On Fixed Points Algorithms for the Blind Estimation of a SISO FIR System

Ignat Domanov and Lieven De Lathauwer

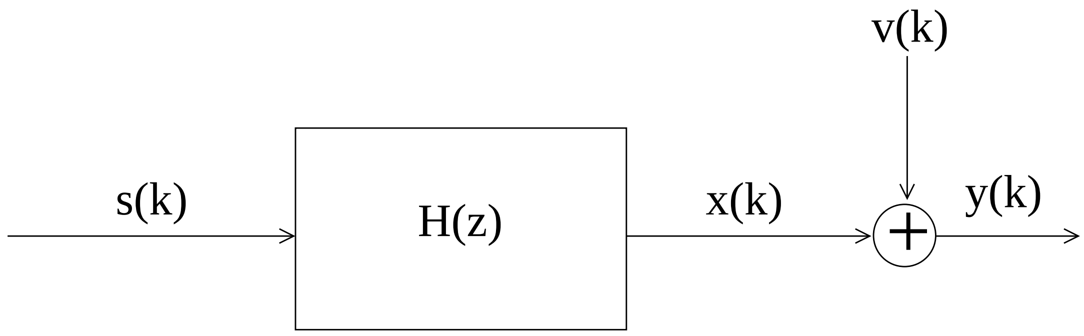
Group Science, Engineering and Technology, KULeuven Campus Kortrijk
and
Department of Electrical Engineering (ESAT), SCD, Katholieke Universiteit Leuven

Outline

- Introduction.
  - Problem of blind estimation of FIR channels based on HOS $\Rightarrow$ polynomial optimization problem.
  - Link of the problem with structured PC decomposition an Single Step Least Squares (SS-LS) algorithm.
- New results.
  - New representation of the cost function and explicit expression of complex gradient.
  - Convergence properties of SS-LS algorithm.
  - Enhanced Plane Search (EPS) procedure
  - Algorithms: "Krasnoselskii" version of SS-LS algorithm, Gradien Descent algorithm improved by EPS with real and complex steps.
- Simulations.

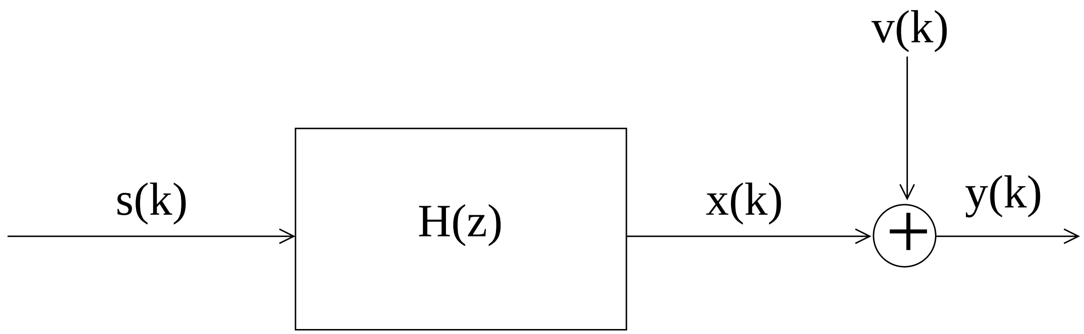
1. Problem formulation

Consider SISO system



- $s(k)$ — **unobserved input**
- $H(z)$ — **unknown transfer function**
- $v(k)$ — **Gaussian additive noise**
- $y(k)$ — **observed channel output**

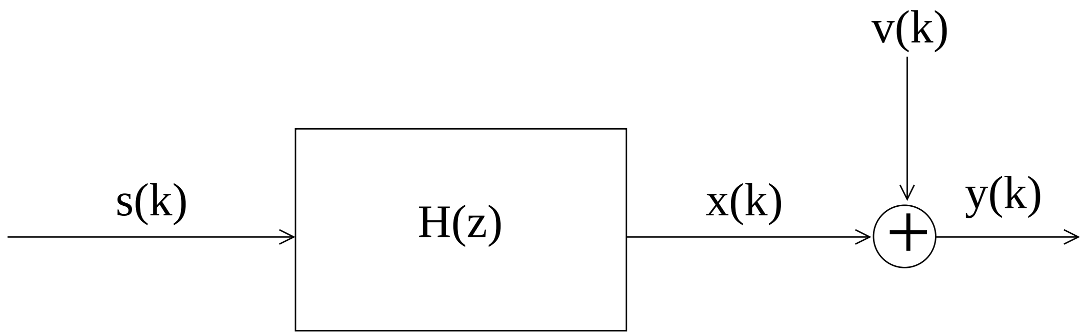
1. Problem formulation



- $s(k)$ — **unobserved input**
- $H(z)$ — **unknown transfer function**
- $v(k)$ — **Gaussian additive noise**
- $y(k)$ — **observed channel output**

The goal : estimate $h(k)$ from the observed system output $y(k)$.

1. Problem formulation



Assume FIR $\Rightarrow$

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h(l)s(k-l) \end{cases}$$

We need $\mathbf{h} = (\mathbf{h(0)}, \mathbf{h(1)}, \ldots, \mathbf{h(L)})^{\mathbf{T}} =: (\mathbf{h_0}, \mathbf{h_1}, \ldots, \mathbf{h_L})^{\mathbf{T}}$.

2. Second Order Statistics

Substitute

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h(l)s(k-l) \end{cases}$$

into

$$c_{2,y}(\tau) := E(y(k)y(k+\tau)) \text{ autocorrelation function}$$

for $\tau = 0, \pm 1, \pm 2, \ldots$

2. Second Order Statistics

Substitue

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h(l)s(k-l) \end{cases}$$

into

$$c_{2,y}(\tau) := E(y(k)y(k+\tau)) \text{ autocorrelation function}$$

for $\tau = 0, \pm 1, \pm 2, \ldots$

if

- $s(k)$ be stationary white noise with finite variance $\gamma_{2,s}$
- $v(k)$ be Gaussian noise $s(n)$.
- $s(k)$ and $v(k)$ be statistically independent.

2. Second Order Statistics

Substitue

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h(l)s(k-l) \end{cases}$$

into

$$c_{2,y}(\tau) := E(y(k)y(k+\tau)) \text{ autocorrelation function}$$

for $\tau = 0, \pm 1, \pm 2, \ldots$

if

- $s(k)$ be stationary white noise with finite variance $\gamma_{2,s}$
- $v(k)$ be Gaussian noise $s(n)$.
- $s(k)$ and $v(k)$ be statistically independent.

then

$$c_{2,y}(\tau) = c_{2,s}(\tau) + c_{2,v}(\tau) = \gamma_{2,s} \sum_{l=0}^{L} h_l h_{l+\tau} + c_{2,v}(\tau)$$

for $\tau = 0, \pm 1, \pm 2, \ldots$

2. Second Order Statistics (Example: L=1)

Example:

$$y(k) = h_0 s(k) + h_1 s(k-1) + v(k),$$

$$c_{2,y}(\tau) = \gamma_{2,s} \sum_{l=0}^{1} h_l h_{l+\tau} + c_{2,v}(\tau), \qquad \tau = 0, \pm 1, \pm 2, \ldots$$

$$\begin{pmatrix} \vdots \\ c_{2,y}(-2) \\ c_{2,y}(-1) \\ c_{2,y}(0) \\ c_{2,y}(1) \\ c_{2,y}(2) \\ \vdots \end{pmatrix} = \gamma_{2,s} \begin{pmatrix} \vdots & \\ 0 & 0 \\ 0 & h_0 \\ h_0 & h_1 \\ h_1 & 0 \\ 0 & 0 \\ \vdots & \end{pmatrix} \begin{pmatrix} h_0 \\ h_1 \end{pmatrix} + \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

2. Second Order Statistics (Example: L=1)

Example:

$$y(k) = h_0 s(k) + h_1 s(k-1) + v(k),$$

$$c_{2,y}(\tau) = \gamma_{2,s} \sum_{l=0}^{1} h_l h_{l+\tau} + c_{2,v}(\tau), \qquad \tau = 0, \pm 1, \pm 2, \ldots$$

$$\begin{pmatrix} \vdots \\ c_{2,y}(-2) \\ c_{2,y}(-1) \\ c_{2,y}(0) \\ c_{2,y}(1) \\ c_{2,y}(2) \\ \vdots \end{pmatrix} = \gamma_{2,s} \begin{pmatrix} \vdots & \\ 0 & 0 \\ 0 & h_0 \\ h_0 & h_1 \\ h_1 & 0 \\ 0 & 0 \\ \vdots & \end{pmatrix} \begin{pmatrix} h_0 \\ h_1 \end{pmatrix} + \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix} = \gamma_{2,s} \begin{pmatrix} \vdots & \\ 0 & 0 \\ 0 & h_1 \\ h_1 & h_0 \\ h_0 & 0 \\ 0 & 0 \\ \vdots & \end{pmatrix} \begin{pmatrix} h_1 \\ h_0 \end{pmatrix} + \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

