De la modélisation de problèmes directs en traitement d'image à la résolution des problèmes inverses

From the modelization of direct problems in image processing to the resolution of inverse problems

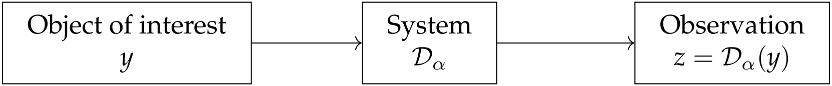
Caroline Chaux

Habilitation à Diriger des Recherches

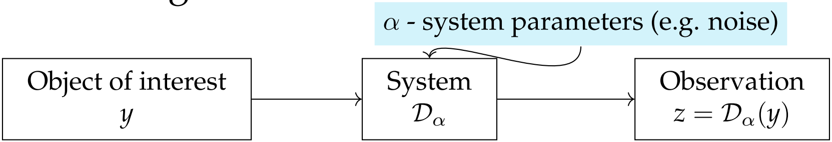
9 Jan. 2019

Contents

From modeling to resolution



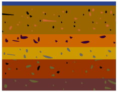
From modeling to resolution

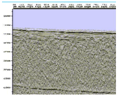


From modeling to resolution

α - system parameters (e.g. noise)

Object of interest y → System $\mathcal{D}_\alpha$ → Observation $z = \mathcal{D}_\alpha(y)$

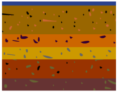


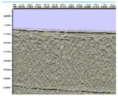
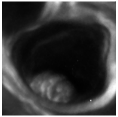


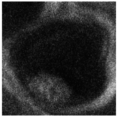
From modeling to resolution

α - system parameters (e.g. noise)

Object of interest y → System $\mathcal{D}_\alpha$ → Observation $z = \mathcal{D}_\alpha(y)$



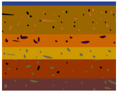



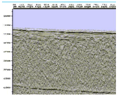


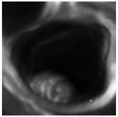
From modeling to resolution

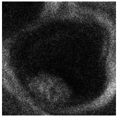
α - system parameters (e.g. noise)

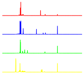
Object of interest y → System $\mathcal{D}_\alpha$ → Observation $z = \mathcal{D}_\alpha(y)$

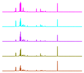








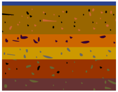


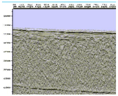
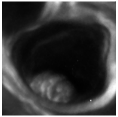


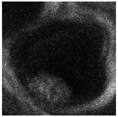
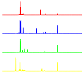
From modeling to resolution

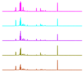
α - system parameters (e.g. noise)

Object of interest y → System $\mathcal{D}_\alpha$ → Observation $z = \mathcal{D}_\alpha(y)$



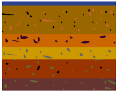


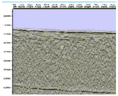
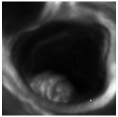
How? → Solving inverse problems

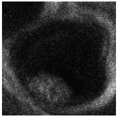
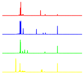
From modeling to resolution

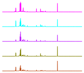
α - system parameters (e.g. noise)

Object of interest y → System $\mathcal{D}_\alpha$ → Observation $z = \mathcal{D}_\alpha(y)$



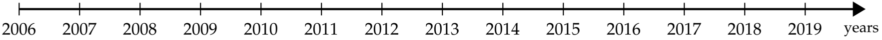





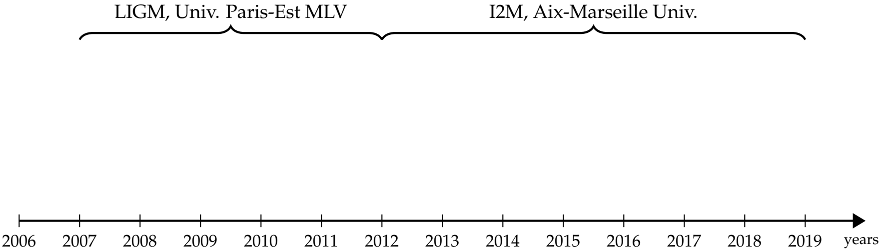
method parameters (e.g. regularization)

How? → Solving inverse problems

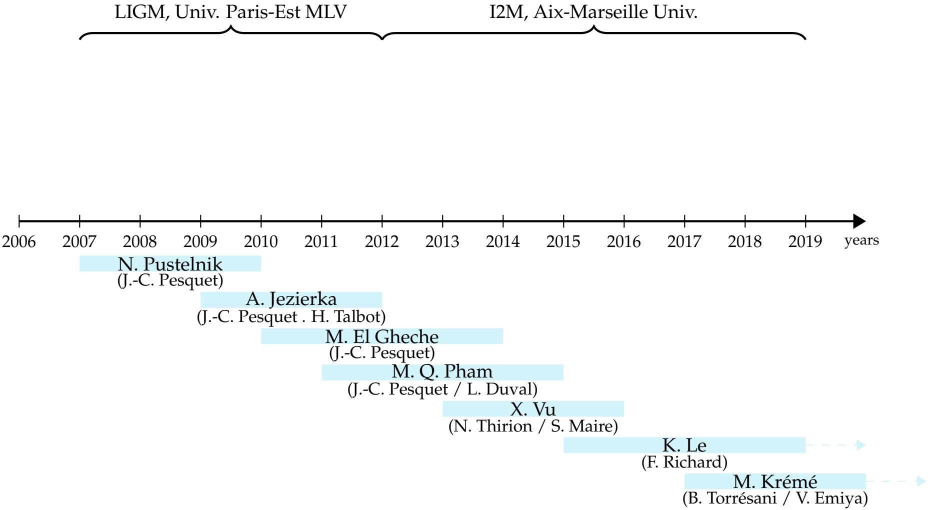
Curriculum


2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
years

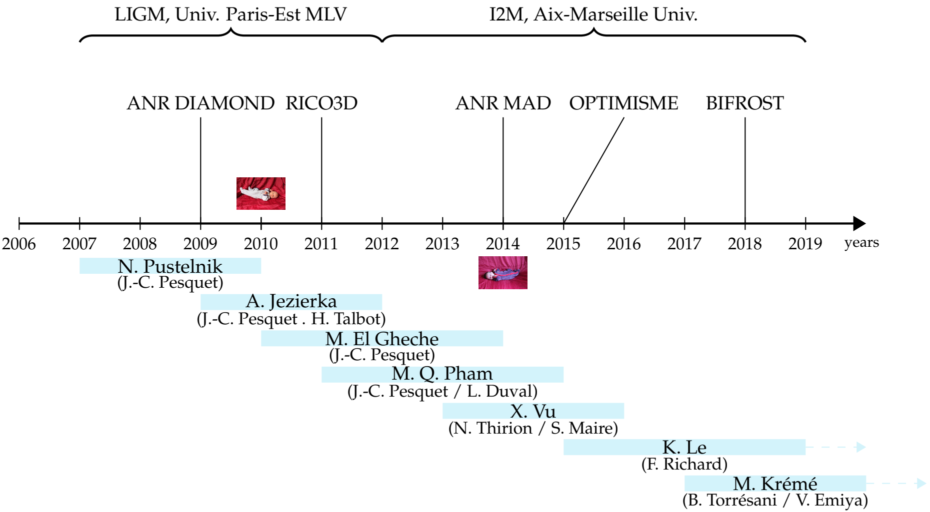
Curriculum


LIGM, Univ. Paris-Est MLV
I2M, Aix-Marseille Univ.
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
years

Curriculum


LIGM, Univ. Paris-Est MLV
I2M, Aix-Marseille Univ.
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
years
N. Pustelnik
(J.-C. Pesquet)
A. Jezierka
(J.-C. Pesquet . H. Talbot)
M. El Gheche
(J.-C. Pesquet)
M. Q. Pham
(J.-C. Pesquet / L. Duval)
X. Vu
(N. Thirion / S. Maire)
K. Le
(F. Richard)
M. Krémé
(B. Torrésani / V. Emiya)

Curriculum


LIGM, Univ. Paris-Est MLV
I2M, Aix-Marseille Univ.
ANR DIAMOND
RICO3D
ANR MAD
OPTIMISME
BIFROST
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
years
N. Pustelnik
(J.-C. Pesquet)
A. Jezierka
(J.-C. Pesquet . H. Talbot)
M. El Gheche
(J.-C. Pesquet)
M. Q. Pham
(J.-C. Pesquet / L. Duval)
X. Vu
(N. Thirion / S. Maire)
K. Le
(F. Richard)
M. Krémé
(B. Torrésani / V. Emiya)

Inverse problem formulation

What?

Recovering the original (unknown data) from distorded observations.

How?

Formulating the inverse problem as a minimization problem

- Variational approach;
- Statistical approach (MAP).

And so

$$\underset{y}{\text{minimize}} \quad \underbrace{f_1(y)}_{\text{Fidelity}} \quad + \quad \underbrace{f_2(y)}_{\text{Regularization}}$$

Minimization problems

- Standard problem:

$$\underset{y \in \mathbb{R}^N}{\text{minimize}} \; \underbrace{f_1(y)}_{\text{Fidelity}} + \underbrace{f_2(y)}_{\text{Regularization}} .$$

- Taking into account several regularizations ($P - 1$ terms):

$$\underset{y \in \mathbb{R}^N}{\text{minimize}} \; f_1(y) + \sum_{p=2}^{P} f_p(y).$$

- Introducing linear operators $(F_p)_{p \in \{1,\dots,P\}}$:

$$\underset{y \in \mathbb{R}^N}{\text{minimize}} \; \sum_{p=1}^{P} f_p(F_p y).$$

- For large size problem or for other reasons, can be interesting to work on data blocks $y^{(p)}$ of size L_p ($y = (y^{(p)})_{p=1}^{P}$)

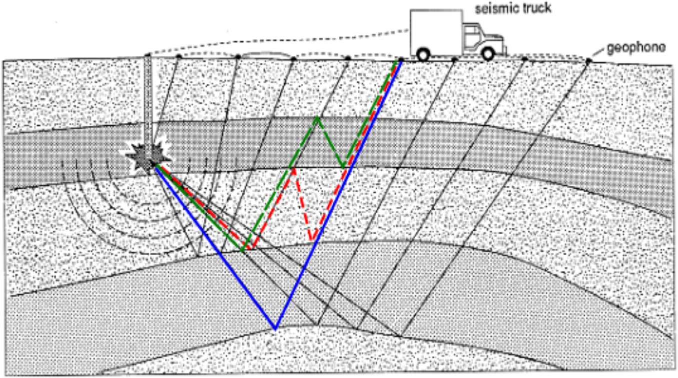
$$\underset{y \in \mathbb{R}^N}{\text{minimize}} \; \sum_{p=1}^{P} f_p(y^{(p)}).$$

Some proximal approaches

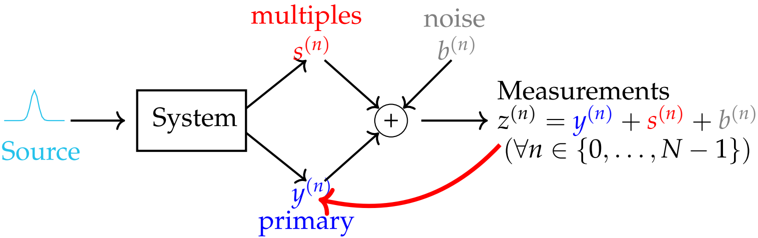
- *Parallel ProXimal Algorithm + (PPXA+)* [Pesquet, Pustelnik, 2012]
- Generalized Forward-Backward [Raguet et al., 2012]
- M+SFBF [Briceño-Arias, Combettes, 2011]
- M+LFBF [Combettes, Pesquet, 2011]
- FB based algorithms [Chambolle, Pock, 2011],[Vũ,2013],[Condat,2013]
- Proximal Alternating Linearized Minimization (PALM) [Bolte et al., 2014]
- An accelerated projection gradient based algorithm [Zhang et al., 2016]
- Block-Coordinate Variable Metric Forward-Backward (BC-VMFB) algorithm [Chouzenoux et al., 2016]

General principal of data acquisition

Land seismic data acquisition



(Multiple + noise) removal strategies



Which strategy for restoring the primary signal $y^{(n)}$ corrupted by the unknown multiples $s^{(n)}$, plus noise $b^{(n)}$?

