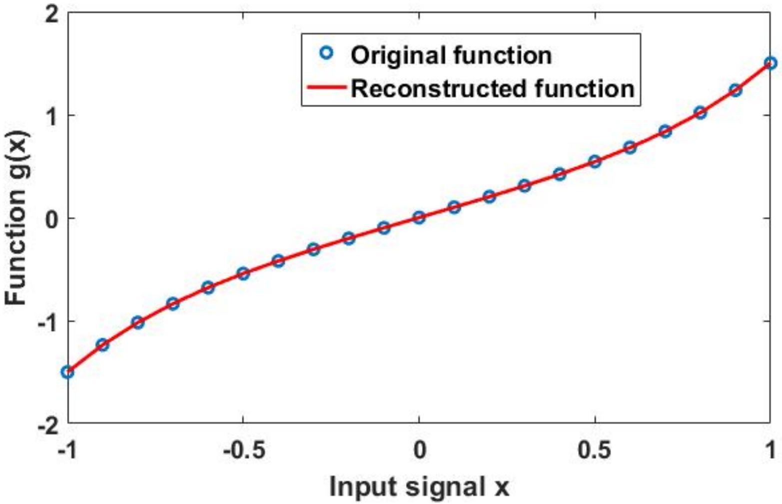

Figure 46: Representation of the polynomial function and the reconstructed function when the input $x \in [-1; 1]$.

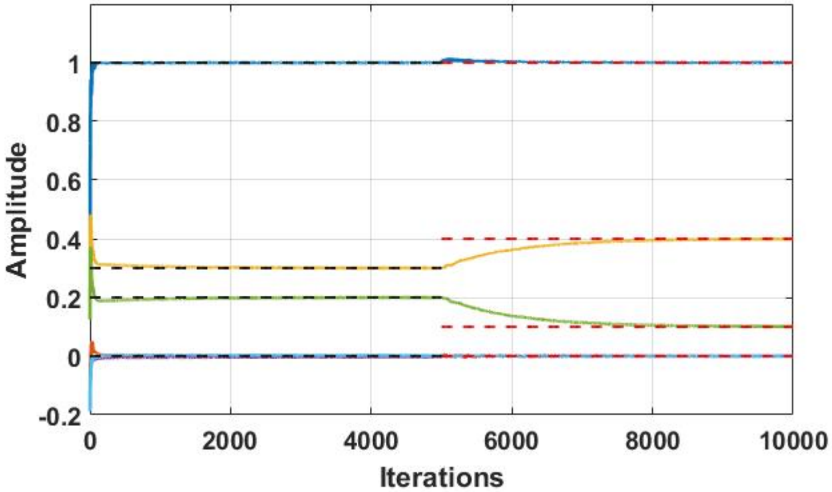

Figure 47: Evolution of the coefficients g_k when a change in their values occurs: $g(x) = x + 0.3x^3 + 0.2x^5$ for the first 5000 iterations (black dotted lines), then $g(x) = x + 0.4x^3 + 0.1x^5$ (red dotted lines).

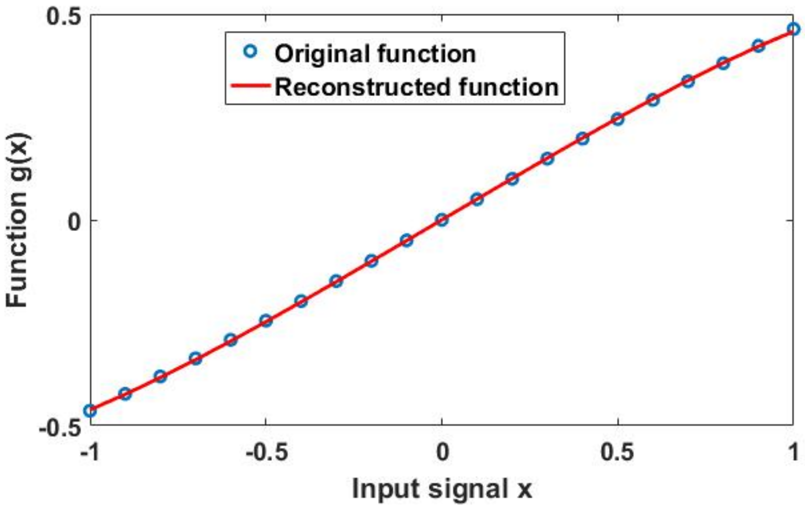

Figure 48: Representation of the arctangent function and the reconstructed function when the input $x \in [-1; 1]$.

Outline

Conclusions

- Contributions in the area of multilinear system identification
- Multilinearity is defined in relation to the individual impulse responses composing the system
- The systems are modeled using tensors
- NKP decomposition and low-rank approximation for systems which are not perfectly separable
- An adaptive method for nonlinear systems (with small nonlinearities)
- Numerous applications, since most real-world systems are nonlinear

Thank you!

Part of this work was supported by a grant of the Romanian Ministry of Education and Research, CNCS – UEFISCDI, project number: PN-III-P1-1.1-PD-2019-0340, within PNCDI III.

Outline