2. Second Order Statistics (Example: L=1)

Example:

$$y(k) = h_0 s(k) + h_1 s(k-1) + v(k),$$

$$c_{2,y}(\tau) = \gamma_{2,s} \sum_{l=0}^{1} h_l h_{l+\tau} + c_{2,v}(\tau), \qquad \tau = 0, \pm 1, \pm 2, \ldots$$

$$\begin{pmatrix} \vdots \\ c_{2,y}(-2) \\ c_{2,y}(-1) \\ c_{2,y}(0) \\ c_{2,y}(1) \\ c_{2,y}(2) \\ \vdots \end{pmatrix} = \gamma_{2,s} \begin{pmatrix} \vdots & \\ 0 & 0 \\ 0 & h_0 \\ h_0 & h_1 \\ h_1 & 0 \\ 0 & 0 \\ \vdots & \end{pmatrix} \begin{pmatrix} h_0 \\ h_1 \end{pmatrix} + \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix} = \gamma_{2,s} \begin{pmatrix} \vdots & \\ 0 & 0 \\ 0 & h_1 \\ h_1 & h_0 \\ h_0 & 0 \\ 0 & 0 \\ \vdots & \end{pmatrix} \begin{pmatrix} h_1 \\ h_0 \end{pmatrix} + \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

We cannot distinguish $\begin{pmatrix} \mathbf{h}_0 \\ \mathbf{h}_1 \end{pmatrix}$ and $\begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_0 \end{pmatrix}$

3. Higher Order Statistics.

Let $y(k)$ be stationary zero-mean non Gaussian complex random process.
Define

$$y_\tau = y_\tau(k) := y(k+\tau)$$

$$y^* \text{ is conjugate to } y$$

$$\begin{aligned}
c_{1,y} &= E(y), \\
c_{2,y}(\tau_1) &= E(y y^*_{\tau_1}), \\
c_{3,y}(\tau_1, \tau_2) &= E(y y_{\tau_1} y^*_{\tau_2}), \\
c_{4,y}(\tau_1, \tau_2, \tau_3) &= E(y^* y_{\tau_1} y^*_{\tau_2} y_{\tau_3}) - E(y^* y_{\tau_1}) E(y^*_{\tau_2} y_{\tau_3}) - E(y^* y^*_{\tau_2}) E(y_{\tau_1} y_{\tau_3}) - E(y^* y_{\tau_3}) E(y_{\tau_1} y^*_{\tau_2}).
\end{aligned}$$

$$\tau_1, \tau_2, \tau_3 = 0, \pm 1, \pm 2, \ldots$$

3. Higher Order Statistics

Substitute

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h_l s(k-l) \end{cases}$$

into

$$\begin{aligned} c_{2,y}(\tau_1) &= E(yy_{\tau_1}^*), \\ c_{3,y}(\tau_1, \tau_2) &= E(yy_{\tau_1}y_{\tau_2}^*), \\ c_{4,y}(\tau_1, \tau_2, \tau_3) &= E(y^*y_{\tau_1}y_{\tau_2}^*y_{\tau_3}) - E(y^*y_{\tau_1})E(y_{\tau_2}^*y_{\tau_3}) - E(y^*y_{\tau_2}^*)E(y_{\tau_1}y_{\tau_3}) - E(y^*y_{\tau_3})E(y_{\tau_1}y_{\tau_2}^*). \end{aligned}$$

3. Higher Order Statistics

Substitute

$$\begin{cases} y(k) = x(k) + v(k), \\ x(k) = (h * s)(k) := \sum_{l=0}^{L} h_l s(k-l) \end{cases}$$

into

$$\begin{aligned} c_{2,y}(\tau_1) &= E(y y_{\tau_1}^*), \\ c_{3,y}(\tau_1, \tau_2) &= E(y y_{\tau_1} y_{\tau_2}^*), \\ c_{4,y}(\tau_1, \tau_2, \tau_3) &= E(y^* y_{\tau_1} y_{\tau_2}^* y_{\tau_3}) - E(y^* y_{\tau_1}) E(y_{\tau_2}^* y_{\tau_3}) - E(y^* y_{\tau_2}^*) E(y_{\tau_1} y_{\tau_3}) - E(y^* y_{\tau_3}) E(y_{\tau_1} y_{\tau_2}^*). \end{aligned}$$

&

Assumptions

- $s(k)$ independent, identically distributed, non Gaussian, zero mean
- $v(k)$ Gaussian, zero mean
- $s(k)$ and $v(k)$ are statistically independent.

3. Higher Order Statistics

Bartlett-Brillinger(1955) & Rosenblatt(1967) formulae

$$c_{2,y}(\tau_1) = \gamma_{2,s} \sum_{l=0}^{L} h_l \overline{h}_{l+\tau_1}, \qquad -L \leq \tau_1 \leq L$$

$$c_{3,y}(\tau_1, \tau_2) = \begin{cases} \gamma_{3,s} \sum_{l=0}^{L} h_l h_{l+\tau_1} \overline{h}_{l+\tau_2}, & -L \leq \tau_1, \tau_2 \leq L, \quad \text{nonsymmetric input} \\ 0, & \text{symmetric input} \end{cases}$$

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \; -L \leq \tau_1, \tau_2, \tau_3 \leq L$$

4. Fourth Order Statistics (Example L=1)

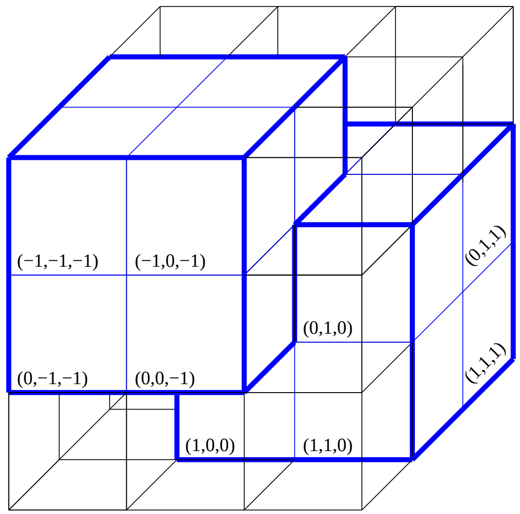
$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{1} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1$$

27 conditions

4. Fourth Order Statistics (Example L=1)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{1} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1$$

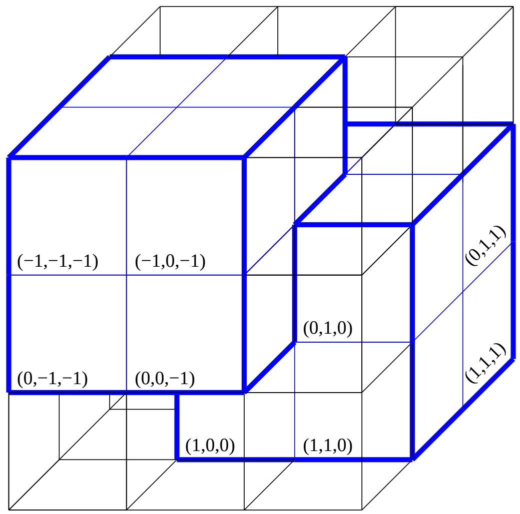
15 conditions



4. Fourth Order Statistics (Example L=1)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{1} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1$$

15 conditions



5 different conditions (complex case)

4. Fourth Order Statistics (Example L=1)

$$c_{4,y}(\tau_1,\tau_2,\tau_3)=\gamma_{4,s}\sum_{l=0}^{1}\overline{h}_l h_{l+\tau_1}\overline{h}_{l+\tau_2}h_{l+\tau_3},\quad \tau_1,\tau_2,\tau_3=0,\pm 1$$

