Short Overview on Blind Equalization

Philippe Ciblat

Télécom ParisTech

Outline

1. Introduction
   - General problem
   - Problem classification
   - Considered problem

2. Statistical framework

3. High-Order Statistics (HOS) algorithms
   - Constant Modulus Algorithm (CMA)
   - Adaptive versions

4. Second-Order Statistics (SOS) algorithms
   - Covariance matching algorithm
   - (Deterministic) Maximum-Likelihood algorithm
   - Some sub-optimal algorithms

5. Other types of algorithms

6. Numerical illustrations: optical-fiber communications use-case

Part 1: Introduction

General model

Unknown signal mixture with additive noise

$$\mathbf{y}(n) = \mathbf{fct}(\mathbf{s}(n)) + \mathbf{w}(n) \quad (1)$$

with

- $\mathbf{y}(n)$: observations vector at time-index n
- $\mathbf{w}(n)$: white Gaussian noise with zero-mean

Find out the multi-variate input $\mathbf{s}(n)$ given

- only a set of observations $\mathbf{y}(n)$
- statistical model for the noise

Blind techniques

Unknown fct without deterministic help of $\mathbf{s}(n)$ to estimate it

Problem classification

- $\mathbf{s}(n)$ belongs to a discrete set: **equalization**
  - Military applications: passive listening
  - Civilian applications: no training sequence
    - Goal 1: remove the header and increase the data rate (be careful: with the same raw data rate)
    - Goal 2: follow very fast variation of wireless channel (be careful: set of observations is small)
- $\mathbf{s}(n)$ belongs to a uncountable set: **source separation**
  - Audio (cocktail party)

The cocktail party effect is the phenomenon of being able to focus one's auditory attention on a particular stimulus while filtering out a range of other stimuli, much the same way that a partygoer can focus on a single conversation in a noisy room.

  - Hyperspectral imaging
  - Cosmology (Cosmic Microwave Background map with Planck data)

Problem classification (cont'd)

In the context of **Blind Source Separation (BSS)**:

- Instantaneous mixture:

$$\mathbf{y}(n) = \mathbf{H}\mathbf{s}(n) + \mathbf{w}(n)$$

  with a unknown matrix $\mathbf{H}$

- Convolutive mixture:

$$\mathbf{y}(n) = \sum_{\ell=0}^{L} \mathbf{H}(\ell)\mathbf{s}(n-\ell) + \mathbf{w}(n)$$

  with a unknown set of matrices $\mathbf{H}(\ell)$

- Nonlinear mixture: **fct** is not linear

BSS field

- Vast community mainly working on the instantaneous case
- Goal: find out $\mathbf{s}(n)$ up to scale and permutation operators

Considered Problem

Go back to **equalization** (done in blindly manner)

Unlike BSS, sources are strongly structured:

- discrete set (often a lattice, i.e., $\mathbb{Z}$-module)
- discrete set with specific properties: constant modulus if PSK
- man-made source (can be even modify to help the blind equalization step)

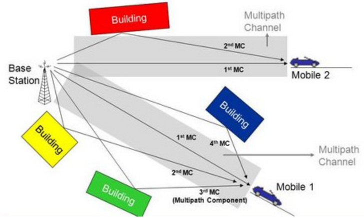
Classification problem rather than Regression problem

First questions

- Do we have a Input/Output model given by Eq. (1)?
- If yes, what is the shape of the mixture given by **fct**?

Signal model

- Single-user context
- Single-antenna context
- Multipath propagation channel



Equivalent discrete-time channel model (by sampling EM wave at the symbol rate)

$$y(n) = \sum_{\ell} h(\ell)s(n-\ell) + w(n), \forall n = 0, \ldots, N-1 \Leftrightarrow \mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{w}$$

where **H** is a band-Toeplitz matrix, N is the frame size

Signal model (cont'd)

Sampling at symbol rate leads to

- no information loss on the symbol sequence
- but information loss on the electro-magnetic wave, and probably on the channel impulse response (our goal, here)

Go back to the "true" receive signal...

$$y(t) = \sum_n s(n)h(t - nT_s) + w(t), \ \forall t \in \mathbb{R}$$

with

- $s(n)$: symbol sequence
- $w(t)$: white Gaussian noise
- $h(t)$: filter coming from the channel and the transmitter

$$\text{occupied band} = \left[-\frac{1+\rho}{2T_s}, \frac{1+\rho}{2T_s}\right]$$

with the roll-off factor $\rho \in (0, 1]$

Signal framework

Shannon-Nyquist sampling theorem $\Rightarrow T = \frac{T_s}{2}$

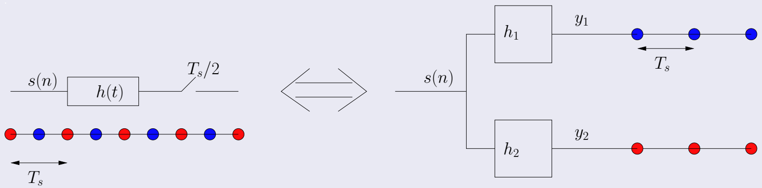
- Scalar framework: no filtering anymore

$$\tilde{y}(n) = y(nT) = \sum_k s(knT_s/2 - kT_s) + \tilde{w}(n)$$

- Vector framework: SIMO filtering

$$\begin{cases} y_1(n) = y(nT_s) = h_1 \star s(n) + w_1(n) \\ y_2(n) = y(nT_s + T_s/2) = h_2 \star s(n) + w_2(n) \end{cases}$$

with $h_1(n) = h(nT_s)$ and $h_2(n) = h(nT_s + T_s/2)$



Problems to be solved

Goals

Estimate

1. **Scalar case:** h_1 given $y_1(n)$ only and h_2 given $y_2(n)$ only, i.e., working with model of Slide 7
2. **Vector case:** $\mathbf{h} = [h_1, h_2]^{\mathrm{T}}$ given $\mathbf{y}(n) = [y_1(n), y_2(n)]^{\mathrm{T}}$ jointly