- Methodology for primary/multiple adaptive separation based on approximate templates
- Variational approach
- Proximal methods to solve the resulting optimization problem

Multi-model

- J models $r_j^{(n)}$ are known (available)
- Imperfect in time, amplitude and frequency
- **Assumption**: models linked to $\bar{s}^{(n)}$ throughout time varying filters (FIR)

$$\bar{s}^{(n)} = \sum_{j=0}^{J-1} \sum_{p=0}^{P_j-1} \bar{h}_j^{(n)}(p) r_j^{(n-p)}$$

where

- $\bar{h}_j^{(n)}$: unknown impulse reponse of the filter corresponding to model j and time n (P_j tap coefficients)
- New definition: $P = \sum_{j=0}^{J-1} P_j$.

Matrix form:

$$\underbrace{z}_{\text{observed signal}} = \mathbf{R}\underbrace{\bar{\mathbf{h}}}_{\text{filter}} + \underbrace{\bar{y}}_{\text{primary}} + \underbrace{b}_{\text{noise}}$$

MAP estimation of $(y, \mathbf{h})$

Assumptions:

- $\bar{y}$: realization of a random vector Y, of probability density:
  $(\forall y \in \mathbb{R}^N) \qquad f_Y(y) \propto \exp(-\varphi(Fy))$
  $F \in \mathbb{R}^{K\times N}$: linear operator, φ is chosen separable
- $\bar{\mathbf{h}}$: realization of a random vector H, of probability density:
  $(\forall \mathbf{h} \in \mathbb{R}^{NP})\ f_H(\mathbf{h}) \propto \exp(-\rho(\mathbf{h}))$
  H is independent of Y.
- b: realization of a random vector B, of probability density:
  $(\forall b \in \mathbb{R}^N) \qquad f_B(b) \propto \exp(-\psi(b))$
  B is assumed to be independent from Y and H

MAP estimation of $(y, \mathbf{h})$

Assumptions:

- $\bar{y}$: realization of a random vector Y, of probability density:
  $(\forall y \in \mathbb{R}^N) \qquad f_Y(y) \propto \exp(-\varphi(Fy))$
  $F \in \mathbb{R}^{K \times N}$: linear operator, φ is chosen separable
- $\bar{\mathbf{h}}$: realization of a random vector H, of probability density:
  $(\forall \mathbf{h} \in \mathbb{R}^{NP})\ f_H(\mathbf{h}) \propto \exp(-\rho(\mathbf{h}))$
  H is independent of Y.
- b: realization of a random vector B, of probability density:
  $(\forall b \in \mathbb{R}^N) \qquad f_B(b) \propto \exp(-\psi(b))$
  B is assumed to be independent from Y and H

MAP estimation of $(y, \mathbf{h})$

$$\underset{y \in \mathbb{R}^N, \mathbf{h} \in \mathbb{R}^{NP}}{\text{minimize}} \quad \underbrace{\psi(z - \mathbf{R}\mathbf{h} - y)}_{\text{fidelity: linked to noise}} + \underbrace{\varphi(Fy)}_{\textit{a priori} \text{ on the signal}} + \underbrace{\rho(\mathbf{h})}_{\textit{a priori} \text{ on the filters}}$$

♣

Problem to be solved

MAP estimation of $(y, \mathbf{h})$

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}} \quad \underbrace{\psi(z - \mathbf{R}\mathbf{h} - y)}_{\text{fidelity: linked to noise}} + \underbrace{\varphi(Fy)}_{\textit{a priori} \text{ on the signal}} + \underbrace{\rho(\mathbf{h})}_{\textit{a priori} \text{ on the filters}}$$

- **Difficulty: Choosing the good regularization parameters**
- **Proposed: Use a constrained minimization problem**

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}} \quad \psi(z - \mathbf{R}\mathbf{h} - y) + \iota_D(Fy) + \iota_C(\mathbf{h})$$

About convex set D

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi(z-\mathbf{R}\mathbf{h}-y)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$\iota_{\mathrm{D}}(x)=\begin{cases}0 & \text{if } x\in D\\ +\infty & \text{otherwise.}\end{cases}$$

- $F\in\mathbb{R}^{K\times N}$: analysis frame operator
- $\{\mathbb{K}_l \mid l\in\{1,\ldots,\mathcal{L}\}\}\subset\{1,\ldots,K\}$
- $D=D_1\times\cdots\times D_{\mathcal{L}}$ with
  $D_l=\{(x_k)_{k\in\mathbb{K}_l} \mid \sum_{k\in\mathbb{K}_l}\varphi_\ell(x_k)\le\beta_l\}$, where
  $\forall l\in\{1,\ldots,\mathcal{L}\}$, $\beta_l\in\,]0,+\infty[$, and $\varphi_l:\mathbb{R}\to[0,+\infty[$ is a lower-semicontinuous convex function.

About convex set C

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi(z-\mathbf{Rh}-y)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$C = C_1 \cap C_2 \cap C_3$$

- $C_1 = \left\{\mathbf{h}\in\mathbb{R}^{PN}\ :\ \rho(\mathbf{h}) = \sum_{j=0}^{J-1}\rho_j(h_j)\le\tau\right\}$
-

$$C_2 = \left\{h \mid \forall p,\ \forall n\in\left\{0,\dots,\left\lfloor\frac{N}{2}\right\rfloor-1\right\}\quad \left|h^{(2n+1)}(p)-h^{(2n)}(p)\right|\le\varepsilon_p\right\}$$

$$C_3 = \left\{h \mid \forall p,\ \forall n\in\left\{1,\dots,\left\lfloor\frac{N-1}{2}\right\rfloor\right\}\quad \left|h^{(2n)}(p)-h^{(2n-1)}(p)\right|\le\varepsilon_p\right\}$$

Proposed algorithm

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}} \ \Psi\left(\begin{bmatrix} y \\ \mathbf{h} \end{bmatrix}\right) + \iota_D(Fy) + \iota_C(\mathbf{h})$$

- $\Psi : \mathbb{R}^{N+NP} \to \mathbb{R} : \begin{bmatrix} y \\ \mathbf{h} \end{bmatrix} \mapsto \psi(z - \mathbf{R}\mathbf{h} - y)$ is convex and differentiable with μ-Lipschitzian gradient ($\mu \in]0,+\infty[$) i.e.

$$\left(\forall (u,v) \in \mathbb{R}^{2(N+NP)}\right) \quad \|\nabla\Psi(u) - \nabla\Psi(v)\| \leq \mu\|u-v\|,$$

- $(\forall i \in \mathbb{N})$, $\gamma^{[i]} \in [\epsilon, \frac{1-\epsilon}{\beta}]$ where

$$\beta = \mu + \sqrt{\|F\|^2 + 3} \text{ and } \epsilon \in]0, \frac{1}{\beta+1}[$$

$\rightsquigarrow$ **Use the M+LFBF algorithm [Combettes and Pesquet, 2012]**

Algorithm M+LFBF [Combettes and Pesquet, 2012]

Set $y^{[0]} \in \mathbb{R}^N$, $\mathbf{h}^{[0]} \in \mathbb{R}^{NP}$, $v^{[0]} \in \mathbb{R}^K$, $u^{[0]} \in \mathbb{R}^{NP}$

for $i = 0, 1, \ldots$ **do**

Gradient computation

$$\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* v^{[i]} \\ u^{[i]} \end{bmatrix} \right)$$

Projection computation

$$s_2^{[i]} = v^{[i]} + \gamma^{[i]} F y^{[i]}, \quad w_1^{[i]} = s_2^{[i]} - \gamma^{[i]} \Pi_D((\gamma^{[i]})^{-1} s_2^{[i]})$$
$$t_2^{[i]} = u^{[i]} + \gamma^{[i]} \mathbf{h}^{[i]}, \quad w_2^{[i]} = t_2^{[i]} - \gamma^{[i]} \Pi_C((\gamma^{[i]})^{-1} t_2^{[i]})$$

Averaging

$$q_1^{[i]} = w_1^{[i]} + \gamma^{[i]} F s_1^{[i]}, \quad v^{[i+1]} = v^{[i]} - s_2^{[i]} + q_1^{[i]}$$
$$q_2^{[i]} = w_2^{[i]} + \gamma^{[i]} t_1^{[i]}, \quad u^{[i+1]} = u^{[i]} - t_2^{[i]} + q_2^{[i]}$$

Update

$$\begin{bmatrix} y^{[i+1]} \\ \mathbf{h}^{[i+1]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* w_1^{[i]} \\ w_2^{[i]} \end{bmatrix} \right)$$

end for

Algorithm M+LFBF [Combettes and Pesquet, 2012]

Set $y^{[0]} \in \mathbb{R}^N$, $\mathbf{h}^{[0]} \in \mathbb{R}^{NP}$, $v^{[0]} \in \mathbb{R}^K$, $u^{[0]} \in \mathbb{R}^{NP}$

for $i = 0, 1, \ldots$ **do**

Gradient computation

$$\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla\Psi \left(\begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* v^{[i]} \\ u^{[i]} \end{bmatrix} \right)$$

Projection computation

$$s_2^{[i]} = v^{[i]} + \gamma^{[i]} F y^{[i]}, \quad w_1^{[i]} = s_2^{[i]} - \gamma^{[i]} \Pi_D((\gamma^{[i]})^{-1} s_2^{[i]})$$

$$t_2^{[i]} = u^{[i]} + \gamma^{[i]} \mathbf{h}^{[i]}, \quad w_2^{[i]} = t_2^{[i]} - \gamma^{[i]} \Pi_C((\gamma^{[i]})^{-1} t_2^{[i]})$$

Averaging

$$q_1^{[i]} = w_1^{[i]} + \gamma^{[i]} F s_1^{[i]}, \quad v^{[i+1]} = v^{[i]} - s_2^{[i]} + q_1^{[i]}$$

$$q_2^{[i]} = w_2^{[i]} + \gamma^{[i]} t_1^{[i]}, \quad u^{[i+1]} = u^{[i]} - t_2^{[i]} + q_2^{[i]}$$

Update

$$\begin{bmatrix} y^{[i+1]} \\ \mathbf{h}^{[i+1]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla\Psi \left(\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* w_1^{[i]} \\ w_2^{[i]} \end{bmatrix} \right)$$

end for

Algorithm M+LFBF [Combettes and Pesquet, 2012]

Set $y^{[0]} \in \mathbb{R}^N, \mathbf{h}^{[0]} \in \mathbb{R}^{NP}, v^{[0]} \in \mathbb{R}^K, u^{[0]} \in \mathbb{R}^{NP}$

for $i = 0, 1, \ldots$ **do**

Gradient computation

$$\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* v^{[i]} \\ u^{[i]} \end{bmatrix} \right)$$

Projection computation

$$s_2^{[i]} = v^{[i]} + \gamma^{[i]} F y^{[i]}, \quad w_1^{[i]} = s_2^{[i]} - \gamma^{[i]} \Pi_D((\gamma^{[i]})^{-1} s_2^{[i]})$$

$$t_2^{[i]} = u^{[i]} + \gamma^{[i]} \mathbf{h}^{[i]}, \quad w_2^{[i]} = t_2^{[i]} - \gamma^{[i]} \Pi_C((\gamma^{[i]})^{-1} t_2^{[i]})$$

Averaging

$$q_1^{[i]} = w_1^{[i]} + \gamma^{[i]} F s_1^{[i]}, \quad v^{[i+1]} = v^{[i]} - s_2^{[i]} + q_1^{[i]}$$

$$q_2^{[i]} = w_2^{[i]} + \gamma^{[i]} t_1^{[i]}, \quad u^{[i+1]} = u^{[i]} - t_2^{[i]} + q_2^{[i]}$$