15 conditions

$$\begin{cases} c_{-1,-1,-1}=c_{0,1,0}=\overline{c}_{0,0,1}=\overline{c}_{1,0,0} & =\gamma_{4,s}h_0^2\overline{h}_0\overline{h}_1 \\ c_{-1,0,-1}=\overline{c}_{1,0,-1} & =\gamma_{4,s}h_0^2\overline{h}_1^2 \\ c_{-1,-1,0}=c_{0,-1,-1}=c_{0,1,0}=c_{1,1,0} & =\gamma_{4,s}h_0\overline{h}_0h_1\overline{h}_1 \\ c_{-1,0,0}=c_{0,0,-1}=\overline{c}_{0,-1,0}=\overline{c}_{1,1,1} & =\gamma_{4,s}h_0\overline{h}_1^2h_1 \\ c_{0,0,0} & =\gamma_{4,s}(h_0^2\overline{h}_0^2+h_1^2\overline{h}_1^2) \end{cases}$$

Vector $\begin{pmatrix} h_0 \\ h_1 \end{pmatrix}$ can be reconstructed up to scaling factor

4. Fourth Order Statistics (General case)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1, \ldots$$

4. Fourth Order Statistics (General case)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1, \ldots$$

Number of different equations (complex-valued signals)

L	1	2	3	4	5
N	5	16	40	84	153

4. Fourth Order Statistics (General case)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1, \ldots$$

Number of different equations (complex-valued signals)

L	1	2	3	4	5
N	5	16	40	84	153

- Algebraic methods: use some of the equations

4. Fourth Order Statistics (General case)

$$c_{4,y}(\tau_1, \tau_2, \tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1, \tau_2, \tau_3 = 0, \pm 1, \ldots$$

Number of different equations (complex-valued signals)

L	1	2	3	4	5
N	5	16	40	84	153

- Algebraic methods: use some of the equations
- Optimization methods: use all equations(solve in LS sense)

4. Fourth Order Statistics (General case)

$$c_{4,y}(\tau_1,\tau_2,\tau_3) = \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3}, \quad \tau_1,\tau_2,\tau_3 = 0, \pm 1, \ldots$$

Number of different equations (complex-valued signals)

L	1	2	3	4	5
N	5	16	40	84	153

- Algebraic methods: use some of the equations
- Optimization methods: use all equations(solve in LS sence)

Cost function

$$f(h_0, h_1, \ldots, h_L) = \sum_{-L \le \tau_1,\tau_2,\tau_3 \le L} \left| c_{4,y}(\tau_1,\tau_2,\tau_3) - \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2 .$$

Cost function (depends on nonnegative $\lambda_2, \lambda_3, \lambda_4$)

$$f_{\lambda_2,\lambda_3,\lambda_4}(h) = \lambda_2 \sum_{-L\le\tau_1\le L} \left| c_{2,y}(\tau_1) - \gamma_{2,s} \sum_{l=0}^{L} h_l \overline{h}_{l+\tau_1} \right|^2 +$$

$$\lambda_3 \sum_{-L\le\tau_1,\tau_2\le L} \left| c_{3,y}(\tau_1,\tau_2) - \gamma_{3,s} \sum_{l=0}^{L} h_l h_{l+\tau_1} \overline{h}_{l+\tau_2} \right|^2$$

$$\lambda_4 \sum_{-L\le\tau_1,\tau_2,\tau_3\le L} \left| c_{4,y}(\tau_1,\tau_2,\tau_3) - \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2 .$$

- cost function proposed by Lii & Rosenblatt (1982)
- real-valued signals & $\lambda_3 = 0$ Tugnait (1987) (*'a brute-force technique'*)
- complex-valued signals & $\lambda_3 = 0$ Tugnait (1995)

Number N of different $c_{4,y}(\tau_1, \tau_2, \tau_3)$ for complex-valued signals

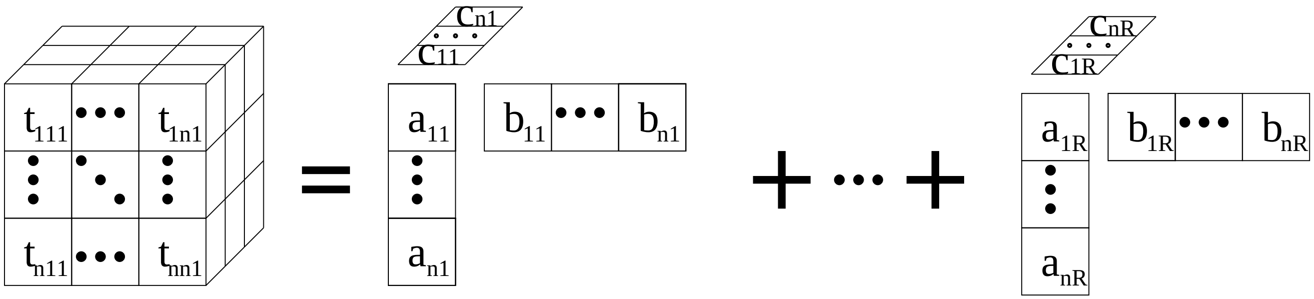
L	1	2	3	4	5
N	5	16	40	84	153

SNR is small $\Rightarrow$ $\lambda_2 := 0$,
input signal has symmetric distribution $\Rightarrow$ the term with λ_3 is absent.

5. Recall: PARAFAC/CANDECOMP decomposition

Given $T = (t_{ijk})_1^n$.

PARAFAC/CANDECOMP decomposition of T is the decomposition into a sum of R rank-one tensors:



$$A = \begin{pmatrix} a_{11} & \dots & a_{1R} \\ \vdots & \vdots & \vdots \\ a_{n1} & \dots & a_{nR} \end{pmatrix}, B = \begin{pmatrix} b_{11} & \dots & b_{1R} \\ \vdots & \vdots & \vdots \\ b_{n1} & \dots & b_{nR} \end{pmatrix}, C = \begin{pmatrix} c_{11} & \dots & c_{1R} \\ \vdots & \vdots & \vdots \\ c_{n1} & \dots & c_{nR} \end{pmatrix}$$

A, B, C are called loading factors.

Alternating Least Squares algorithm. (ALS)

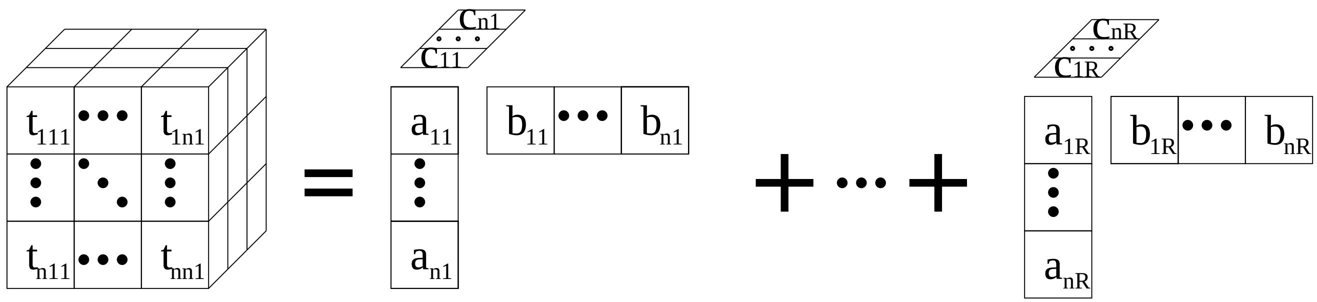
6. PC decomposition of cumulant:
C.E.R. Fernandes, G.Favier, J.C.M.Mota (2008)

$$f_{0,0,1}(h) = \sum_{-L\leq\tau_1,\tau_2,\tau_3\leq L} \left| c_{4,y}(\tau_1, \tau_2, \tau_3) - \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2$$

find minimum of $f_{0,0,1}(\cdot) \Leftrightarrow$ find P/C decomposition of $(2L+1) \times (2L+1) \times (2L+1)$ tensor $T := (c_{4,y}(\tau_1, \tau_2, \tau_3))_{-L}^{L}$

6. PC decomposition of cumulant:
C.E.R. Fernandes, G.Favier, J.C.M.Mota (2008)

Find



$A = B = \mathbf{H}$, $C = \mathbf{H}^* diag(h^*)$, where

$$\mathbf{H} = \mathbf{H}(h) = \begin{pmatrix} 0 & 0 & \dots & h_0 \\ \vdots & \vdots & \iddots & \vdots \\ 0 & h_0 & \dots & h_{L_1} \\ h_0 & h_1 & \dots & h_L \\ \vdots & \vdots & \iddots & \vdots \\ h_{L-1} & h_L & \dots & 0 \\ h_L & 0 & \dots & 0 \end{pmatrix}$$

ALS

— is very slow (many cheap iterations) and does not exploit the symmetry : $A = B$, etc.