Glossary:

- without training sequence
  ⇝ Non-Data-aided (NDA) or blind/unsupervised
- with training sequence
  ⇝ Data-aided (DA) or supervised
- with decision-feedback
  ⇝ Decision-Directed (DD)

Part 2: Statistical framework

Available data statistics

- Only $\{\mathbf{y}(n)\}_{n=0}^{N-1}$ is available to estimate $\mathbf{H}$
- What is an algorithm here? a function depending only on $\{\mathbf{y}(n)\}_{n=0}^{N-1}$...

... a statistic of the random process $\mathbf{y}(n)$

$$\Theta\left(\{\mathbf{y}(n)\}_{n=0}^{N-1}\right)$$

Choice of Θ:

- *P*-order polynomial: moments of the random process
  Question: which orders are relevant? listen to the talk
- A Deep Neural Network (DNN)
  Question: how calculating the weights? see Slide 37

A not-so toy example

$$\mathbf{y}(n) = \mathbf{H}\mathbf{s}(n) + \mathbf{w}(n)$$

with

- $\mathbf{y}(n)$ is a vector of length L
- $\mathbf{H}$ is a $L \times L$ square full rank matrix
- $\mathbf{s}(n)$, $\mathbf{w}(n)$ are i.i.d. circularly-symmetric Gaussian vectors with zero-mean and variances σ_s^2 and σ_w^2 respectively

Results

- $\mathbf{y}(n)$ Gaussian with zero-mean and correlation matrix $\mathbf{R}(\mathbf{H}) = \sigma_s^2 \mathbf{H}\mathbf{H}^{\mathrm{H}} + \sigma_w^2 \mathbf{Id}_L$
- $\mathbf{R}(\mathbf{H}) = \mathbf{R}(\mathbf{H}\mathbf{U})$ for any unitary matrix $\mathbf{U}$
- Principal Component Analysis (PCA) is a **deadlock**

$\mathbf{s}(n)$ has to be non-Gaussian
$\Rightarrow$ Independent Component Analysis (ICA)

Scalar case

Go back to blind equalization

$$y(n) = h \star s(n) + w(n)$$

As $y(n)$ is stationary, second-order information lies in

$$S(e^{2i\pi f}) = \sum_m r(m) e^{-2i\pi fm} = \sigma_s^2 |h(e^{2i\pi f})|^2 + \sigma_w^2$$

with

- $r(m) = \mathbb{E}[y(n+m)\overline{y(n)}]$
- $h(\mathfrak{z}) = \sum_\ell h(\ell)\mathfrak{z}^{-\ell}$, with $\mathfrak{z} = e^{2i\pi f}$

Results

- Lack of information on the channel impulse response, except if
  - $h(\mathfrak{z})$ is phase minimum ($h(\mathfrak{z}) \neq 0$ if $|\mathfrak{z}| > 1$)
  - non-stationary signal
  - non-Gaussian signal (by resorting to high-order statistics) : OK for PAM, PSK, QAM sources

Scalar case: the pavement of the HOS road

Let $X = [X_1, \ldots, X_N]$ be a real-valued random vector of length N.

Characteristic function of the first kind (MGF)

$$\Psi_X \ : \boldsymbol{\omega} \mapsto \mathbb{E}[e^{i\boldsymbol{\omega}^{\mathrm{T}}\mathbf{x}}] \quad \left(= \int p_X(\mathbf{x}) e^{i\boldsymbol{\omega}^{\mathrm{T}}\mathbf{x}} d\mathbf{x} \right)$$

Moments (of order s) $\propto$ component of Taylor series expansion of Ψ_X for s-th order

Example: $N = 2$; Second-order means $\mathbb{E}[X_1^2]$,$\mathbb{E}[X_2^2]$, and $\mathbb{E}[X_1 X_2]$

Characteristic function of the second kind (CGF)

$$\Phi_X \ : \boldsymbol{\omega} \mapsto \log(\Psi_X(\boldsymbol{\omega}))$$

Cumulants (of order s) $\propto$ component of Taylor series expansion of Φ_X for s-th order

Useful properties

- Why cumulants? let X and Y be independent vectors

$$\Psi_{[X,Y]}(\omega) = \Psi_X(\omega_1).\Psi_Y(\omega_2) \text{ but } \Phi_{[X,Y]}(\omega) = \Phi_X(\omega_1) + \Phi_Y(\omega_2)$$

- $X = [X_1, \cdots, X_N]$ and $Y = [Y_1, \cdots, Y_N]$ be independent vectors

$$\text{cum}_s(X_{i_1}+Y_{i_1}, \cdots, X_{i_s}+Y_{i_s}) = \text{cum}_s(X_{i_1}, \cdots, X_{i_s}) + \text{cum}_s(Y_{i_1}, \cdots, Y_{i_s})$$