Update

$$\begin{bmatrix} y^{[i+1]} \\ \mathbf{h}^{[i+1]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* w_1^{[i]} \\ w_2^{[i]} \end{bmatrix} \right)$$

end for

Algorithm M+LFBF [Combettes and Pesquet, 2012]

Set $y^{[0]} \in \mathbb{R}^N$, $\mathbf{h}^{[0]} \in \mathbb{R}^{NP}$, $v^{[0]} \in \mathbb{R}^K$, $u^{[0]} \in \mathbb{R}^{NP}$

for $i = 0, 1, \ldots$ **do**

Gradient computation

$$\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* v^{[i]} \\ u^{[i]} \end{bmatrix} \right)$$

Projection computation

$$s_2^{[i]} = v^{[i]} + \gamma^{[i]} F y^{[i]}, \quad w_1^{[i]} = s_2^{[i]} - \gamma^{[i]} \Pi_D((\gamma^{[i]})^{-1} s_2^{[i]})$$

$$t_2^{[i]} = u^{[i]} + \gamma^{[i]} \mathbf{h}^{[i]}, \quad w_2^{[i]} = t_2^{[i]} - \gamma^{[i]} \Pi_C((\gamma^{[i]})^{-1} t_2^{[i]})$$

Averaging

$$q_1^{[i]} = w_1^{[i]} + \gamma^{[i]} F s_1^{[i]}, \quad v^{[i+1]} = v^{[i]} - s_2^{[i]} + q_1^{[i]}$$

$$q_2^{[i]} = w_2^{[i]} + \gamma^{[i]} t_1^{[i]}, \quad u^{[i+1]} = u^{[i]} - t_2^{[i]} + q_2^{[i]}$$

Update

$$\begin{bmatrix} y^{[i+1]} \\ \mathbf{h}^{[i+1]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla \Psi \left(\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* w_1^{[i]} \\ w_2^{[i]} \end{bmatrix} \right)$$

end for

Algorithm M+LFBF [Combettes and Pesquet, 2012]

Set $y^{[0]} \in \mathbb{R}^N, \mathbf{h}^{[0]} \in \mathbb{R}^{NP}, v^{[0]} \in \mathbb{R}^K, u^{[0]} \in \mathbb{R}^{NP}$

for $i = 0, 1, \ldots$ **do**

Gradient computation

$$\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla\Psi\left(\begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* v^{[i]} \\ u^{[i]} \end{bmatrix} \right)$$

Projection computation

$$s_2^{[i]} = v^{[i]} + \gamma^{[i]} F y^{[i]}, \quad w_1^{[i]} = s_2^{[i]} - \gamma^{[i]} \Pi_D((\gamma^{[i]})^{-1} s_2^{[i]})$$
$$t_2^{[i]} = u^{[i]} + \gamma^{[i]} \mathbf{h}^{[i]}, \quad w_2^{[i]} = t_2^{[i]} - \gamma^{[i]} \Pi_C((\gamma^{[i]})^{-1} t_2^{[i]})$$

Averaging

$$q_1^{[i]} = w_1^{[i]} + \gamma^{[i]} F s_1^{[i]}, \quad v^{[i+1]} = v^{[i]} - s_2^{[i]} + q_1^{[i]}$$
$$q_2^{[i]} = w_2^{[i]} + \gamma^{[i]} t_1^{[i]}, \quad u^{[i+1]} = u^{[i]} - t_2^{[i]} + q_2^{[i]}$$

Update

$$\begin{bmatrix} y^{[i+1]} \\ \mathbf{h}^{[i+1]} \end{bmatrix} = \begin{bmatrix} y^{[i]} \\ \mathbf{h}^{[i]} \end{bmatrix} - \gamma^{[i]} \left(\nabla\Psi\left(\begin{bmatrix} s_1^{[i]} \\ t_1^{[i]} \end{bmatrix} \right) + \begin{bmatrix} F^* w_1^{[i]} \\ w_2^{[i]} \end{bmatrix} \right)$$

end for

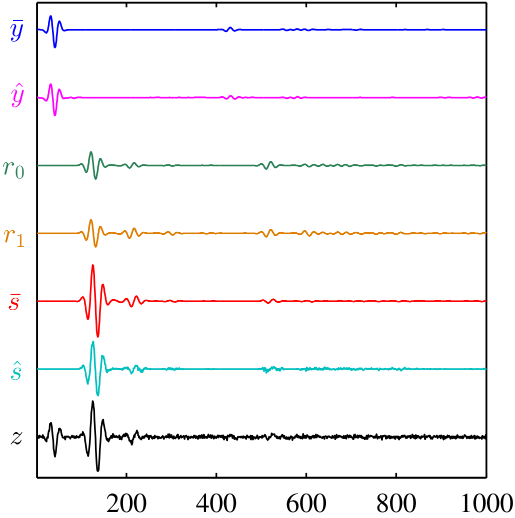
Context

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}} \; \Psi\left(\begin{bmatrix} y \\ \mathbf{h} \end{bmatrix}\right) + \iota_D(Fy) + \iota_C(\mathbf{h})$$

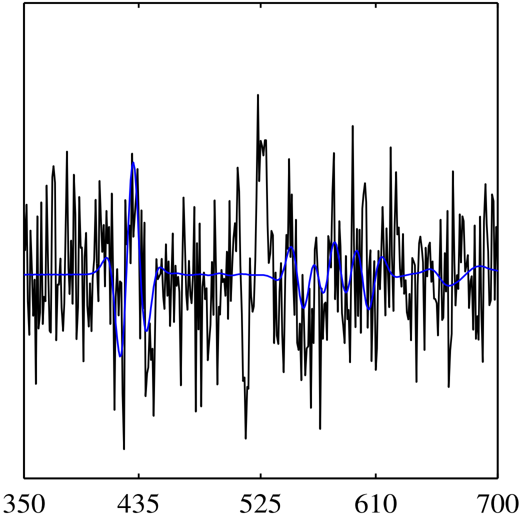
- $\Psi = \|\cdot\|^2 \;\Leftrightarrow\; B \sim \mathcal{N}(0, \sigma^2)$
- $\varphi_l = |\cdot|$ and F: Symlet wavelets of length 8 over 4 resolution levels.
- $\rho_j = \ell_{1,2}$ and constraints choice: $\varepsilon_{0,p} = 0.1$ and $\varepsilon_{1,p} = 0.07$ for every p.
- Number of templates : $J = 2$
- Filter length: $P_0 = 10, P_1 = 14$
- Iteration number: 10000
  (stopping criterion at iteration i if $\|y^{[i+1]} - y^{[i]}\| < 10^{-6}$)
- Signal length: $N = 1024$

Synthetic data (1D)



Simulated seismic signals:
noise level $\sigma = 0.08$

Synthetic data (1D)

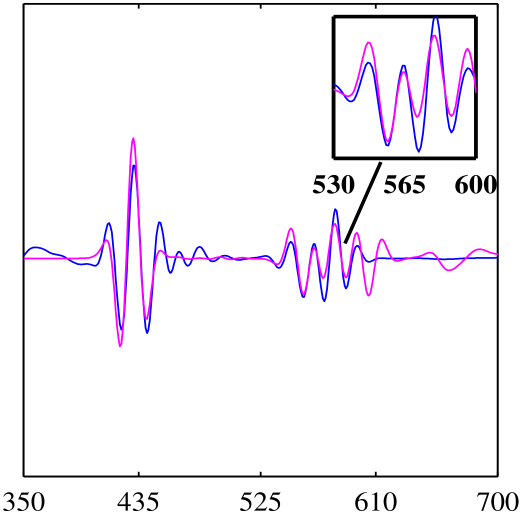


Simulated seismic signals:
noise level $\sigma = 0.08$

Close-up:

input data: z ($\sigma = 0.08$)
primary: $\bar{y}$

Synthetic data (1D)



Simulated seismic signals:
noise level $\sigma = 0.08$

Close-up:

input data: z ($\sigma = 0.08$)
primary: $\overline{y}$

Results
output separated primary: $\widehat{y}$
primary: $\overline{y}$

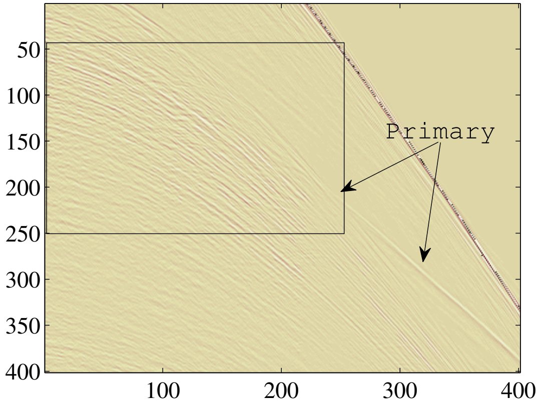
Extension to 2D signals

$$z^{(\mathbf{n})} = \overline{y}^{(\mathbf{n})} + \overline{s}^{(\mathbf{n})} + b^{(\mathbf{n})}$$

- $\mathbf{n} = (n_t, n_x)$, $n_t \in \mathbb{N}_t \triangleq \{0, \ldots, N_t - 1\}$: the time index; $n_x \in \mathbb{N}_x \triangleq \{0, \ldots, N_x - 1\}$: the sensor index
- $\mathbf{n} \in \mathcal{N} \triangleq \{(n_t, n_x) | \, n_t \in \mathbb{N}_t, \, n_x \in \mathbb{N}_x\}$.
- $$\overline{s}^{(\mathbf{n})} = \sum_{j=0}^{J-1} \sum_{p=0}^{P_j - 1} \overline{h}_j^{(\mathbf{n})}(p) r_j^{(n_t - p, n_x)}$$

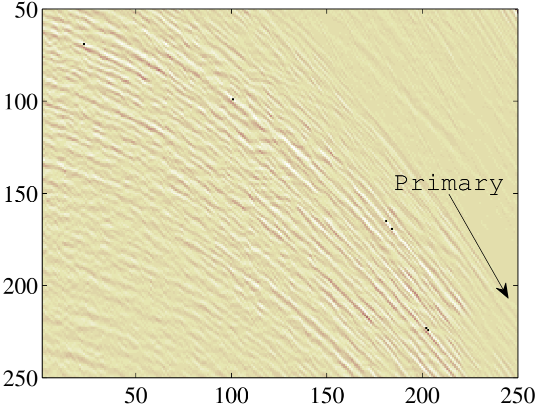
$\hookrightarrow$ smooth variations of the filters h along time and along sensors, F: 2D sparse representation.