\+ always converges monotonically

7. Single-Step Least Squares algorithm (SS-LS)

$$f_{0,0,1}(h) = \sum_{-L\le\tau_1,\tau_2,\tau_3\le L} \left| c_{4,y}(\tau_1,\tau_2,\tau_3) - \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2$$
$$= \|\gamma_{4,s}(\mathbf{H} \odot \mathbf{H} \odot \mathbf{H}^*)\mathbf{h}^* - \mathrm{vec}(\mathbf{C}_{[1]})\|^2.$$

Definition: Khatri-Rao product $\odot$:

$$\text{for } A = (a_1 \ldots a_n) \text{ and } B = (b_1 \ldots b_n),$$
$$A \odot B := (a_1 \otimes b_1 \ldots a_n \otimes b_n).$$

IDEA of SS-LS :

- normalize h_{old}
- $h_{new} = \left[(\gamma_{4,s}(\mathbf{H}(h_{old}) \odot \mathbf{H}(h_{old}) \odot \mathbf{H}(h_{old})^*))^{\#}\, \mathrm{vec}(\mathbf{C}_{[1]}) \right]^*$

SS-LS - algorithm (Fernandes Carlos Estêvão R., Favier Gérard, Mota João Cesar M)

1. build $\mathbf{H}^{(r-1)} = \mathbf{H}(1/{h_0}^{(r-1)}\mathbf{h}^{(r-1)})$
2. Compute $\mathbf{G}^{(r-1)}$ using
$$\mathbf{G}^{(r-1)} = \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)*}.$$
3. Minimize the cost function
$$\psi(\mathbf{h}^*, \mathbf{h}^{(r-1)}) = \|\mathrm{vec}(\mathbf{C}_{[1]}) - \gamma_{4,s}\mathbf{G}^{(r-1)}\mathbf{h}^*\|^2$$
so that
$$\mathbf{h}^{(r)} = \left[\gamma_{4,s}^{-1}\mathbf{G}^{(r-1)\#}\mathrm{vec}(\mathbf{C}_{[1]})\right]^*.$$
4. Iterate until $\|\mathbf{h}^{(r)} - \mathbf{h}^{(r-1)}\|/\|\mathbf{h}^{(r)}\| \leq \varepsilon$.

SS-LS - algorithm (Fernandes Carlos Estêvão R., Favier Gérard, Mota João Cesar M)

1. build $\mathbf{H}^{(r-1)} = \mathbf{H}(1/{h_0}^{(r-1)}\mathbf{h}^{(r-1)})$
2. Compute $\mathbf{G}^{(r-1)}$ using
$$\mathbf{G}^{(r-1)} = \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)*}.$$
3. Minimize the cost function
$$\psi(\mathbf{h}^*, \mathbf{h}^{(r-1)}) = \|\text{vec}(\mathbf{C}_{[1]}) - \gamma_{4,s}\mathbf{G}^{(r-1)}\mathbf{h}^*\|^2$$
so that
$$\mathbf{h}^{(r)} = \left[\gamma_{4,s}^{-1}\mathbf{G}^{(r-1)\#}\text{vec}(\mathbf{C}_{[1]})\right]^*.$$
4. Iterate until $\|\mathbf{h}^{(r)} - \mathbf{h}^{(r-1)}\|/\|\mathbf{h}^{(r)}\| \leq \varepsilon$.

ALS

— is very slow (many cheap iterations) and does not exploit the symmetry : $A = B$, etc.

\+

SS-LS

\+ fast (if converges), exploits the symmetry,

—

SS-LS - algorithm (Fernandes Carlos Estêvão R., Favier Gérard, Mota João Cesar M)

1. build $\mathbf{H}^{(r-1)} = \mathbf{H}(1/{h_0}^{(r-1)}\mathbf{h}^{(r-1)})$
2. Compute $\mathbf{G}^{(r-1)}$ using
$$\mathbf{G}^{(r-1)} = \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)} \odot \mathbf{H}^{(r-1)*}.$$
3. Minimize the cost function
$$\psi(\mathbf{h}^*, \mathbf{h}^{(r-1)}) = \|\mathrm{vec}(\mathbf{C}_{[1]}) - \gamma_{4,s}\mathbf{G}^{(r-1)}\mathbf{h}^*\|^2$$
so that
$$\mathbf{h}^{(r)} = \left[\gamma_{4,s}^{-1}\mathbf{G}^{(r-1)\#}\mathrm{vec}(\mathbf{C}_{[1]})\right]^*.$$
4. Iterate until $\|\mathbf{h}^{(r)} - \mathbf{h}^{(r-1)}\|/\|\mathbf{h}^{(r)}\| \leq \varepsilon$.

ALS

— is very slow (many cheap iterations) and does not exploit the symmetry : $A = B$, etc.

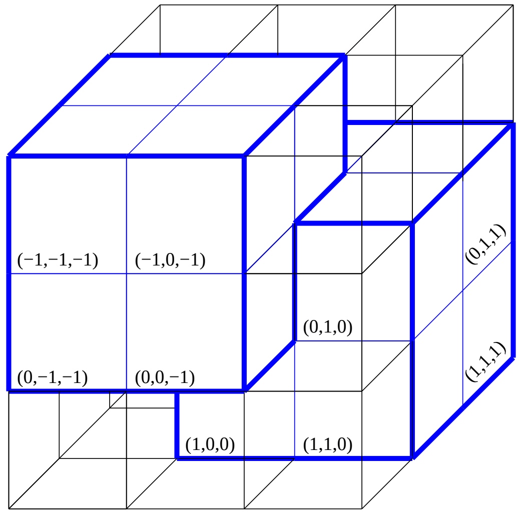
\+ but always converges monotonically

SS-LS

\+ fast (if converges), exploits the symmetry,

— convergence is not guaranted

ALS and SS-LS fail for many data(Example)1/2



For any $x, y \in \mathbb{R}$ define

$$
\begin{aligned}
c_{-1,-1,-1} = c_{0,1,0} = \overline{c}_{0,0,1} = \overline{c}_{1,0,0} &= -4 && = \gamma_{4,s} h_0^2 \overline{h}_0 \overline{h}_1 \\
c_{-1,0,-1} = \overline{c}_{1,0,-1} &= 2x + jy && = \gamma_{4,s} h_0^2 \overline{h}_1^2 \\
c_{-1,-1,0} = c_{0,-1,-1} = c_{0,1,1} = c_{1,1,0} &= 10 - jy && = \gamma_{4,s} h_0 \overline{h}_0 h_1 \overline{h}_1 \\
c_{-1,0,0} = c_{0,0,-1} = \overline{c}_{0,-1,0} = \overline{c}_{1,1,1} &= -x - 13 && = \gamma_{4,s} h_0 \overline{h}_1^2 h_1 \\
c_{0,0,0} &= -4 && = \gamma_{4,s} (h_0^2 \overline{h}_0^2 + h_1^2 \overline{h}_1^2)
\end{aligned}
$$

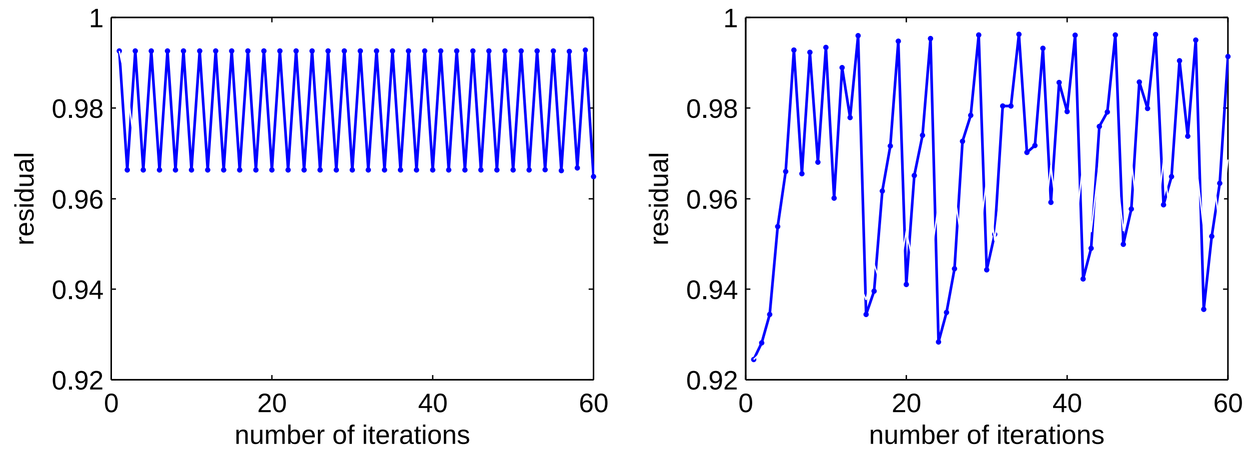
ALS and SS-LS fail for many data(Example)2/2

Let $\gamma_{4,s} = -4$.