- $X = [X_1, \cdots, X_N]$ with at least two independent components

$$\text{cum}_N(X_1, \cdots, X_N) = 0$$

- $X = [X_1, \cdots, X_N]$ Gaussian vector

$$\text{cum}_s(X_{i_1}, \cdots, X_{i_s}) = 0 \quad \text{if} \quad s \geq 3$$

Remarks

- No HOS information for Gaussian vector
- "Distance" to the Gaussian distribution ⇒ (normalized) Kurtosis

$$\kappa_x = \frac{\text{cum}_4(x, \overline{x}, x, \overline{x})}{(\mathbb{E}[|x|^2])^2}$$

Fourth-order information: the trispectrum

$$S_4(e^{2i\pi f_1}; e^{2i\pi f_2}; e^{2i\pi f_3}) = \sum_{m_1,m_2,m_3} \mathrm{cum}_4(m_1, m_2, m_3) e^{-2i\pi(f_1 m_1 + f_2 m_2 + f_3 m_3)}$$
$$= \kappa_s h(e^{2i\pi f_1})\overline{h(e^{2i\pi f_2})h(e^{2i\pi f_3})}h(e^{2i\pi(-f_1+f_2+f_3)})$$

with $\mathrm{cum}_4(m_1, m_2, m_3) = \mathrm{cum}(y(n), y(n+m_1), \overline{y(n-m_2)}, \overline{y(n-m_3)})$

Remarks

- Trispectrum provides information enough on channel impulse response
- Question: how carrying out algorithms using it (see Part 3)

Vector case

Go back to the signal model

$$\mathbf{y}(n) = \mathbf{h} \star s(n) + \mathbf{w}(n)$$

with $\mathbf{y}(n) = [y_1(n), y_2(n)]^{\mathrm{T}}$ and $\mathbf{h}(n) = [h_1(n), h_2(n)]^{\mathrm{T}}$

Reminder: oversampling or symbol rate sampling with two RX

As $\mathbf{y}(n)$ is stationary, second-order information lies in

$$\mathbf{S}(e^{2i\pi f}) = \sum_m \mathbf{R}(m) e^{-2i\pi f m} = \sigma_s^2 \mathbf{h}(e^{2i\pi f}) \mathbf{h}(e^{2i\pi f})^{\mathrm{H}}$$

with $\mathbf{R}(m) = \mathbb{E}\left[\mathbf{y}(n+m)\mathbf{y}(n)^{\mathrm{H}}\right]$ and $\mathbf{h}(e^{2i\pi f}) = \sum_\ell \mathbf{h}(\ell) e^{-2i\pi f \ell}$

Results

- Unique solution if $\mathbf{h}(\mathfrak{z})$ is phase minimum ($\mathbf{h}(\mathfrak{z}) \neq 0$ if $|\mathfrak{z}| > 1$)
- Unrestrictive assumption since often $h_1(\mathfrak{z}) \neq h_2(\mathfrak{z}), \forall \mathfrak{z}$, i.e., no common root, i.e., $h_1(\mathfrak{z})$ and $h_2(\mathfrak{z})$ are prime jointly
- Information enough on channel impulse response

Vector case: a cyclostationarity point-of-view

Go back to the continuous-time signal model

$$y(t) = \sum_k s(k)h(t - kT_s) + w(t)$$

Its autocorrelation is periodic with period T_s

$$t \mapsto r(t, \tau) = \mathbb{E}\left[y_a(t+\tau)\overline{y_a(t)}\right]$$

Result

- $\tilde{y}(n)$ cyclostationary with period $(T_s/T) = 2$
- By denoting $\tilde{s} = (s_0, 0, s_1, 0, \cdots)$, we have

$$\tilde{y}(n) = \tilde{h} \star \tilde{s}_n$$

Remark : Cyclostationary discrete-time signal with period 1 is stationary

Cyclostationary second-order information

Fourier series expansion of the correlation:

$$n \mapsto r(n,m) = \mathbb{E}[\tilde{y}(n+m)\overline{\tilde{y}(n)}] = r^{(0)}(m) + r^{(1/2)}(m)e^{2i\pi(1/2)n}$$

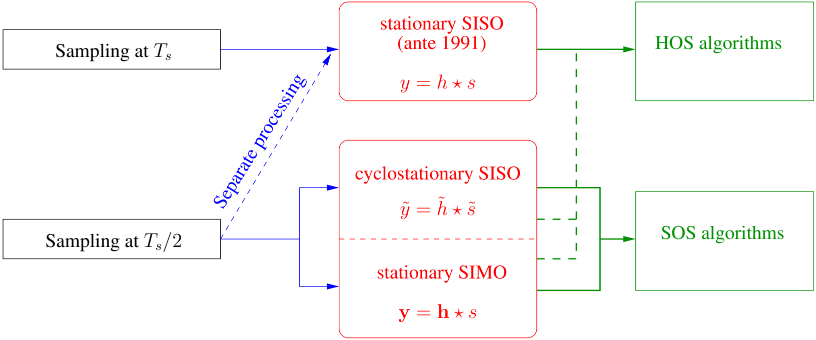
- $\alpha \in \{0, 1/2\}$: cyclic frequencies
- $\{r^{(\alpha)}(m)\}_m$: set of cyclic correlation at cyclic frequency α
- $S^{(\alpha)}(e^{2i\pi f}) = \sum_m r^{(\alpha)}(m)e^{-2i\pi fm}$: cyclic spectrum at cyclic frequency α