Real data



Seismic data with
a partially appearing primary
- size 400×400

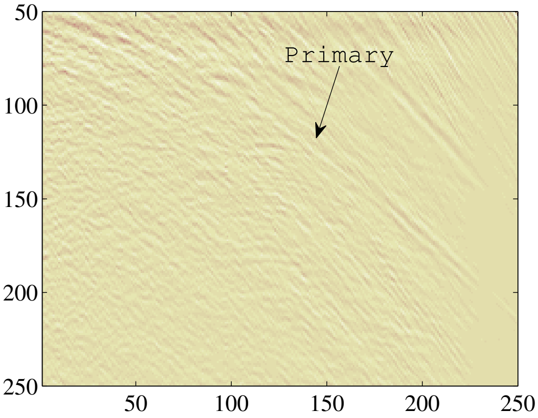
Real data



Seismic data with a partially appearing primary
- size 400×400

cropped of recorded seismic data: z
- size 256×256

Real data



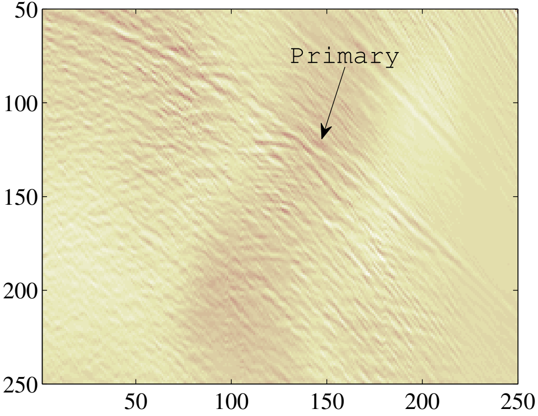
Seismic data with
a partially appearing primary
- size 400×400

cropped of recorded
seismic data: z
- size 256×256

Reconstructed image by

[Ventosa *et al.*, 2012]

Real data



Seismic data with
a partially appearing primary
- size 400×400

cropped of recorded
seismic data: z
- size 256×256

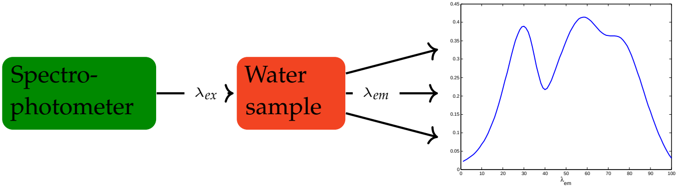
Reconstructed image by

[Ventosa *et al.*, 2012]

Our 2D method

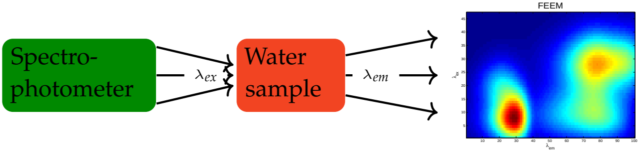
3D fluorescence spectroscopy

- **Problem**: identifying dissolved fluorescent substances in water solutions
- **Method**: fluorescence spectroscopy technique
- **Data acquisition:**



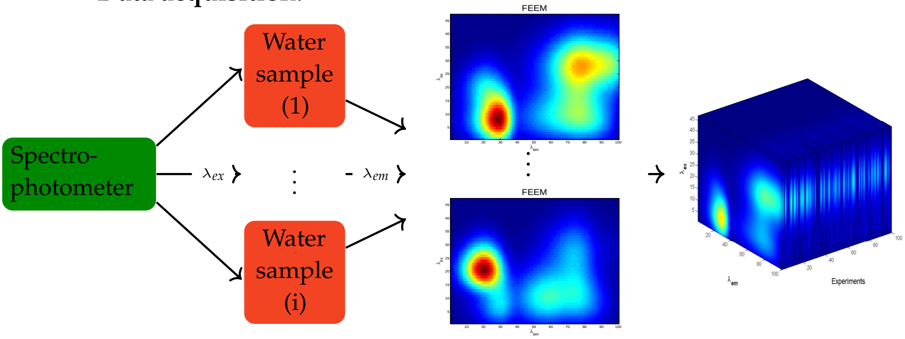
3D fluorescence spectroscopy

- **Problem**: identifying dissolved fluorescent substances in water solutions
- **Method**: fluorescence spectroscopy technique
- **Data acquisition:**



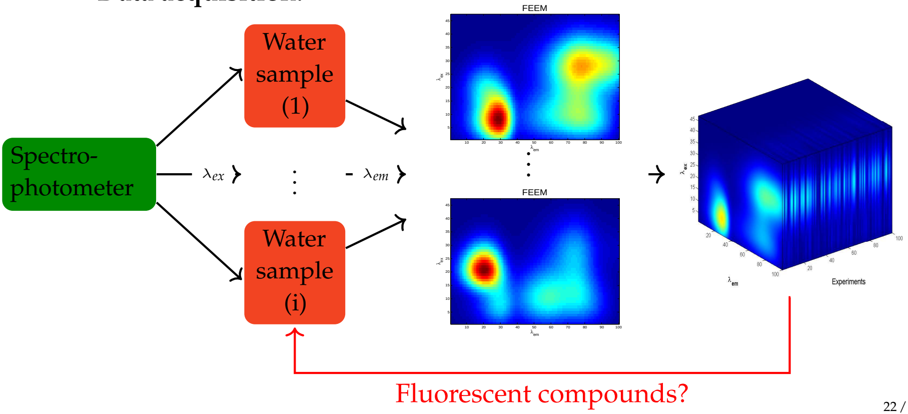
3D fluorescence spectroscopy

- **Problem**: identifying dissolved fluorescent substances in water solutions
- **Method**: fluorescence spectroscopy technique
- **Data acquisition:**



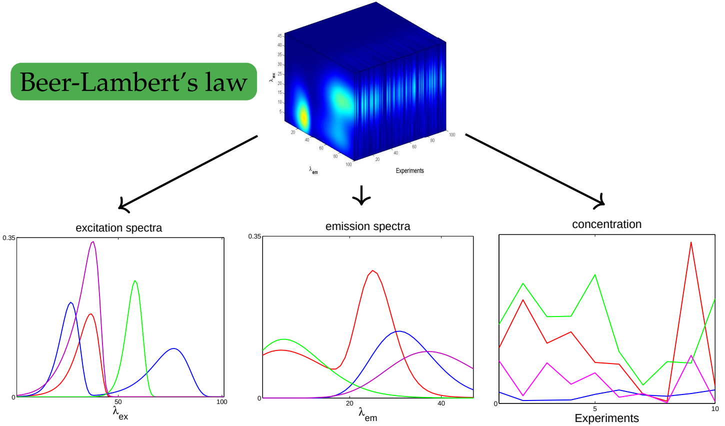
3D fluorescence spectroscopy

- **Problem**: identifying dissolved fluorescent substances in water solutions
- **Method**: fluorescence spectroscopy technique
- **Data acquisition:**



3D fluorescence spectroscopy and tensors

Beer-Lambert's law



$$\overline{\mathcal{T}} = \sum_{r=1}^{\overline{R}} \bar{\mathbf{a}}_r^{(1)} \circ \bar{\mathbf{a}}_r^{(2)} \circ \bar{\mathbf{a}}_r^{(3)}$$

$\circ$: outer product

(Canonical) Polyadic Decomposition (CPD)

Tensor form: [Harshman1927]

Tensor rank: $\bar{R}$

Loading vectors: $\bar{\mathbf{a}}_r^{(1)}, \bar{\mathbf{a}}_r^{(2)}, \ldots, \bar{\mathbf{a}}_r^{(N)}$

Loading matrices: $\bar{\mathbf{A}}^{(1)}, \bar{\mathbf{A}}^{(2)}, \ldots, \bar{\mathbf{A}}^{(N)}$

$$\overline{\mathcal{T}} = \sum_{r=1}^{\overline{R}} \underbrace{\bar{\mathbf{a}}_r^{(1)} \circ \bar{\mathbf{a}}_r^{(2)} \circ \ldots \circ \bar{\mathbf{a}}_r^{(N)}}_{\text{Rank-1 tensor}} = [\![\bar{\mathbf{A}}^{(1)}, \bar{\mathbf{A}}^{(2)}, \ldots, \bar{\mathbf{A}}^{(N)}]\!]$$

$\forall n \in \{1, 2, \ldots, N\}$, $\bar{\mathbf{a}}_r^{(n)} \in \mathbb{R}^{I_n}$ and $\bar{\mathbf{A}}^{(n)} \in \mathbb{R}^{I_n \times \overline{R}}$

Canonical Polyadic Decomposition (CPD) (2)

Scalar form:

$$\bar{t}_{i_1 \dots i_N} = \sum_{r=1}^{\bar{R}} \bar{a}^{(1)}_{i_1 r} \bar{a}^{(2)}_{i_2 r} \dots \bar{a}^{(N)}_{i_N r}$$

Matrix form: [Cichocki2009]

$$\overline{\mathbf{T}}^{(n)}_{I_n, I_{-n}} = \bar{\mathbf{A}}^{(n)} (\overline{\mathbf{Z}}^{(-n)})^\top, \quad n \in \{1, \dots, N\},$$

$\overline{\mathbf{T}}^{(n)}_{I_n, I_{-n}} \in \mathbb{R}_+^{I_n \times I_{-n}}$: the matrix obtained by unfolding $\overline{\mathcal{T}}$ in the n-th mode, $I_{-n} = I_1 \dots I_N / I_n$; for all $n \in \{1, \dots, N\}$,

$$\overline{\mathbf{Z}}^{(-n)} = \bar{\mathbf{A}}^{(N)} \odot \dots \bar{\mathbf{A}}^{(n+1)} \odot \bar{\mathbf{A}}^{(n-1)} \odot \bar{\mathbf{A}}^{(1)} \in \mathbb{R}_+^{I_{-n} \times \bar{R}}$$

$\odot$: Khatri-Rao product.

Objective: tensor decomposition

- **Input**: Observed tensor $\mathcal{T}$
- **Output**: Estimated loading factors $\widehat{\mathbf{a}}_r^{(n)}$ for all $n \in \{1, \ldots, N\}$

Constraint:

- Loading factors $\bar{\mathbf{a}}_r^{(n)}$ entrywise nonnegative

Difficulties:

- Large dimension tensors
- Rank $\overline{R}$ unknown $\rightarrow$ needs to be estimated (overestimation problems)

Proximal algorithm for CP decomposition

$$\overline{\mathcal{T}} = \sum_{r=1}^{\overline{R}} \bar{\mathbf{a}}_r^{(1)} \circ \ldots \circ \bar{\mathbf{a}}_r^{(N)} = [\![\bar{\mathbf{A}}^{(1)}, \ldots, \bar{\mathbf{A}}^{(N)}]\!].$$

Tensor structure: naturally leads to consider N blocks corresponding to the loading matrices $\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}$

Proposed optimization problem

$$\underset{\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R},\, n \in \{1,\ldots,N\}}{\text{minimize}} \quad \mathcal{F}(\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}) + \mathcal{R}_1(\mathbf{A}^{(1)}) + \ldots + \mathcal{R}_N(\mathbf{A}^{(N)})$$

Fidelity term

- $\mathcal{F}(\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)})$: quadratic data fidelity term

$$\mathcal{F}(\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}) = \frac{1}{2}\|\mathcal{T} - [\![\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}]\!]\|_F^2$$
$$= \frac{1}{2}\|\mathbf{T}^{(n)}_{I_n, I_{-n}} - \mathbf{A}^{(n)}\mathbf{Z}^{(-n)^\top}\|_F^2$$

- Gradient matrices of $\mathcal{F}$ with respect to $\mathbf{A}^{(n)}$, $\forall n = 1, \ldots, N$

$$\nabla_n \mathcal{F}(\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}) = -(\mathbf{T}^{(n)}_{I_n, I_{-n}} - \mathbf{A}^{(n)}\mathbf{Z}^{(-n)^\top})\mathbf{Z}^{(-n)}$$

Regularization terms

- $\mathcal{R}_n(\mathbf{A}^{(n)})$: block dependent penalty terms enforcing sparsity and nonnegativity

$$\mathcal{R}_n(\mathbf{A}^{(n)}) = \sum_{i_n=1}^{I_n} \sum_{r=1}^{R} \rho_n(a_{i_n r}^{(n)}) \qquad \forall n \in \{1, \dots, N\}$$

where loading matrices $\mathbf{A}^{(n)} = (a_{i_n r}^{(n)})_{(i_n, r) \in \{1,\dots,I_n\} \times \{1,\dots,R\}}$

$$\rho_n(\omega) = \begin{cases} \alpha^{(n)} |\omega|^{\pi^{(n)}} & \text{if } \eta_{\min}^{(n)} \leq \omega \leq \eta_{\max}^{(n)} \\ +\infty & \text{otherwise} \end{cases}$$

$\alpha^{(n)} \in]0, +\infty[$, $\pi^{(n)} \in \mathbb{N}^*$, $\eta_{\min}^{(n)} \in [-\infty, +\infty[$ and $\eta_{\max}^{(n)} \in [\eta_{\min}^{(n)}, +\infty]$
$\Rightarrow$ block dependent but constant within a block regularization parameters

Preconditioning

- Preconditioner matrix $\mathbf{P}$ for the n-th block, $\forall n \in \{1, \ldots, N\}$

$$\mathbf{P}^{(n)}(\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}) = \mathbf{A}^{(n)}(\mathbf{Z}^{(-n)\top}\mathbf{Z}^{(-n)}) \oslash \mathbf{A}^{(n)}$$

$\forall n \in \{1, \ldots, N\}$, $\mathbf{A}^{(n)}$ must be non zero
$\oslash$: Hadamard entry-wise division
(Preconditioning: extension of the one used in NMF [Lee and Seung, 2001])