- ALS: the loading factors are "symmetric" but have no Hankel structure
- for $\mathbf{h}^{(0)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \mathbf{h}^{(1)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

  SS-LS: $\mathbf{h}^{(0)} \to \mathbf{h}^{(1)} \to \mathbf{h}^{(0)} \to \mathbf{h}^{(1)} \ldots$

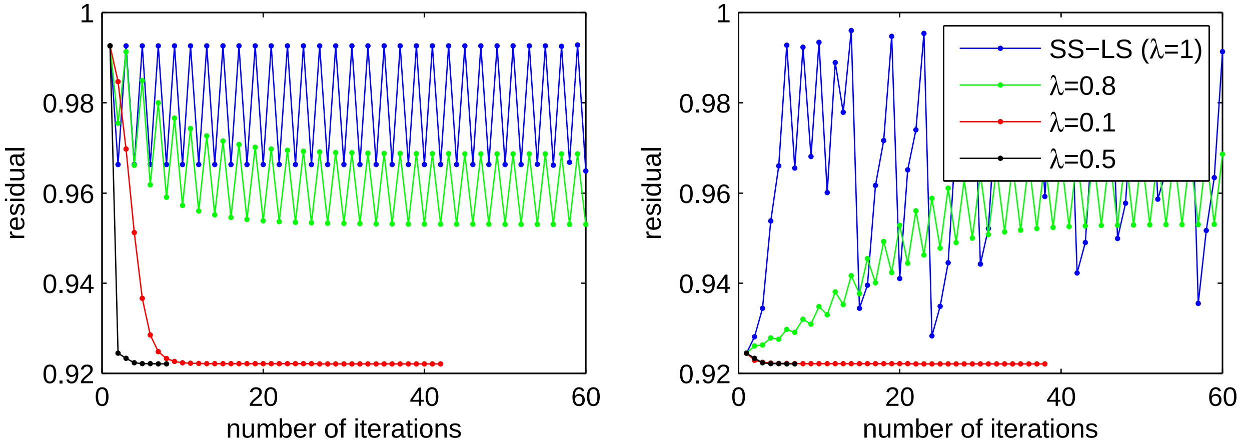
Let $x = -1$ and $y = -2$.



Krasnoselskii procedure:

$$\mathbf{h}^{(1)} \to \mathbf{h}^{(2)} \to \mathbf{h}^{(3)} \to \mathbf{h}^{(4)} \to \dots$$
$$\mathbf{h}^{(r)} \leftarrow \lambda \mathbf{h}^{(r)} + (1-\lambda)\mathbf{h}^{(r-1)}$$

with some fixed $\lambda \in [0, 1]$.



8. New representation of the cost function

Recall

$$f_{0,0,1}(h) = \sum_{-L\le\tau_1,\tau_2,\tau_3\le L} \left| c_{4,y}(\tau_1,\tau_2,\tau_3) - \gamma_{4,s}\sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1}\overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2$$

$$= \|\gamma_{4,s}\underbrace{(\mathbf{H}(h)\odot\mathbf{H}(h)\odot\mathbf{H}(h)^*)}_{\mathbf{G}(h)}\mathbf{h}^* - \mathrm{vec}(\mathbf{C}_{[1]})\|^2.$$

Proposition 1

Let V_{L+1} be the shift matrix defined in by: $\quad V_{L+1} : \mathbf{e}_{L+1}^{(L+1)} \to \mathbf{e}_{L}^{(L+1)} \to \cdots \to \mathbf{e}_{1}^{(L+1)} \to 0$;

Then

$$f(\mathbf{h}) = \gamma_{4,s}^2 \left(\|\mathbf{h}\|^8 + 2\sum_{k=1}^{L} |\langle \mathbf{h}, V_{L+1}^k \mathbf{h}\rangle|^4 \right) - 2\gamma_{4,s}\mathrm{vec}(\mathbf{C}_{[1]})^H \mathbf{G}(\mathbf{h})\mathbf{h}^* + \|\mathrm{vec}(\mathbf{C}_{[1]})\|^2.$$

9. Complex gradient of the cost function

Let $\mathbf{f}$ be a real-valued function that is analytic with respect to $\mathbf{h}$ and $\mathbf{h}^*$

We can define the complex gradient operator : $\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*}$.

h is a critical point of $\mathbf{f}$ iff $\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*} = 0$.

9. Complex gradient of the cost function

Let $\mathbf{f}$ be a real-valued function that is analytic with respect to $\mathbf{h}$ and $\mathbf{h}^*$

We can define the complex gradient operator : $\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*}$.

h is a critical point of $\mathbf{f}$ iff $\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*} = 0$.

Proposition 2

The complex gradient of

$$f_{0,0,1}(h) = \sum_{-L \leq \tau_1, \tau_2, \tau_3 \leq L} \left| c_{4,y}(\tau_1, \tau_2, \tau_3) - \gamma_{4,s} \sum_{l=0}^{L} \overline{h}_l h_{l+\tau_1} \overline{h}_{l+\tau_2} h_{l+\tau_3} \right|^2$$

$$= \|\gamma_{4,s} (\underbrace{\mathbf{H}(h) \odot \mathbf{H}(h) \odot \mathbf{H}(h)^*}_{\mathbf{G}(h)}) \mathbf{h}^* - \text{vec}(\mathbf{C}_{[1]})\|^2.$$

is

$$\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*} = 4\gamma_{4,s}^2 \left[\mathbf{G}(\mathbf{h})^H \mathbf{G}(\mathbf{h}) \right]^* \mathbf{h} - 4\gamma_{4,s} \left[\mathbf{G}(\mathbf{h})^H \text{vec}(\mathbf{C}_{[1]}) \right]^*.$$

10. Comparison of the complex gradient with the part of SS-LS algorithm

Step 3. of SS-LS algorithm
Minimize the cost function

$$\psi(\mathbf{h}^*, \mathbf{h}^{(r-1)}) = \|\mathrm{vec}(\mathbf{C}_{[1]}) - \gamma_{4,s}\mathbf{G}^{(r-1)}\mathbf{h}^*\|^2$$

so that

$$\mathbf{h}^{(r)} = (\gamma_{4,s}^{-1}\mathbf{G}^{(r-1)\#}\mathrm{vec}(\mathbf{C}_{[1]})^*.$$

complex gradient

$$\frac{\partial \mathbf{f}}{\partial \mathbf{h}^*} = 4\gamma_{4,s}^2 \left[\mathbf{G}(\mathbf{h})^H\mathbf{G}(\mathbf{h})\right]^* \mathbf{h} - 4\gamma_{4,s}\left[\mathbf{G}(\mathbf{h})^H\mathrm{vec}(\mathbf{C}_{[1]})\right]^*.$$

11. Convergence of SS-LS algorithm

Proposition 3
Let SS-LS algorithm converge to $\mathbf{h}^\infty := (\mathbf{h}_0^\infty, \ldots, \mathbf{h}_L^\infty)$. Then $\mathbf{h}_0^\infty \in \mathbb{R}$ and

1. if $\mathbf{h}_0^\infty < 0$ then $\mathbf{h}^\infty$ is not a global minimum of f;
2. if $\mathbf{h}_0^\infty > 0$ then $\mathbf{h}^\infty$ is proportional to some local minimum of f.