Results

$$S^{(0)}(e^{2i\pi f}) = \sigma_s^2|\tilde{h}(e^{2i\pi f})|^2, \; S^{(\frac{1}{2})}(e^{2i\pi f}) = \sigma_s^2\tilde{h}(e^{2i\pi f})\overline{\tilde{h}(e^{2i\pi(f+\frac{1}{2})})}$$

- Cyclic spectra provide information enough on channel impulse response
- Question: how carrying out algorithms using it (see Part 4)

Take-home message



Part 3: High-Order Statistics based Algorithms

Principle

- Usually the algorithms rely on blind deconvolution principle, i.e., retrieving the symbol sequence $\{s(n)\}_n$ directly from $\{y(n)\}_n$
- Talk done with the stationary SISO model

$$\min_p \mathbb{E}\left[f(z(n))\right]$$

with

- $z(n) = p \star y(n)$
- p the equalizer filter
- f a nonlinear and nonquadratic cost function

Some algorithms

Sato Algorithm [Sato1975]

$$J = \mathbb{E}\left[(z(n) - \text{sign}(z(n)))^2\right]$$

Constant Modulus Algorithm (CMA) [Godard1980]

$$J = \mathbb{E}\left[\left(|z(n)|^2 - C\right)^2\right]$$

with $C = \mathbb{E}[|s_n|^4]/\mathbb{E}[|s_n|^2]$

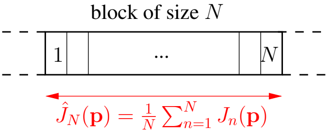
Kurtosis Minimization (KM) [ShalviWeinstein1990]

$$J = |\kappa_z|$$

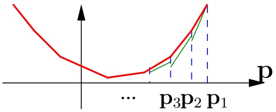
Implementation issue

How finding the minimum of $J(\mathbf{p}) = \mathbb{E}[J_n(\mathbf{p})]$?

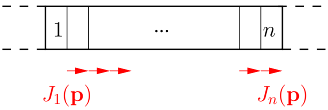
Blockwise processing



We replace $J(\mathbf{p})$ with $\hat{J}_N(\mathbf{p})$



Adaptive processing



We replace $J(\mathbf{p})$ with $J_n(\mathbf{p})$ at time/iteration n

- LMS
- Newton

Gradient algo.

$$\mathbf{p}_{i+1} = \mathbf{p}_i - \mu \frac{\partial \hat{J}_N(\mathbf{p})}{\partial \overline{\mathbf{p}}}\Big|_{\mathbf{p}_i}$$

(Stochastic) Gradient algo.

$$\mathbf{p}_{n+1} = \mathbf{p}_n - \mu \frac{\partial J_n(\mathbf{p})}{\partial \overline{\mathbf{p}}}\Big|_{\mathbf{p}_n}$$

Application to CMA

Adaptive implementation

$$\begin{aligned}\mathbf{p}_{n+1} &= \mathbf{p}_n - \mu\overline{\mathbf{y}_{L_p}(n)}z(n)(|z(n)|^2 - \text{Const}) \\ &= \mathbf{p}_n - \mu\overline{\mathbf{y}_{L_p}(n)}(z(n) - F_{\text{cma}}(z(n)))\end{aligned}$$

with

- $\mathbf{y}_{L_p}(n) = [y(n), \cdots, y(n - L_p)]^{\mathrm{T}}$
- $F_{\text{cma}}(z(n)) = z(n)(1 + C - |z(n)|^2)$
- if KM, $F_{\text{km}}(z(n)) = z(n)(1 + \text{sgn}(\kappa_s)|z(n)|^2)$

Special case: training sequence (known $s(n)$)

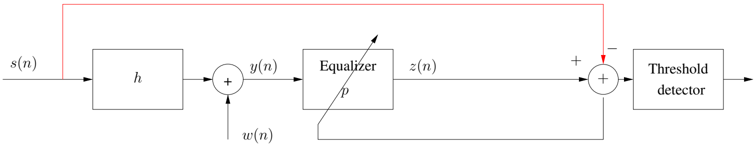
$$J = \mathbb{E}[|z(n) - s(n)|^2]$$

Adaptive implementation

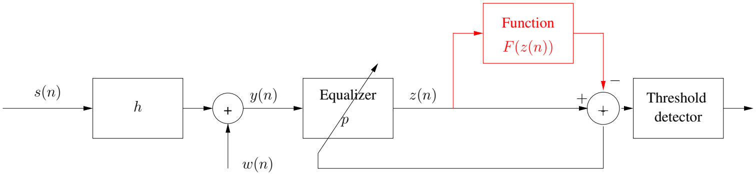
$$\mathbf{p}_{n+1} = \mathbf{p}_n - \mu\overline{\mathbf{y}_{L_p}(n)}(z(n) - s(n))$$

- $s(n)$ may be replaced by $\hat{s}(n)$ after initial convergence (DD)
- $s(n)$ is replaced by $F(z(n))$ which plays the role of “training”