Proximity operator

Proximity operator of $\mathcal{R}_n$ associated with $\mathbf{P}^{(n)}$

$$\mathrm{prox}_{\gamma[k]^{-1}\mathbf{P}^{(n)}[k],\mathcal{R}_n}(y) = \left(\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)}[k],\rho_n}(y^{(i)})\right)_{i\in\{1,\ldots,RI_n\}}$$

$(\forall y = (y^{(i)})_{i\in\{1,\ldots,RI_n\}} \in \mathbb{R}^{RI_n})$, where $(\forall i \in \{1,\ldots,RI_n\})$, $(\forall v \in \mathbb{R})$

$$\boxed{\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)},\rho_n}(v) = \min\left\{\eta_{\max}^{(n)}, \max\left\{\eta_{\min}^{(n)}, \mathrm{prox}_{\gamma[k]\alpha^{(n)}(p_i^{(n)}[k])^{-1}|\,.\,|^{\pi^{(n)}}}(v)\right\}\right\}}$$

(separable structure, diagonal preconditioning matrices, componentwise calculation)

Proximity operator

Proximity operator of $\mathcal{R}_n$ associated with $\mathbf{P}^{(n)}$

$$\mathrm{prox}_{\gamma[k]^{-1}\mathbf{P}^{(n)}[k],\mathcal{R}_n}(y) = \left(\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)}[k],\rho_n}(y^{(i)})\right)_{i\in\{1,\dots,RI_n\}}$$

$(\forall y = (y^{(i)})_{i\in\{1,\dots,RI_n\}} \in \mathbb{R}^{RI_n})$, where $(\forall i \in \{1,\dots,RI_n\})$, $(\forall v \in \mathbb{R})$

$$\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)},\rho_n}(v) = \min\left\{\eta_{\max}^{(n)}, \max\left\{\eta_{\min}^{(n)}, \mathrm{prox}_{\gamma[k]\alpha^{(n)}(p_i^{(n)}[k])^{-1}|\,.\,|^{\pi^{(n)}}}(v)\right\}\right\}$$

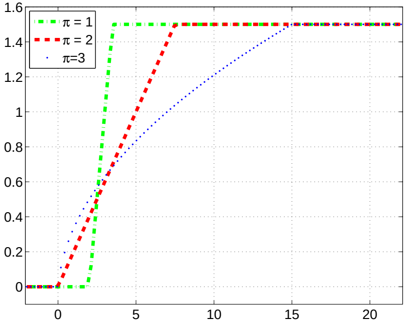
Example:

$\mathrm{prox}_{\rho_n}(v)$ where $v \in [-2, 22]$, $[\eta_{\min}^{(n)}, \eta_{\max}^{(n)}] = [0, 1.5]$, $\alpha^{(n)} = 2$ and

1) $\pi^{(n)} = 1$

2) $\pi^{(n)} = 2$

3) $\pi^{(n)} = 3$



Experiments on simulated data

- Simulated tensor $\overline{\mathcal{T}}$: (uni or bimodal type) emission and excitation spectra, $\overline{R} = 5$
- Simulated observed tensor: $\mathcal{T} = \overline{\mathcal{T}} + \mathcal{B}$, $\mathcal{B}$: white Gaussian noise
- 2 considered cases:
  1. **3D tensor**: $\overline{\mathcal{T}} \in \mathbb{R}_+^{100\times100\times100}$
     + Noiseless case: no noise added, $\widehat{R} = 6$ (overestimation)
  2. **4D tensor**: $\overline{\mathcal{T}} \in \mathbb{R}_+^{100\times100\times100\times100}$
     + Noisy case: SNR $= 18.46$ dB, $\widehat{R} = 7$ (overestimation)
- Error measure:
  1. Signal to Noise Ratio defined as $\text{SNR} = 20\log_{10}\frac{\|\overline{\mathcal{T}}\|_F}{\|\widehat{\mathcal{T}}-\overline{\mathcal{T}}\|_F}$
  2. Estimation error: $E_1 = 10\log_{10}\left(\frac{\sum_{n=1}^{N}\|\widehat{\mathbf{A}}^{(n)}(1{:}\overline{R})-\bar{\mathbf{A}}^{(n)}\|_1}{\sum_{n=1}^{N}\|\bar{\mathbf{A}}^{(n)}\|_1}\right)$
  3. Over-factoring error $E_2 = 10\log_{10}\left(\|\sum_{r=\overline{R}+1}^{\widehat{R}}\widehat{\mathbf{a}}_r^{(1)}\circ\ldots\circ\widehat{\mathbf{a}}_r^{(N)}\|_1\right)$

Numerical results - 3D tensor

Noisy case

Elapsed time (s)	BC-VMFB	N-way	fast HALS
For 50 iterations	0.2	11	0.5
To reach stopping conditions ($\mathbf{E}_1$, $\mathbf{E}_2$) dB	75 (-11.2, -409)	8 (-12.5, 30.6)	8 (-12.5, 30.6)

Noiseless case

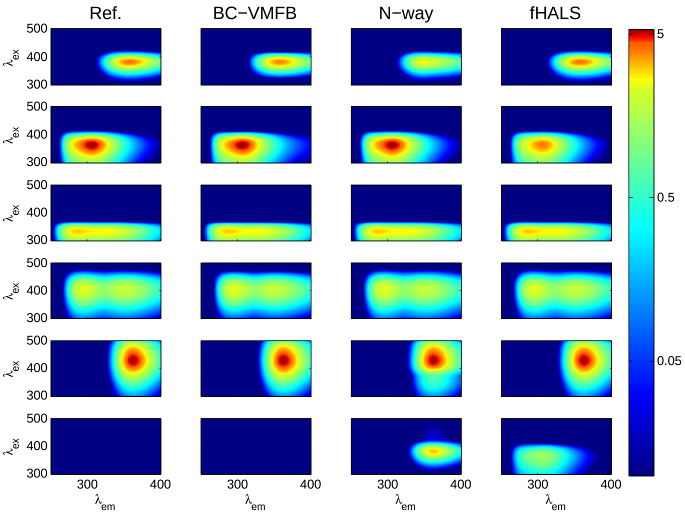
Elapsed time (s)	BC-VMFB	N-way	fast HALS
To reach stopping conditions ($\mathbf{E}_1$, $\mathbf{E}_2$) dB	74 (-15, -409)	80 (-8.7, 31.7)	3.7 (-6.1, 31.7)

Computation time comparison: BC-VMFB (with penalty), N-way [Bro, 1997], fast HALS [Phan et al., 2013] using the same initial value

BC-VMFB:

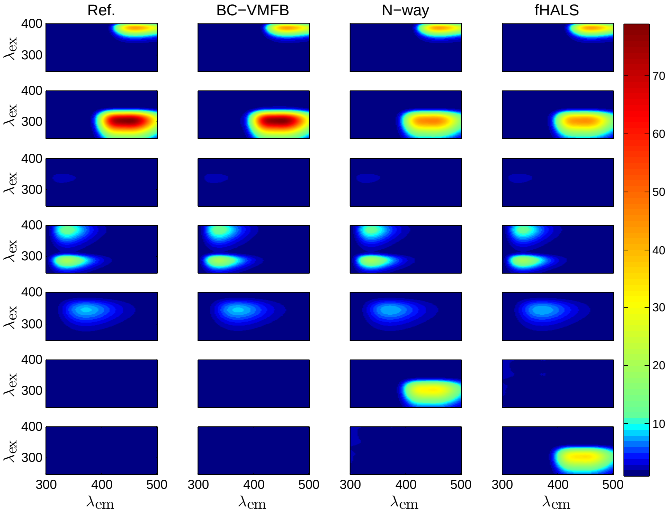
+ Fastest computation time / iteration
+ Smallest estimation error E_1 (noisy case), overestimation error E_2 (both cases)

Visual results: 3D tensor, noiseless case



Penalized BC-VMFB $\alpha = 0.05$

Visual results: 4D tensor, noisy case



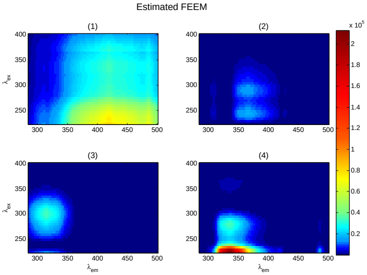
Computer simulation: real experimental data - water monitoring to detect pollutants

- Data were acquired automatically every 3 minutes, during a 10 days monitoring campaign performed on water extracted from an urban river $\Rightarrow$ tensor of size $36 \times 111 \times 2594$.
- The excitation wavelengths range from 225nm to 400nm with a 5nm bandwidth, whereas the emission wavelengths range from 280nm to 500nm with a 2nm bandwidth.
- The FEEM have been pre-processed using the Zepp's method (negative values were set to 0).

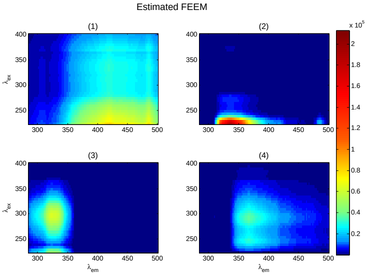
Contamination

During this experiment, a contamination with diesel oil appeared 7 days after the beginning of the monitoring.

Results: assuming that $\widehat{R} = 4$

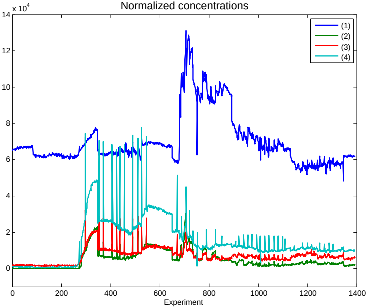


penalized BC-VMFB algorithm

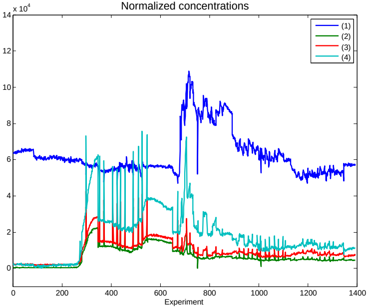


Bro's *N*-way algorithm

Results: concentrations



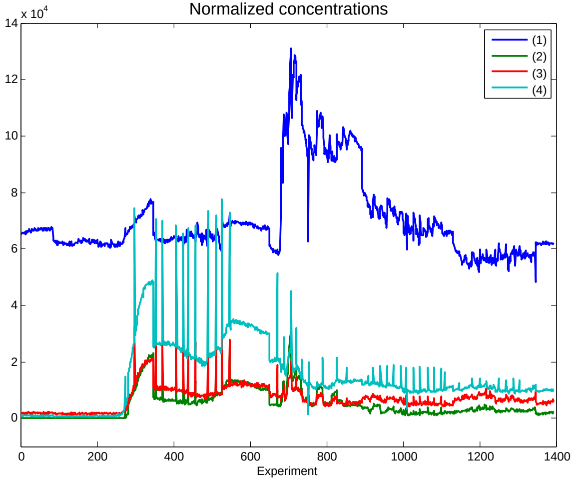
penalized BC-VMFB algorithm



Bro's N-way algorithm

Case $\widehat{R} = 4$

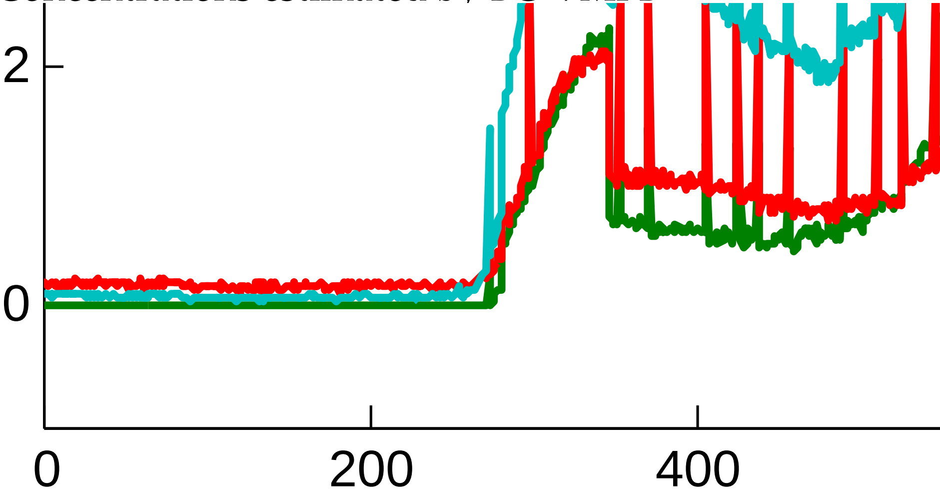
Concentrations estimated by BC-VMFB



↑ Day 7

Case $\widehat{R} = 4$

Concentrations estimated by BC-VMFB



Conclusions

- Inverse problems study from model to resolution through parameterization.
- Performance study on simulated data but also on real data.
- Elaboration of efficient methods based on wavelets, optimization, proximal algorithms.