12. Fixed point interpretation of SS-LS algorithm

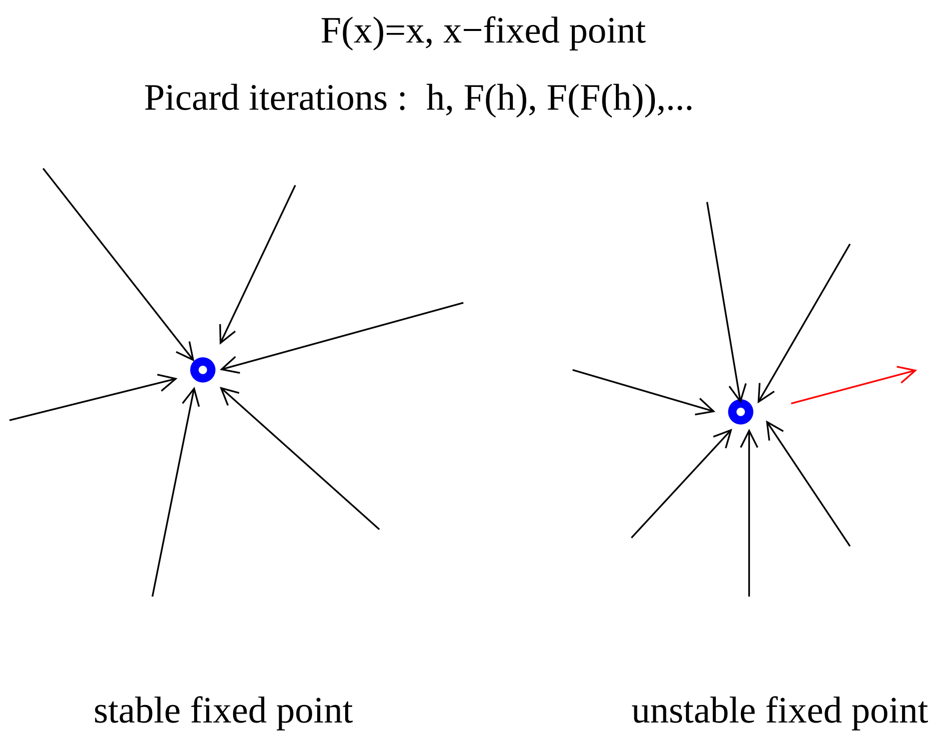
Step 3) of SS-LS can be rewritten as

$$\mathbf{h}^{(r)} = \Phi(\mathbf{h}^{(r-1)}) := F(\mathbf{h}^{(r-1)}/\mathbf{h}_0^{(r-1)}), \quad r = 1, 2, \ldots$$

where

$$F(\mathbf{h}) := \left[\gamma_{4,s}^{-1} \left(\mathbf{G}(\mathbf{h})^H \mathbf{G}(\mathbf{h})\right)^{-1} \mathbf{G}(\mathbf{h})^H \mathrm{vec}(\mathbf{C}_{[1]})\right]^* .$$

Thus, the channel estimated by SS-LS algorithm is a fixed point of Φ.



Classical replacement

$$\mathbf{h}^{(r)} = \Phi(\mathbf{h}^{(r-1)}) \qquad -Picard \quad iterations$$

by

$$\mathbf{h}^{(r)} = \Phi_\lambda(\mathbf{h}^{(r-1)}) := \mathbf{h}^{(r-1)}(1-\lambda) + \Phi(\mathbf{h}^{(r-1)})\lambda \qquad -Krasnoselskii \quad iterations$$

does not work.

Algorithm SS-LS-K$_\lambda$

Choose $\mathbf{h}^{(0)}$, $\lambda \in [0, 1]$;
$r \leftarrow 0$;

1. update $r \leftarrow r + 1$;

$$\mathbf{h}^{(r)} \leftarrow \frac{\mathbf{h}^{(r-1)}}{\|\mathbf{h}^{(r-1)}\|}(1 - \lambda) + \frac{F(\mathbf{h}^{(r-1)})}{\|F(\mathbf{h}^{(r-1)})\|}\lambda$$

2. iterate until $\|\mathbf{h}^{(r)} - \mathbf{h}^{(r-1)}\| / \|\mathbf{h}^{(r-1)}\| < \varepsilon$.

13. Enhanced Plane Search Procedure (EPS) 1/4

$$\min_{\mathbf{z}\in\mathbb{C}^m} f(\mathbf{z})$$
$$\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots, \mathbf{z}^{(k)}, \mathbf{z}^{(k+1)}, \ldots$$

Enhanced Line Search (ELS):

$$\mathbf{z}_{k+1} = (1-\lambda_{k+1})\mathbf{z}_k + \lambda_{k+1}\mathbf{z}_{k+1},$$

where

$$\lambda_{k+1} = \underset{\lambda\in\mathbb{R}(\text{resp. }\mathbb{C})}{\operatorname{argmin}} f((1-\lambda_{k+1})\mathbf{z}_k + \lambda_{k+1}\mathbf{z}_{k+1}). \tag{1}$$

Enhanced Plane Search (EPS):

$$\mathbf{z}_{k+1} = \mu_{k+1}\mathbf{z}_k + \lambda_{k+1}\mathbf{z}_{k+1},$$

where

$$(\mu_{k+1}, \lambda_{k+1}) = \underset{\mu,\lambda\in\mathbb{R}(\text{resp. }\mathbb{C})}{\operatorname{argmin}} f(\mu\mathbf{z}_k + \lambda\mathbf{z}_{k+1}). \tag{2}$$

in many cases(f.e. PC) cost of (1)$\approx$ cost of (2)

13. Enhanced Plane Search Procedure (EPS) 2/4

Lemma

Let

$$f(\mathbf{z}) := \|\mathbf{u}(\mathbf{z}, \mathbf{z}^*) - \mathbf{b}\|^2$$
$$\mathbf{b} \in \mathbb{C}^n, \mathbf{z}_1, \mathbf{z}_2 \in \mathbb{C}^n \text{ be given}$$
$$\mathbf{u}(\mathbf{z}, \mathbf{z}^*) : \mathbb{C}^m \to \mathbb{C}^n$$
$$\mathbf{u}(\lambda\mathbf{z}, \overline{\lambda}\mathbf{z}^*) = g(\lambda, \overline{\lambda})\mathbf{u}(\mathbf{z}, \mathbf{z}^*)$$

Then the problems

$$\lambda_{k+1} = \underset{\lambda \in \mathbb{R}(\text{resp. } \mathbb{C})}{\operatorname{argmin}} f((1-\lambda)\mathbf{z}_1 + \lambda\mathbf{z}_2),$$
$$(\mu_{k+1}, \lambda_{k+1}) = \underset{\mu, \lambda \in \mathbb{R}(\text{resp. } \mathbb{C})}{\operatorname{argmin}} f(\mu\mathbf{z}_1 + \lambda\mathbf{z}_2)$$

have approximately the same cost.

Rajih Myriam, Comon Pierre and Harshman Richard A.,*Enhanced Line Search: A Novel Method to Accelerate PARAFAC*, SIAM J. Matrix Anal. Appl., 2008, v.30, N.3, pp.1128–1147.

Nion Dimitri, De Lathauwer Lieven, *An enhanced line search scheme for complex-valued tensor decompositions. Application in DS-CDMA*, Signal Processing, 2008, v.88, N.3, pp. 749 - 755.

13. Enhanced Plane Search Procedure (EPS) 3/4

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}\in\mathbb{C}^{n\times R}} \|\mathbf{A}(\mathbf{C}\odot\mathbf{B})^T - \mathbf{T}\|^2,$$

$(\Delta\mathbf{A}, \Delta\mathbf{B}, \Delta\mathbf{C})$ search direction

$$\min_{\lambda\in\mathbb{C}} \|(\mathbf{A}+\lambda\Delta\mathbf{A})\left[(\mathbf{C}+\lambda\Delta\mathbf{C})\odot(\mathbf{B}+\lambda\Delta\mathbf{B})\right]^T - \mathbf{T}\|^2, \tag{3}$$

$$\min_{\mu,\lambda\in\mathbb{C}} \|(\mu\mathbf{A}+\lambda\Delta\mathbf{A})\left[(\mu\mathbf{C}+\lambda\Delta\mathbf{C})\odot(\mu\mathbf{B}+\lambda\Delta\mathbf{B})\right]^T - \mathbf{T}\|^2. \tag{4}$$

Let us set

$$\mathbf{z} := \mathbf{z}_1 := \begin{pmatrix}\mathbf{Vec}(\mathbf{A})\\ \mathbf{Vec}(\mathbf{B})\\ \mathbf{Vec}(\mathbf{C})\end{pmatrix}, \quad \mathbf{u}(\mathbf{z},\mathbf{z}^*) := \mathbf{Vec}(\mathbf{A}(\mathbf{C}\odot\mathbf{B})^T),$$

$$\mathbf{z}_2 := \begin{pmatrix}\mathbf{Vec}(\Delta\mathbf{A})\\ \mathbf{Vec}(\Delta\mathbf{B})\\ \mathbf{Vec}(\Delta\mathbf{C})\end{pmatrix}, \quad \mathbf{b} := \mathbf{Vec}(\mathbf{T}).$$

Then

$$\mathbf{u}(\lambda\mathbf{z}, \overline{\lambda}\mathbf{z}^*) = g(\lambda,\overline{\lambda})\mathbf{u}(\mathbf{z},\mathbf{z}^*) \tag{5}$$

with $g(\lambda,\overline{\lambda}) := \lambda^3$.