Take-home message



Adaptive trained equalizer scheme



Adaptive blind equalizer scheme

Part 4: Second-Order Statistics based Algorithms

Principle

- Usually the algorithms rely on blind identification principle, i.e., retrieving the filter $\mathbf{h} = [\mathbf{h}(0)^{\mathrm{T}}, \cdots, \mathbf{h}(L)^{\mathrm{T}}]^{\mathrm{T}}$
- Talk done with the stationary SIMO model

$$\underbrace{\begin{bmatrix} \mathbf{y}(n) \\ \vdots \\ \mathbf{y}(n-N) \end{bmatrix}}_{\mathbf{Y}_N(n)} = \underbrace{\begin{bmatrix} \mathbf{h}(0) & \cdots & \mathbf{h}(L) & \cdots & 0 \\ \vdots & \ddots & & \ddots & \vdots \\ 0 & \cdots & \mathbf{h}(0) & \cdots & \mathbf{h}(L) \end{bmatrix}}_{\mathcal{T}(\mathbf{h})} \underbrace{\begin{bmatrix} s(n) \\ \vdots \\ s(n-N-L) \end{bmatrix}}_{\mathbf{S}_{N+L}(n)}$$

with $\mathcal{T}(\mathbf{h})$ a $2(N+1) \times (N+L+1)$ Sylvester matrix

Result

If $\mathbf{h}(\mathfrak{z}) \neq 0,\ \forall \mathfrak{z}$ and $N > L$, then $\mathcal{T}(\mathbf{h})$ is full column rank and left-invertible

Covariance matrix algorithm

Question: what is the best second-order algorithm?

Let

- $\mathbf{R}(\mathbf{h}) = \mathbb{E}[\mathbf{Y}_N(n)\mathbf{Y}_N(n)^{\mathrm{H}}]$ and $\hat{\mathbf{R}}_{N_{obs}} = \frac{1}{N_{obs}} \sum_{n=0}^{N_{obs}-1} \mathbf{Y}_N(n)\mathbf{Y}_N(n)^{\mathrm{H}}$
- $\mathbf{r}(\mathbf{h}) = [\Re\{\mathrm{vec}(\mathbf{R}(\mathbf{h}))\}^{\mathrm{T}}, \Im\{\mathrm{vec}(\mathbf{R}(\mathbf{h}))\}^{\mathrm{T}}]^{\mathrm{T}}$
- $\hat{\mathbf{r}}_{N_{obs}} = [\Re\{\mathrm{vec}(\hat{\mathbf{R}}_{N_{obs}})\}^{\mathrm{T}}, \Im\{\mathrm{vec}(\hat{\mathbf{R}}_{N_{obs}})\}^{\mathrm{T}}]^{\mathrm{T}}$

Result

$$\sqrt{N_{obs}}(\hat{\mathbf{r}}_{N_{obs}} - \mathbf{r}(\mathbf{h})) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \Gamma_{\mathbf{h}}),$$

i.e.,

$$\hat{\mathbf{r}}_{N_{obs}} \approx \mathbf{r}(\mathbf{h}) + \mathbf{w}_{N_{obs}}$$

with $\mathbf{w}_{N_{obs}}$ zero-mean Gaussian noise with covariance matrix $\Gamma_{\mathbf{h}}/N_{obs}$

Covariance matching algorithm (cont'd)

Maximum-Likelihood based on $\hat{\mathbf{r}}_{N_{obs}}$ instead of data $\mathbf{Y} = \mathbf{Y}_{N_{obs}}(N_{obs})$

$$\frac{1}{N_{obs}} \log(p(\hat{\mathbf{r}}_{N_{obs}}|\mathbf{h})) \approx -(\hat{\mathbf{r}}_{N_{obs}} - \mathbf{r}(\mathbf{h}))^{\mathrm{T}} \Gamma_{\mathbf{h}}^{-1} (\hat{\mathbf{r}}_{N_{obs}} - \mathbf{r}(\mathbf{h})) - \frac{\log(\det(\Gamma_{\mathbf{h}}))}{2N_{obs}} + \text{constant}$$

Result

$$\hat{\mathbf{h}}_{\mathrm{cm}} = \arg\min_{\mathbf{h}} \left\| \Gamma_{\mathbf{h}}^{-\frac{1}{2}} (\hat{\mathbf{r}}_{N_{obs}} - \mathbf{r}(\mathbf{h})) \right\|^2$$

with $\|\mathbf{W}^{\frac{1}{2}}\mathbf{x}\|^2 = \mathbf{x}^{\mathrm{H}}\mathbf{W}\mathbf{x}$

Ping-pong procedure for update $\Gamma_{\mathbf{h}}$

Maximum Likelihood algorithm

Question: Maximum Likelihood based on **Y**

$$\mathbf{Y} = \mathcal{T}(\mathbf{h})\mathbf{S} + \mathbf{W}$$

with **W** white zero-mean Gaussian noise and *unknown* **S**

$\max_{\mathbf{h}} p(\mathbf{Y}\|\mathbf{h}) = \int p(\mathbf{Y}\|\mathbf{h},\mathbf{S})p(\mathbf{S})d\mathbf{S}$	$\max_{\mathbf{h}} p(\mathbf{Y}\|\mathbf{h}) = \int p(\mathbf{Y}\|\mathbf{h},\mathbf{S})e^{-\mathbf{S}^{\mathrm{H}}\Gamma_s^{-1}\mathbf{S}}d\mathbf{S}$	$\max_{\mathbf{h},\mathbf{S}} p(\mathbf{Y}\|\mathbf{h},\mathbf{S})$
almost always untractable	tractable but not optimal	tractable but not optimal
TRUE ML	GAUSSIAN ML	DETERMINISTIC ML