- Numerous fruitful exchanges with biologists, chimists, physicists, doctors.
- Interdisciplinary projects.
- Opportunities for fruitful collaborations and shared student supervision.

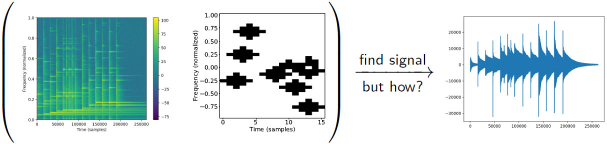
a lot remains (hopefully) to be done ...

Perspectives

> I can not imagine the future without inversion, who has never dreamed of inverting time?

A lot of projects in progress

- Audio inpainting;



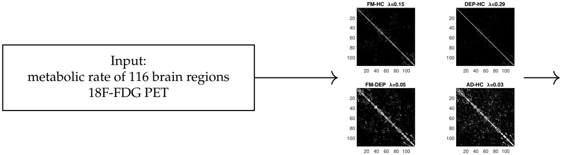
- Graphical models;
- NMR spectroscopy.

Perspectives

> I can not imagine the future without inversion, who has never dreamed of inverting time?

A lot of projects in progress

- Audio inpainting;
- Graphical models;



Pair	Group	NB	SVM	S-LDA
1	FM	69.23	**86.02**	84.62
	HC	58.93	85.54	**94.64**
2	DEP	90.62	91.18	**93.75**
	HC	87.50	90.99	**96.43**
3	FM	90.38	95.34	**98.08**
	DEP	81.25	94.79	**96.88**
4	AD	73.68	93.24	**97.37**
	HC	82.14	93.58	**98.21**

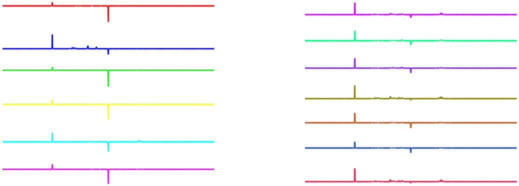
- NMR spectroscopy.

Perspectives

I can not imagine the future without inversion, who has never dreamed of inverting time?

A lot of projects in progress

- Audio inpainting;
- Graphical models;
- NMR spectroscopy.

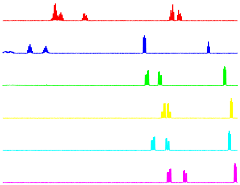


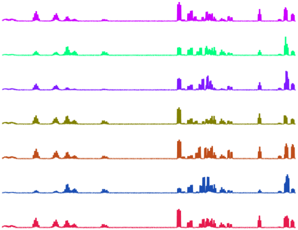
Perspectives

> I can not imagine the future without inversion, who has never dreamed of inverting time?

A lot of projects in progress

- Audio inpainting;
- Graphical models;
- NMR spectroscopy.





Perspectives

> I can not imagine the future without inversion, who has never dreamed of inverting time?

A lot of projects in progress

- Audio inpainting;
- Graphical models;
- NMR spectroscopy.

Thank you!

Appendix

Proximity operator

Definition

Let φ be a lower semi-continuous convex function. For all $x \in \mathbb{R}^N$, prox_φ is the unique minimizer of

$$y \mapsto \varphi(y) + \frac{1}{2}\|x - y\|^2$$

♣

Examples:

C a non-empty closed convex subset of $\mathbb{R}^N$.

$$\begin{aligned}\mathrm{prox}_{\iota_C}(x) &= \underset{y \in \mathbb{R}^N}{\text{minimize}}\ \iota_C(y) + \frac{1}{2}\|x - y\|^2 \\ &= \underbrace{\underset{y \in C}{\text{minimize}}\ \|x - y\|^2}_{\Pi_C(x):\text{projection operator onto } C}\end{aligned}$$

Proximity operator

Definition

Let φ be a lower semi-continuous convex function. For all $x \in \mathbb{R}^N$, prox_φ is the unique minimizer of

$$y \mapsto \varphi(y) + \frac{1}{2}\|x - y\|^2$$

♣

Examples:

C a non-empty closed convex subset of $\mathbb{R}^N$.

$$\begin{aligned}\text{prox}_{\iota_C}(x) &= \underset{y \in \mathbb{R}^N}{\text{minimize}}\ \iota_C(y) + \frac{1}{2}\|x - y\|^2 \\ &= \underbrace{\underset{y \in C}{\text{minimize}}\ \|x - y\|^2}_{\Pi_C(x):\text{projection operator onto } C}\end{aligned}$$

Proximity operator

Definition

Let φ be a lower semi-continuous convex function. For all $x \in \mathbb{R}^N$, prox_φ is the unique minimizer of

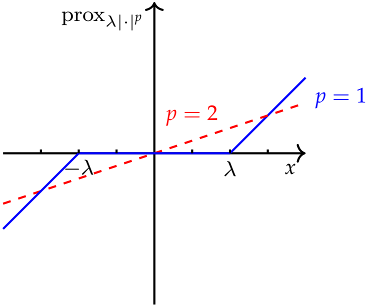
$$y \mapsto \varphi(y) + \frac{1}{2}\|x - y\|^2$$

Examples:

$(\forall x \in \mathbb{R})$

$a)$ $\text{prox}_{\lambda|\cdot|^2}(x) = \underbrace{\frac{1}{1+2\lambda}x}_{\text{“Wiener” filter}}$

$b)$ $\text{prox}_{\lambda|\cdot|}(x) = \underbrace{\text{sign}(x)\max\left(|x| - \lambda, 0\right)}_{\text{shrinkage operator}}$



Proximity operator

- let $\varphi : \mathbb{R} \to]-\infty, +\infty]$ be a proper lower semi-continuous function. The proximity operator is defined as

$$\mathrm{prox}_{\varphi} : \mathbb{R} \to \mathbb{R} : v \mapsto \arg\min_{u \in \mathbb{R}} \frac{1}{2} \|u - v\|^2 + \varphi(u),$$

- let $\varphi : \mathbb{R}^L \to]-\infty, +\infty]$ be a proper lower semi-continuous function. The proximity operator associated with a Symmetric Positive Definite (SPD) matrix **P** is defined as

$$\mathrm{prox}_{\mathbf{P},\varphi} : \mathbb{R}^L \to \mathbb{R}^L : \mathbf{v} \mapsto \arg\min_{\mathbf{u} \in \mathbb{R}^L} \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_{\mathbf{P}}^2 + \varphi(\mathbf{u}),$$

where $\forall \mathbf{x} \in \mathbb{R}^L$, $\|\mathbf{x}\|_{\mathbf{P}}^2 = \langle \mathbf{x}, \mathbf{P}\mathbf{x} \rangle$ and $\langle \cdot, \cdot \rangle$ is the inner product.

Remark : Note that if **P** reduces to the identity matrix, then the two definitions coincides.

Marine seismic data acquisition

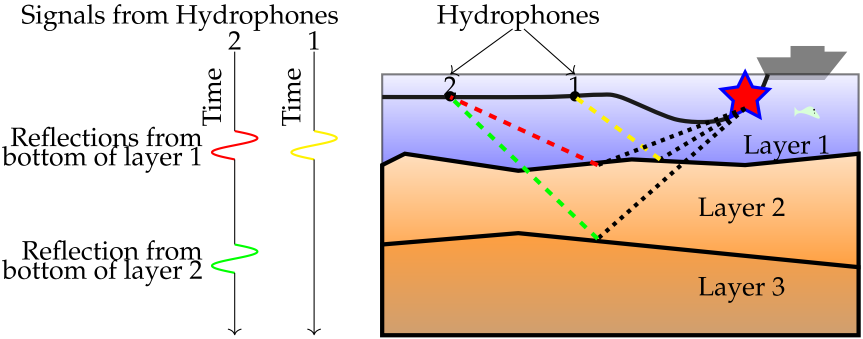
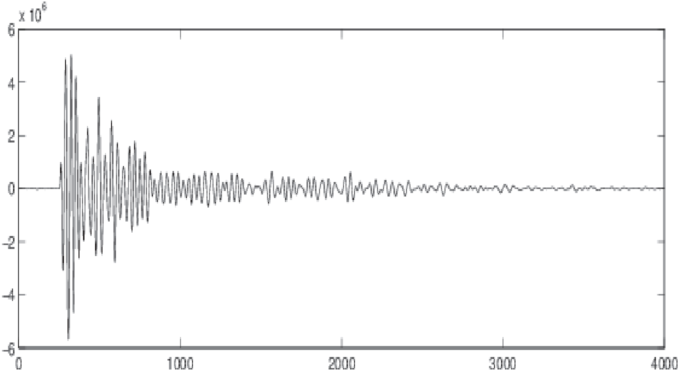
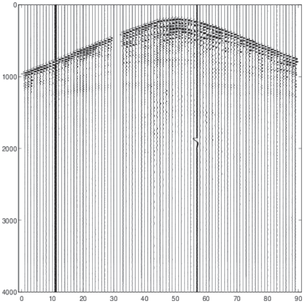


Figure: The seismic reflection method.

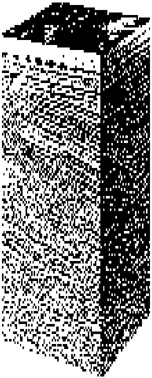
Resulting seismic nD data



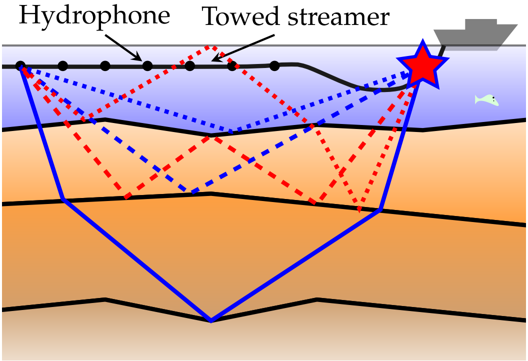
Resulting seismic nD data



Resulting seismic nD data



Seismic multiple reflection



Solid blue: primaries; dashed red: multiple reflection disturbances.

Problem reformulation

$$\underbrace{z}_{\text{observed signal}} = \mathbf{R}\underbrace{\bar{\mathbf{h}}}_{\text{filter}} + \underbrace{\bar{y}}_{\text{primary}} + \underbrace{b}_{\text{noise}}$$

where

- $\bar{s} = \sum_{j=0}^{J-1} R_j \bar{h}_j = \mathbf{R}\bar{\mathbf{h}} = \left[\bar{s}^{(0)}, \; \cdots \; , \; \bar{s}^{(N-1)}\right]^\top$
- $\mathbf{R} = [R_0 \cdots R_{J-1}]$, R_j is a block diagonal matrix
- $\bar{\mathbf{h}} = \left[\bar{h}_0^\top \cdots \bar{h}_{J-1}^\top\right]^\top$
- $\bar{h}_j^{(n)} = \left[\bar{h}_j^{(0)}(p') \; \cdots \; \bar{h}_j^{(0)}(p' + P_j - 1) \; \cdots \; \bar{h}_j^{(N-1)}(p') \; \cdots \; \bar{h}_j^{(N-1)}(p' + P_j - 1)\right]^\top$

Estimation of y

Assumption: $\bar{y}$ is a realization of a random vector Y, whose probability density is given by:

$$(\forall y \in \mathbb{R}^N) \qquad f_Y(y) \propto \exp(-\varphi(Fy))$$

$F \in \mathbb{R}^{K\times N}$: linear operator.
φ is chosen separable:

$$\big(\forall x = (x_k)_{1\le k\le K} \in \mathbb{R}^K\big) \qquad \varphi(x) = \sum_{k=1}^{K} \varphi_k(x_k)$$

where, for all $k \in \{1, \ldots, K\}$, $\varphi_k \colon \mathbb{R} \to]-\infty, +\infty]$.