13. Enhanced Plane Search Procedure (EPS) 4/4

Suppose additionally that $\mathbf{C} = \mathbf{A}^H$

$$\min_{\mathbf{A},\mathbf{B}\in\mathbb{C}^{n\times R}} \|\mathbf{A}(\mathbf{A}^H \odot \mathbf{B})^T - \mathbf{T}\|^2,$$

$(\Delta\mathbf{A}, \Delta\mathbf{B})$ search direction

$$\min_{\lambda\in\mathbb{C}} \|(\mathbf{A} + \lambda\Delta\mathbf{A})\left[(\mathbf{A}^H + \overline{\lambda}\Delta\mathbf{A}^H) \odot (\mathbf{B} + \lambda\Delta\mathbf{B})\right]^T - \mathbf{T}\|^2, \tag{6}$$

$$\min_{\mu,\lambda\in\mathbb{C}} \|(\mu\mathbf{A} + \lambda\Delta\mathbf{A})\left[(\mu\mathbf{A}^H + \lambda\Delta\mathbf{A}^H) \odot (\mu\mathbf{B} + \lambda\Delta\mathbf{B})\right]^T - \mathbf{T}\|^2. \tag{7}$$

Let us set

$$\mathbf{z} := \mathbf{z}_1 := \begin{pmatrix}\mathbf{Vec}(\mathbf{A})\\ \mathbf{Vec}(\mathbf{B})\end{pmatrix}, \quad \mathbf{u}(\mathbf{z}, \mathbf{z}^*) := \mathbf{Vec}(\mathbf{A}(\mathbf{A}^H \odot \mathbf{B})^T),$$
$$\mathbf{z}_2 := \begin{pmatrix}\mathbf{Vec}(\Delta\mathbf{A})\\ \mathbf{Vec}(\Delta\mathbf{B})\end{pmatrix}, \quad \mathbf{b} := \mathbf{Vec}(\mathbf{T}).$$

Then

$$\mathbf{u}(\lambda\mathbf{z}, \overline{\lambda}\mathbf{z}^*) = g(\lambda, \overline{\lambda})\mathbf{u}(\mathbf{z}, \mathbf{z}^*) \tag{8}$$

with $g(\lambda, \overline{\lambda}) := \lambda^2\overline{\lambda}$.

Simulations

- $L = 5$
- $h = \ldots$
- lenght of the input = 3000 (QPSK)
- SNR=0 dB
- $\varepsilon = 0.001$.