Deterministic Maximum Likelihood

$$(\hat{\mathbf{h}}, \hat{\mathbf{S}})_{\mathrm{ML}} = \arg\min_{\mathbf{h},\mathbf{S}} \|\mathbf{Y} - \mathcal{T}(\mathbf{h})\mathbf{S}\|^2$$

Maximum Likelihood algorithm (cont'd)

- Minimization on **S** (without constraint):

$$\hat{\mathbf{S}}_{\mathrm{ML}} = (\mathcal{T}(\mathbf{h})^{\mathrm{H}}\mathcal{T}(\mathbf{h}))^{-1}\mathcal{T}(\mathbf{h})^{\mathrm{H}}\mathbf{Y}$$

- Then minimization on **h**:

$$\hat{\mathbf{h}}_{\mathrm{ML}} = \arg\min_{\mathbf{h}} \| \underbrace{(\mathbf{Id} - \mathcal{T}(\mathbf{h})(\mathcal{T}(\mathbf{h})^{\mathrm{H}}\mathcal{T}(\mathbf{h}))^{-1}\mathcal{T}(\mathbf{h})^{\mathrm{H}})}_{P_{\mathbf{h}}^{\perp}} \mathbf{Y}\|^2$$

with $P_{\mathbf{h}}^{\perp}$ the projection on $\mathrm{sp}(\mathcal{T}(\mathbf{h}))^{\perp}$

$$\hat{\mathbf{h}}_{\mathrm{ml}} = \arg\max_{\mathbf{h}} \mathbf{h}^{\mathrm{H}}\underline{\mathbf{Y}}^{\mathrm{H}}(\mathcal{T}(\mathbf{h})^{\mathrm{H}}\mathcal{T}(\mathbf{h}))^{-1}\underline{\mathbf{Y}}\mathbf{h}$$

- Quadratic cost function / **Y** $\Rightarrow$ Second ordre is fine
- Non-quadratic cost function / **h** $\Rightarrow$ Ping-pong procedure

Subspace algorithm: principle

Signal model:

$$\mathbf{y}(n) = \mathbf{A}(\theta)\mathbf{s}(n)$$

Main required property:

$$\mathrm{sp}(\mathbf{A}(\theta)) = \mathrm{sp}(\mathbf{A}(\theta')) \Longleftrightarrow \theta = \theta'$$

Algorithm main step:

$$\hat{\theta} = \arg\min_{\theta} \mathrm{distance}(\mathrm{vect}(\mathbf{y}(n)),\ \mathrm{sp}(\mathbf{A}(\theta)))$$

Example 1 : source localization (MUSIC)

$$\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \cdots, \mathbf{a}(\theta_p)]$$

with

- $\mathbf{a}(\theta) = [1, e^{2i\pi\theta}, \cdots, e^{2i\pi(M-1)\theta}]^{\mathrm{T}}$ (steering vector)
- $M > p$

Subspace algorithm: application to blind equalization

$$\mathbf{Y}_N(n) = \mathcal{T}(\mathbf{h})\mathbf{S}_{N+L}(n),$$

i.e.,

$$\mathbf{A} \longleftrightarrow \mathcal{T}(\mathbf{h}) \quad \text{and} \quad \theta \longleftrightarrow \mathbf{h}$$

Result

Let $\mathcal{T}(\mathbf{h}')$ be a Sylvester matrix associated with $\mathbf{h}'$
If $N \geq L$ and $\mathbf{h}(\mathfrak{z}) \neq 0\ \forall \mathfrak{z} \in \mathbb{C}$, then

$$\text{sp}(\mathcal{T}(\mathbf{h}')) = \text{sp}(\mathcal{T}(\mathbf{h})) \Longleftrightarrow \mathbf{h}' = \alpha\mathbf{h}$$

up to a constant α

Proof: using rational space or $\mathbb{C}[X]$-module

Subspace algorithm: practical implementation

White source $\Rightarrow \mathbf{R} = \mathbb{E}[\mathbf{Y}\mathbf{Y}^{\mathrm{H}}] = \mathcal{T}(\mathbf{h})\mathcal{T}(\mathbf{h})^{\mathrm{H}} \Rightarrow \mathrm{sp}(\mathbf{R}) = \mathrm{sp}(\mathcal{T}(\mathbf{h}))$

- Let Π be the projector on Ker($\mathbf{R}$) $\Rightarrow \Pi\mathbf{x} = 0 \quad \text{iff} \quad \mathbf{x} \in \mathrm{sp}(\mathbf{R}_{\mathbf{Y}})$
- Then $\mathbf{h}$ is the unique vector such that $\Pi\mathcal{T}(\mathbf{h}) = 0$
- In practice, $\mathbf{R}$ (resp. Π) is estimated by $\hat{\mathbf{R}}$ (resp. $\hat{\Pi}$).