Estimation : filter $\mathbf{h}$ and noise b

- **Assumption**: $\bar{\mathbf{h}}$ is a realization of a random vector H, whose probability density can be expressed as:
  $(\forall \mathbf{h} \in \mathbb{R}^{NP})\ f_H(\mathbf{h}) \propto \exp(-\rho(\mathbf{h}))$
  H is independent of Y.
- **Assumption**: b is a realization of a random vector B, of probability density:

$$(\forall b \in \mathbb{R}^N) \qquad f_B(b) \propto \exp(-\psi(b))$$

B is assumed to be independent from Y and H

Estimation : filter $\mathbf{h}$ and noise b

- **Assumption**: $\bar{\mathbf{h}}$ is a realization of a random vector H, whose probability density can be expressed as:
  $(\forall \mathbf{h} \in \mathbb{R}^{NP})\ f_H(\mathbf{h}) \propto \exp(-\rho(\mathbf{h}))$
  H is independent of Y.
- **Assumption**: b is a realization of a random vector B, of probability density:

$$(\forall b \in \mathbb{R}^N) \qquad f_B(b) \propto \exp(-\psi(b))$$

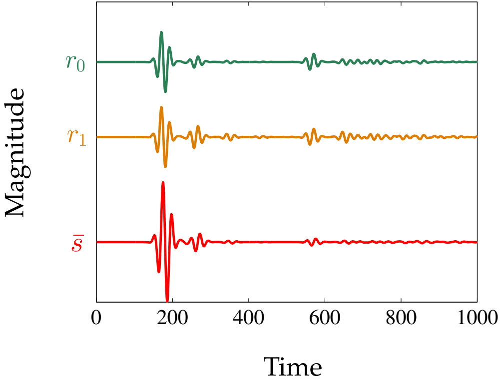
B is assumed to be independent from Y and H

MAP estimation of $(y, \mathbf{h})$

$$\underset{y\in\mathbb{R}^N, \mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}} \quad \underbrace{\psi(z - \mathbf{R}\mathbf{h} - y)}_{\text{fidelity: linked to noise}} + \underbrace{\varphi(Fy)}_{\textit{a priori} \text{ on the signal}} + \underbrace{\rho(\mathbf{h})}_{\textit{a priori} \text{ on the filters}}$$

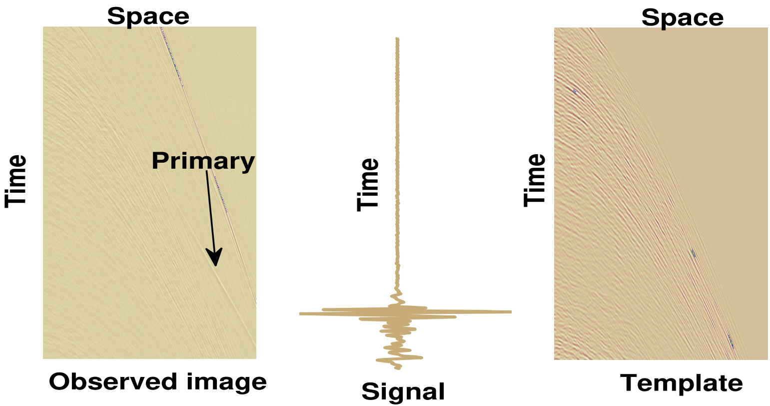
♣

Template



First model, Second model, Multiple

Template


Space
Time
Primary
Observed image
Time
Signal
Space
Time
Template

Sparse representations

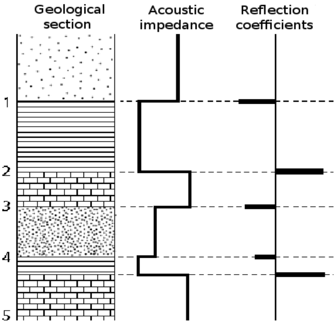
- *Dictionary*: atoms $e_k \in \mathbb{R}^N$, frame analysis operator F
- *Decomposition*: $x = (\langle y \mid e_k \rangle)_{k \in \mathbb{K}} = Fy$
- *Ideal sparsity*: Most x_k are zero

$$\ell_0(x) = \#\{k \ : \ x_k \neq 0\}$$

Applications

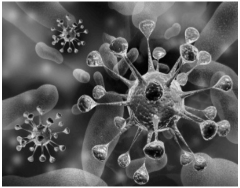
- Discovering the structure of natural images
  Pre-processing, quantization...
- Sparse models for signal processing
  Denoising, coding, restoration, compression...
- Optimization algorithms
  Matching pursuit algorithm [Mallat and Zhang, 1993], iterative hard-thresholding [Starck *et al.*, 2003], proximal splitting techniques [Combettes and Pesquet, 2011]

Sparse signal

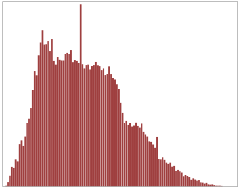


Underground structure

Introducing sparsity



Original image



Histogram of original image

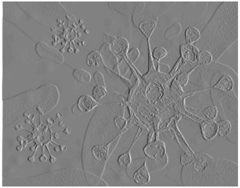
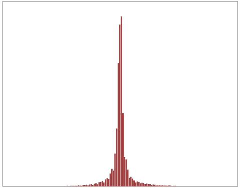


Image gradient magnitude



Histogram of gradients

Some sparse representations

Some tight frames:

- Orthonormal wavelet transform: wavelet basis
- Union of m orthonormal wavelet transforms
- ...

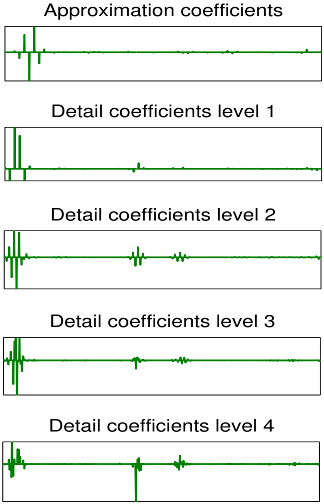
Some non tight frames:

- Biorthogonal wavelet transform
- Dual-tree and complex dual-tree transforms
- ...

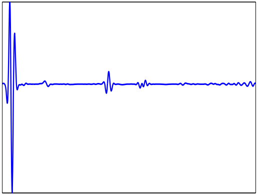
Additional references & some toolboxes & links

- http://arxiv.org/abs/1101.5320: L. Jacques, L. Duval, C. Chaux et G. Peyré, A Panorama on Multiscale Geometric Representations, Intertwining Spatial, Directional and Frequency Selectivity (Signal Processing, Vol. 91, No. 12, 2011)
- http://www.laurent-duval.eu/siva-panorama-multiscale-geometric-representations.html

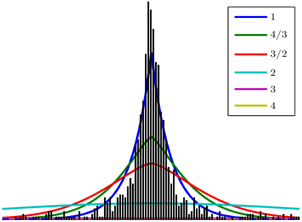
Evaluations of the sparsity



Wavelet coefficients (Fy)



Seismic data (y)



Histogram of wavelet coefficients

About convex set C

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi\big(z-\mathbf{Rh}-y\big)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$C = C_1 \cap C_2 \cap C_3$$

- $C_1 = \Big\{\mathbf{h} \in \mathbb{R}^{PN}\ :\ \rho(\mathbf{h}) = \sum_{j=0}^{J-1} \rho_j(h_j) \le \tau\Big\}$
- $\rho_j(h_j) = \|h_j\|_{\ell_2}^2 = \sum_{n=0}^{N-1} \sum_{p=p'}^{p'+P_j-1} |h_j^{(n)}(p)|^2$

About convex set C

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi\big(z-\mathbf{R}\mathbf{h}-y\big)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$C = C_1 \cap C_2 \cap C_3$$

- $C_1 = \left\{ \mathbf{h} \in \mathbb{R}^{PN} \ : \ \rho(\mathbf{h}) = \sum_{j=0}^{J-1} \rho_j(h_j) \leq \tau \right\}$
- $\rho_j(h_j) = \|h_j\|_{\ell_2}^2 = \sum_{n=0}^{N-1} \sum_{p=p'}^{p'+P_j-1} |h_j^{(n)}(p)|^2$
- $\rho_j(h_j) = \|h_j\|_{\ell_1} = \sum_{n=0}^{N-1} \sum_{p=p'}^{p'+P_j-1} |h_j^{(n)}(p)|$

About convex set C

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi(z-\mathbf{R}\mathbf{h}-y)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$C = C_1 \cap C_2 \cap C_3$$

- $C_1 = \left\{\mathbf{h}\in\mathbb{R}^{PN}\ :\ \rho(\mathbf{h}) = \sum_{j=0}^{J-1}\rho_j(h_j) \le \tau\right\}$
- $\rho_j(h_j) = \|h_j\|_{\ell_2}^2 = \sum_{n=0}^{N-1}\sum_{p=p'}^{p'+P_j-1}|h_j^{(n)}(p)|^2$
- $\rho_j(h_j) = \|h_j\|_{\ell_1} = \sum_{n=0}^{N-1}\sum_{p=p'}^{p'+P_j-1}|h_j^{(n)}(p)|$
- $\rho_j(h_j) = \|h_j\|_{\ell_{1,2}} = \sum_{n=0}^{N-1}\left(\sum_{p=p'}^{p'+P_j-1}|h_j^{(n)}(p)|^2\right)^{1/2}$

Hard constraints on the filters $C_2,\ C_3$

Problem to be solved

$$\underset{y\in\mathbb{R}^N,\mathbf{h}\in\mathbb{R}^{NP}}{\text{minimize}}\ \psi(z-\mathbf{Rh}-y)+\iota_D(Fy)+\iota_C(\mathbf{h})$$

$$C = C_1 \cap C_2 \cap C_3$$

Assumption: slow variations of the filters along time.

$$\big(\forall(j,n,p)\big) \qquad |h_j^{(n+1)}(p)-h_j^{(n)}(p)| \le \varepsilon_{j,p}$$

For computational issues, $h \in C_2 \cap C_3$ where

$$C_2 = \left\{h \mid \forall p,\ \forall n \in \left\{0,\dots,\left\lfloor \frac{N}{2} \right\rfloor - 1\right\} \quad \left|h^{(2n+1)}(p) - h^{(2n)}(p)\right| \le \varepsilon_p \right\}$$

$$C_3 = \left\{h \mid \forall p,\ \forall n \in \left\{1,\dots,\left\lfloor \frac{N-1}{2} \right\rfloor\right\} \quad \left|h^{(2n)}(p) - h^{(2n-1)}(p)\right| \le \varepsilon_p \right\}$$

Projections onto a set of hyperslabs of $\mathbb{R}^2$

Projections onto C_2 and C_3 : let $h \in \mathbb{R}^{N_t P}$ and let $g_1 = \Pi_{C_2}(h)$; and $g_2 = \Pi_{C_3}(h)$; then for every $p \in \{p', \ldots, p' + P - 1\}$ and for every $n \in \left\{0, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1\right\}$,

1. if $|h^{(2n+1)}(p) - h^{(2n)}(p)| < \varepsilon_p$, then

$$g_1^{(2n)}(p) = h^{(2n)}(p), \qquad g_1^{(2n+1)}(p) = h^{(2n+1)}(p);$$
$$g_2^{(2n)}(p) = h^{(2n)}(p), \qquad g_2^{(2n-1)}(p) = h^{(2n-1)}(p)$$