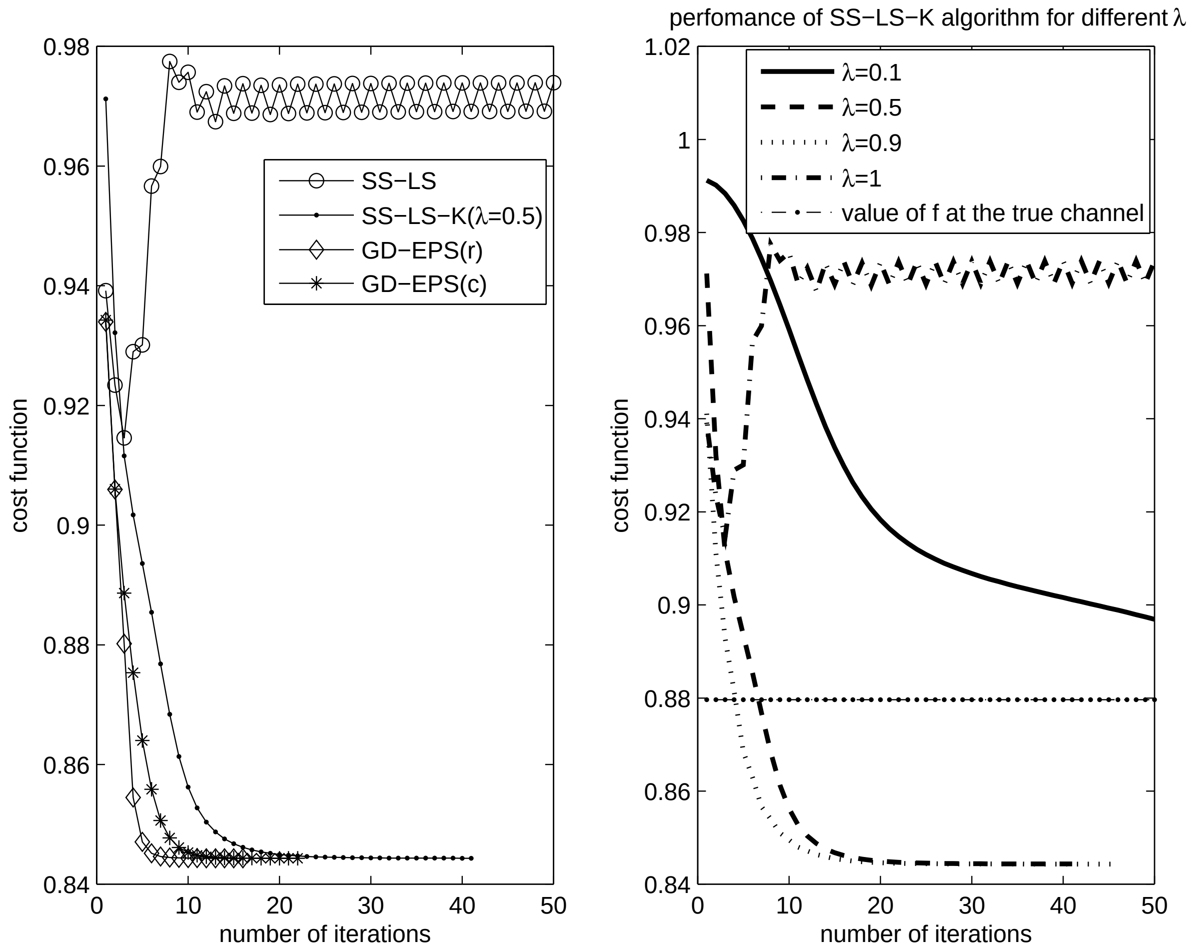
We compare

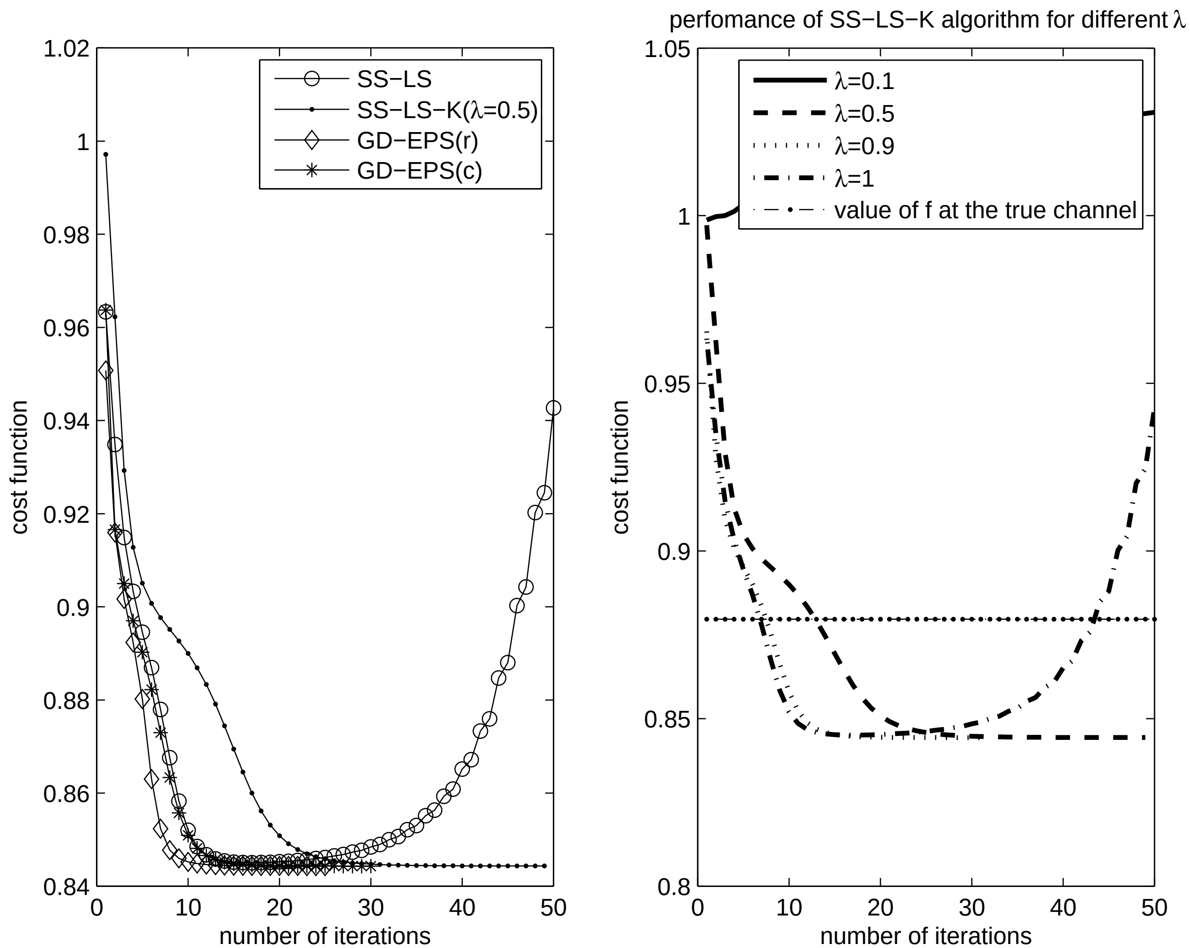
SS-LS $\mathbf{h}^{(r)} \leftarrow F\left(\frac{\mathbf{h}^{(r-1)}}{\mathbf{h}_0^{(r-1)}}\right)$, where $F(\mathbf{h}) := \left[\gamma_{4,s}^{-1}\left(\mathbf{G}(\mathbf{h})^H\mathbf{G}(\mathbf{h})\right)^{-1}\mathbf{G}(\mathbf{h})^H\mathrm{vec}(\mathbf{C}_{[1]})\right]^*$;

SS-LS-K$_\lambda$ $\mathbf{h}^{(r)} \leftarrow \frac{\mathbf{h}^{(r-1)}}{\|\mathbf{h}^{(r-1)}\|}(1-\lambda) + \frac{F(\mathbf{h}^{(r-1)})}{\|F(\mathbf{h}^{(r-1)})\|}\lambda$;

GD-EPS-$\mathbb{R}$ $\mathbf{h}^{(r)} \leftarrow \lambda_1\mathbf{h}^{(r-1)} + \lambda_2\frac{\partial\mathbf{f}}{\partial\mathbf{h}^*}\Big|_{\mathbf{h}=\mathbf{h}^{(r-1)}}$, λ_1 and λ_2 are real;

GD-EPS-$\mathbb{C}$ $\mathbf{h}^{(r)} \leftarrow \lambda_1\mathbf{h}^{(r-1)} + \lambda_2\frac{\partial\mathbf{f}}{\partial\mathbf{h}^*}\Big|_{\mathbf{h}=\mathbf{h}^{(r-1)}}$, λ_1 and λ_2 are complex.

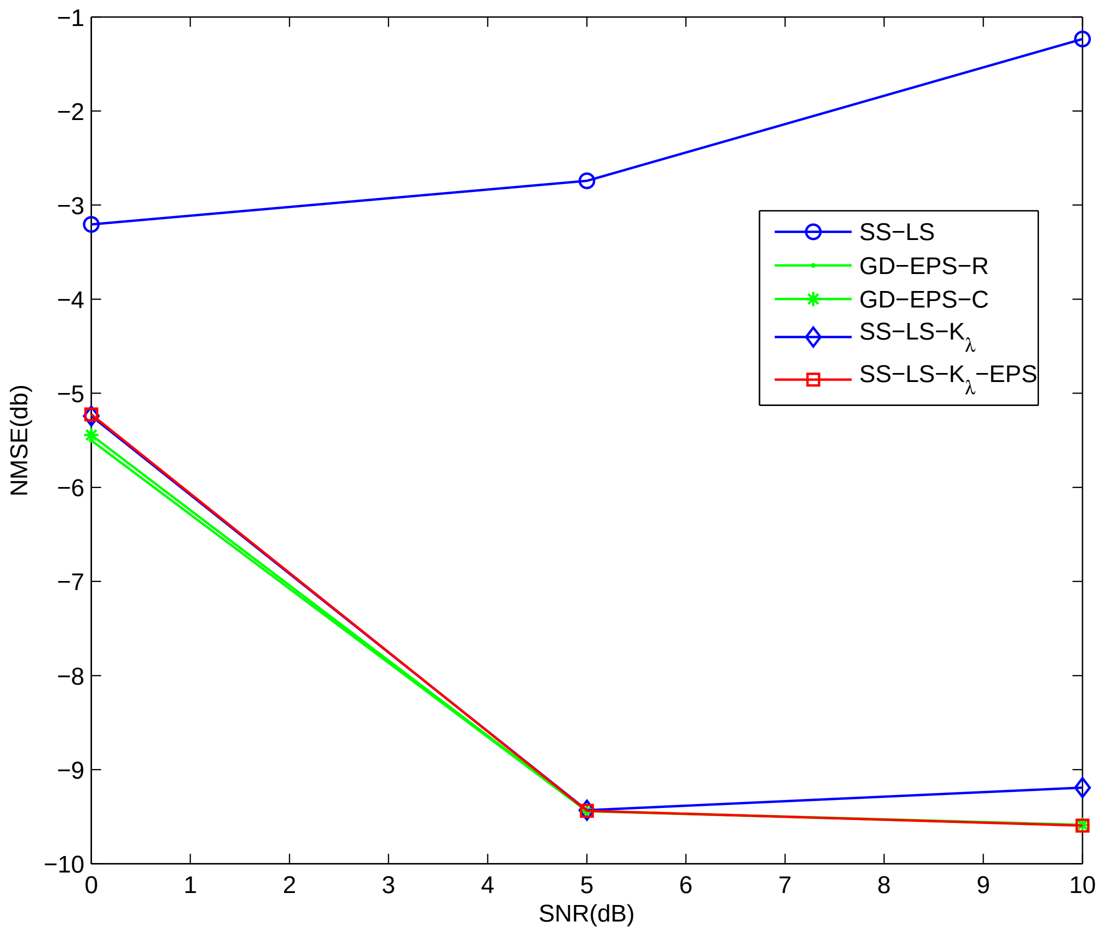

perfomance of SS−LS−K algorithm for different λ
SS−LS
SS−LS−K(λ=0.5)
GD−EPS(r)
GD−EPS(c)
λ=0.1
λ=0.5
λ=0.9
λ=1
value of f at the true channel
cost function
number of iterations
0.98
0.96
0.94
0.92
0.9
0.88
0.86
0.84
1.02
1
0
10
20
30
40
50


SS−LS
SS−LS−K(λ=0.5)
GD−EPS(r)
GD−EPS(c)
cost function
number of iterations
perfomance of SS−LS−K algorithm for different λ
λ=0.1
λ=0.5
λ=0.9
λ=1
value of f at the true channel
cost function
number of iterations

$$s(n) = \text{sign}(\sin n^2) + j\text{sign}(\cos n^2), \qquad n = 1, \ldots, 2000$$

$$h = (1, -0.25 + 1.2j, 0.5, 1 + 0.33j)$$

50 simulation of noise with SNR = 0,5,10.



Conclusions

- We considered the problem of blind estimation of FIR channels. The channels parameters were estimated by nonlinear optimization of a quadratic cumulant matching criterion involving only fourth order cumulants.
- We found a new representation of the cost function and explicit expression of complex gradient.
- We explored convergence properties of SS-LS algorithm.
- We proposed "Krasnoselskii" version of SS-LS algorithm and compared it with SS-LS and Gradien Descent algorithm improved by EPS with real and complex steps.

Thank you for your attention.