$$\hat{\mathbf{h}}_{\mathrm{ss}} = \arg\min_{\|\mathbf{h}\|=1} \|\hat{\Pi}\mathcal{T}(\mathbf{h})\|^2 = \arg\min_{\|\mathbf{h}\|=1} \mathbf{h}^{\mathrm{H}}\mathbf{Q}\mathbf{h}$$

Linear Prediction algorithm

If $h_1(\mathfrak{z})$ and $h_2(\mathfrak{z})$ have no common root, Bezout's theorem holds:
$\exists[g_1(\mathfrak{z}), g_2(\mathfrak{z})]$ polynomials such that $g_1(\mathfrak{z})h_1(\mathfrak{z}) + g_2(\mathfrak{z})h_2(\mathfrak{z}) = 1$

Result

- Finite-degree MA = Finite-degree AR
- $\mathbf{y}(n)$ AR process of order L with innovation $\mathbf{i}(n) = \mathbf{h}(0)s(n)$, i.e.,

$$\mathbf{y}(n) + \sum_{\ell=1}^{L} \mathbf{A}(\ell)\mathbf{y}(n-\ell) = \mathbf{i}(n)$$

Algorithm implementation:

- Solve Yule-Walker equations (to obtain $\mathbf{A}(\ell)$ then $\mathbf{h}(\ell)$)

$$\mathbb{E}[\mathbf{i}(n)[\mathbf{y}(n-1)^{\mathrm{H}}, \cdots, \mathbf{y}(n-L)^{\mathrm{H}}]] = 0$$

- Estimate $\mathbf{h}(0)$ with the covariance matrix of the innovation

Part 5: Other types of algorithms

Semi-blind approach

Combining both criteria

- DA (with training sequence)
- blind/NDA (without training sequence)

as follows

$$J(\mathbf{h}) = \alpha J_{\mathrm{NDA}}(\mathbf{h}) + (1-\alpha) J_{\mathrm{DA}}(\mathbf{h})$$

Criteria selection (as an example):

- $J_{\mathrm{DA}}(\mathbf{h})$: ML
- $J_{\mathrm{NDA}}(\mathbf{h})$: Subspace algorithm

Result

Improve the estimation performance, or decrease the training duration

Decision directed approach

DA approach followed by

- NDA well initialized
- DD
  - with hard decisions
  - with soft decisions (turbo-estimation)

An other way: clustering based approach (or a step towards Machine Learning)

$$y(n) = \underbrace{\mathbf{h}^{\mathrm{T}}\mathbf{s}(n)}_{\mathbf{c}(n)} + \mathbf{w}(n)$$

with $\mathbf{s}(n) = [s(n), \ldots, s(n-L)]^{\mathrm{T}}$ and $\mathbf{h} = [h(0), \ldots, s(L)]^{\mathrm{T}}$

- $y(n)$ is a point in $\mathbb{C}$, and belongs to the cluster labelled by one $\mathbf{c}$
- K clusters to characterize (where $K = \text{card}(\mathbf{c})$ is known)
- Apply *unsupervised clustering* algorithm: K-means
- Now, given $\mathbf{c}$, how retrieving $\mathbf{s}(n)$ (with unknown $\mathbf{h}$)

Hidden Markov Model (HMM) approach

- $\mathbf{s}(n)$ is a Markov Chain:
  $\Pr(\mathbf{s}(n)|\mathbf{s}(n-1), \ldots) = \Pr(\mathbf{s}(n)|\mathbf{s}(n-1))$
- $\mathbf{c}(n)$ observation coming from an unknown Markov Chain state
- Forward-Backward algorithm to retrieve $\mathbf{h}$

An other way: clustering based approach (or a step towards Machine Learning) (cont'd)

$$\mathbf{y}(n) = \mathbf{fct}(\mathbf{s}(n)) + \mathbf{w}(n) \Rightarrow \hat{\mathbf{s}}(n) = \text{threshold}\,(\Theta(\mathbf{y}(n)))$$

with

- threshold : activation function
- $\Theta(\bullet)$: $\text{DNN}_{\text{weights}}(\bullet)$

Questions:

- One DNN per channel?
- If yes, training step (so it is not a blind approach)
- Gain in performance or less complex?
- Some papers on Optical-Fiber communications (trained for one fiber configuration)
- One DNN available for a large set of **fct**?

Part 6: Numerical illustrations

Second-order vs high-order algorithms

- Random multipath channel
- SIMO with oversampling of factor 2
- Observation window $1000T_s$

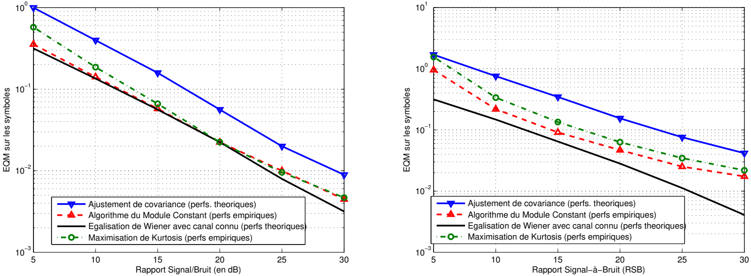


Figure: MSE vs SNR for 4QAM (left) and 16QAM (right) (courtesy of L. Mazet)

High-order algorithm (CMA)

$$\mathbf{y}(n) = \begin{bmatrix} 1 & \beta_1 \\ \beta_2 & 1 \end{bmatrix} .\mathbf{s}(n) + \mathbf{w}(n)$$

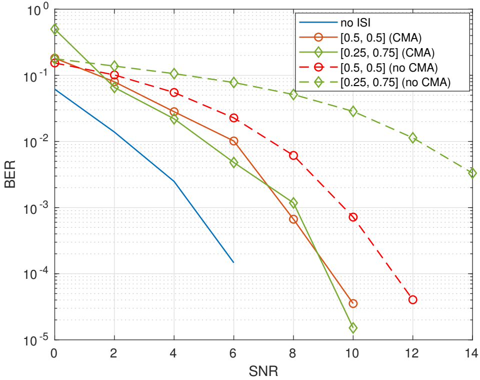


Figure: BER vs SNR with 4QAM (warmup step of 1000 samples)

Time-varying channels

- Stationary SISO model
- 4QAM
- 6-tap equalizer filter p

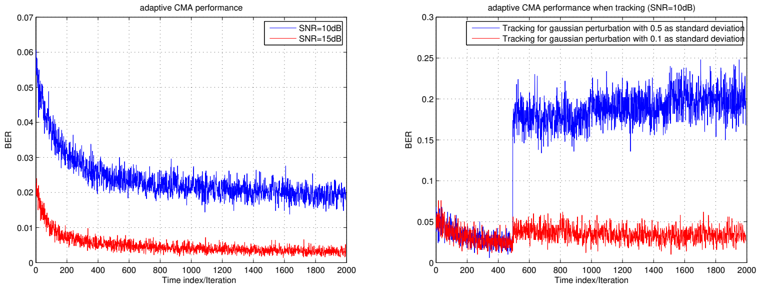
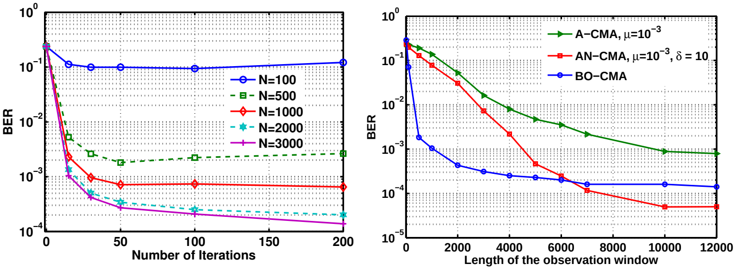


Figure: BER vs iteration: $\mathbf{h} = [0.3, 0.86, 0.39]^{\mathrm{T}}$ (left), $\mathbf{h} \leftarrow \mathbf{h} + \mathrm{std} \times \mathcal{N}(0, 1)$ at time index 500, 1000 and 1500 (right)

Use-case: optical-fiber (simulations)

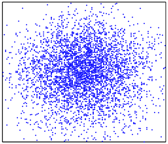
- PolMux 16QAM, 112Gbits/s, range 1000km
- CD=1000ps/nm
- DGD=50ps
- OSNR=20dB



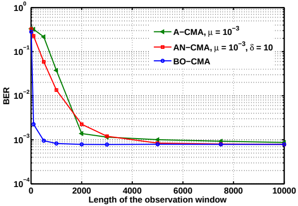
- Blockwise algorithm converges with $N = 1000$ and few iterations
- Adaptive algorithms need more samples to converge
- BER target (@10^{-3}) satisfied

Use-case: optical-fiber (experimentation)

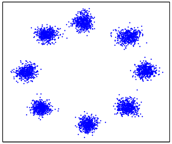
- PolMux 8PSK, 60Gbits/s, range 800km
- SSMF fiber
- OSNR=23.7dB



$\Rightarrow$



$\Rightarrow$



It works!

Conclusion

- Blind equalization works in pratice
- HOS:
  - No in-depth theoretical analysis
  - Drawback: large observation window (not civilian application yet, except optical fiber)
- SOS:
  - In-depth theoretical analysis (when N large enough)
  - Easy to use, espcially when SIMO coming from spatial diversity
- DNN?

References

- D. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communications systems", IEEE Trans. on Communications, Nov. 1980.
- O. Shalvi and E. Weinstein, “New criteria for blind deconvolution of non-minimum phase systems", IEEE Trans. on Information Theory, Mar. 1990.
- A. Benveniste, M. Métivier and P. Priouret, "Adaptive algorithms and stochastic approximations", Springer, 1990.
- L. Tong, G. Xu and T. Kailath, "A new approach to blind identification and equalization of multipath channels", Asilomar, 1991.
- D. Slock, "Blind fractionally-spaced equalization, perfect-reconstruction filter banks and multichannel linear predictor", ICASSP, 1994.
- E. Moulines, P. Duhamel, J.F. Cardoso and S. Mayrargue, "Subspace method for blind equalization of multichannel FIR filters", IEEE Trans. on Signal Processing, Feb. 1995.
- D. Boppana and S.S. Rao, “K-harmonic means clustering based blind equalization in hostile environments”, GLOBECOM, 2003.
- Z. Ding and G. Li, "Blind equalization and identification", Dekker, 2001.
- P. Loubaton, "Signal et télécoms", Hermès, 2004.
- P. Comon and C. Jutten, "Handbook of Blind Source Separation, Independent Component Analysis and Applications", Academic Press, 2010.