2. if $h^{(2n+1)}(p) - h^{(2n)}(p) > \varepsilon_p$, then

$$g_1^{(2n)}(p) = \frac{h^{(2n+1)}(p) + h^{(2n)}(p)}{2} - \frac{\varepsilon_p}{2} \quad g_1^{(2n+1)}(p) = \frac{h^{(2n+1)}(p) + h^{(2n)}(p)}{2} + \frac{\varepsilon_p}{2};$$
$$g_2^{(2n)}(p) = \frac{h^{(2n)}(p) + h^{(2n-1)}(p)}{2} + \frac{\varepsilon_p}{2} \quad g_2^{(2n-1)}(p) = \frac{h^{(2n)}(p) + h^{(2n-1)}(p)}{2} - \frac{\varepsilon_p}{2}$$

3. if $h^{(2n+1)}(p) - h^{(2n)}(p) < -\varepsilon_p$, then

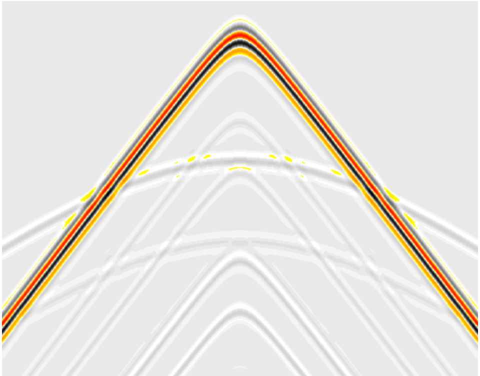
$$g_1^{(2n)}(p) = \frac{h^{(2n+1)}(p) + h^{(2n)}(p)}{2} + \frac{\varepsilon_p}{2} \quad g_1^{(2n+1)}(p) = \frac{h^{(2n+1)}(p) + h^{(2n)}(p)}{2} - \frac{\varepsilon_p}{2};$$
$$g_2^{(2n)}(p) = \frac{h^{(2n)}(p) + h^{(2n-1)}(p)}{2} - \frac{\varepsilon_p}{2} \quad g_2^{(2n-1)}(p) = \frac{h^{(2n)}(p) + h^{(2n-1)}(p)}{2} + \frac{\varepsilon_p}{2}.$$

Summary

		σ	0.04	0.08	0.16
$\bar{y} - z$		$\ell_1(\times 10^{-2})$	3.88	6.89	13.09
[Ventosa *et al.*, 2012]		$\ell_1(\times 10^{-2})$	5.38	7.87	13.36
$\rho_j = \ell_2$	OR-basis	$\ell_1(\times 10^{-2})$	1.53	2.27	3.34
	SI frame$^{(*)}$	$\ell_1(\times 10^{-2})$	1.19	1.69	2.42
	M-band	$\ell_1(\times 10^{-2})$	1.07	1.41	1.96
$\rho_j = \ell_1$	OR-basis	$\ell_1(\times 10^{-2})$	1.66	2.33	3.37
	SI frame$^{(*)}$	$\ell_1(\times 10^{-2})$	1.23	1.70	2.39
	M-band	$\ell_1(\times 10^{-2})$	1.14	1.47	2.00
$\rho_j = \ell_{1,2}$	OR-basis	$\ell_1(\times 10^{-2})$	1.51	2.25	3.32
	SI frame$^{(*)}$	$\ell_1(\times 10^{-2})$	1.10	1.58	2.32
	M-band	$\ell_1(\times 10^{-2})$	0.95	1.31	1.87

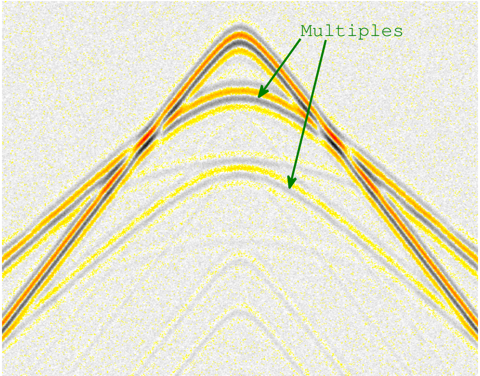
Comparison of the estimated primaries with the 2D proposed version$^{(*)}$ in using three different 2D wavelet transforms, over three noise levels, and three a priori functions $\rho_j \in \{\ell_2, \ell_1, \ell_{1,2}\}$, with [Ventosa *et al.*, 2012]

Synthetic data (2D)



Primary: $\overline{y}$
- size 512×512

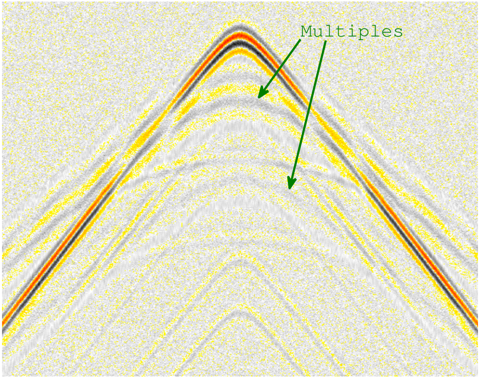
Synthetic data (2D)



Primary: $\overline{y}$
- size 512×512

Observed image: z
- Noise: $\sigma = 0.08$
- SNR = 1.13 dB
- SSIM = 0.16

Synthetic data (2D)



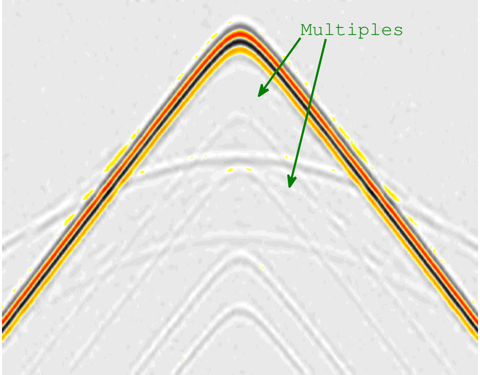
Primary: $\overline{y}$
- size 512×512

Observed image: z
- Noise: $\sigma = 0.08$
- SNR = 1.13 dB
- SSIM = 0.16

Reconstructed image by

[Ventosa *et al.*, 2012]
- SNR = 2.38 dB
- SSIM = 0.13

Synthetic data (2D)



Primary: $\overline{y}$
- size 512×512

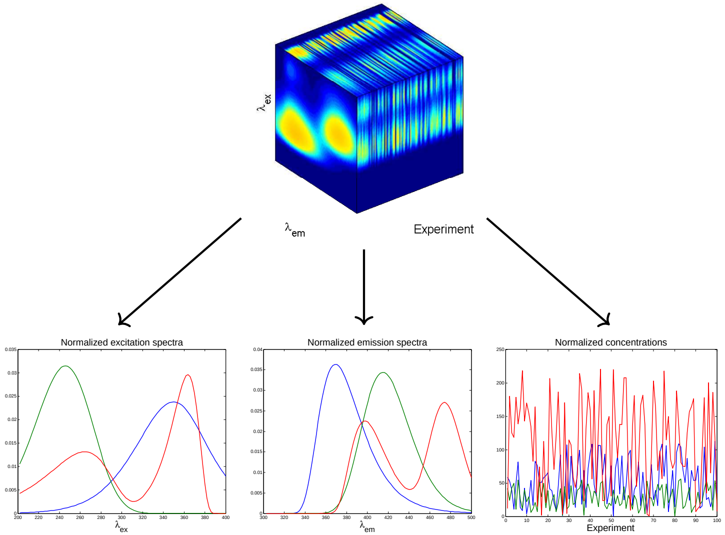
Observed image: z
- Noise: $\sigma = 0.08$
- SNR = 1.13 dB
- SSIM = 0.16

Reconstructed image by

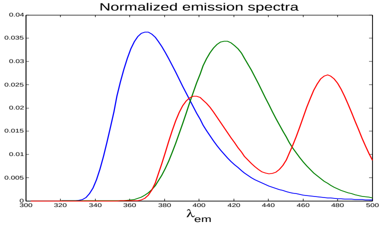
[Ventosa *et al.*, 2012]
- SNR = 2.38 dB
- SSIM = 0.13

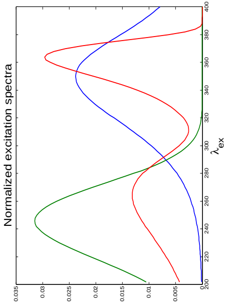
Our method
- SNR = 17.00 dB
- SSIM = 0.74

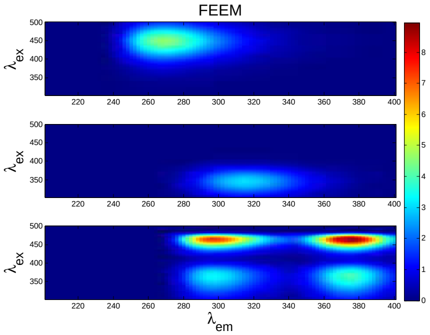
3D fluorescence spectroscopy



Coumpounds characterisation







Standard operations

- Outer product: let $\mathbf{u} \in \mathbb{R}^I, \mathbf{v} \in \mathbb{R}^J$,

$$\boxed{\mathbf{u} \circ \mathbf{v} = \mathbf{u}\mathbf{v}^\top} \in \mathbb{R}^{I \times J}$$

- Khatri-Rao product: let $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_J] \in \mathbb{R}^{I \times J}$ and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_J] \in \mathbb{R}^{K \times J}$

$$\boxed{\mathbf{U} \odot \mathbf{V} = [\mathbf{u}_1 \otimes \mathbf{v}_1, \mathbf{u}_2 \otimes \mathbf{v}_2, \ldots, \mathbf{u}_J \otimes \mathbf{v}_J]} \in \mathbb{R}^{IK \times J}.$$

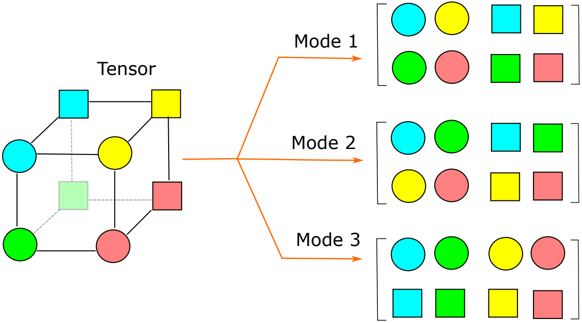
  where $\mathbf{u} \otimes \mathbf{v} = [u_1\mathbf{v}; \ldots; u_I\mathbf{v}] \in \mathbb{R}^{IK}$ (Kronecker product).

- Hadamard division: let $\mathbf{U} \in \mathbb{R}^{I \times J}, \mathbf{V} \in \mathbb{R}^{I \times J}$,

$$\boxed{\mathbf{U} \oslash \mathbf{V} = (u_{ij}/v_{ij})_{i,j}} \in \mathbb{R}^{I \times J}$$

Tensor flattening: example

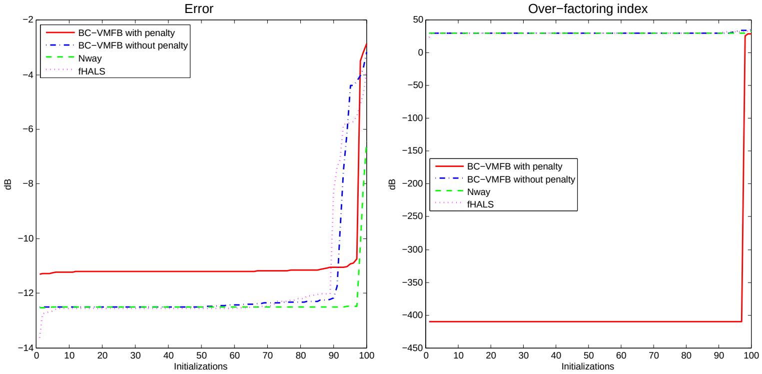
Objective: to handle matrices instead of tensors.



Block coordinate proximal algorithm

Let $\mathbf{x}_0 \in \operatorname{dom} \mathcal{R}$, $k \in \mathbb{N}$ and $\gamma_k \in]0, +\infty[$ *// Initialization step*
for $k = 0, 1, \ldots$ **do** *// k-th iteration of the algorithm*
    Let $j_k \in \{1, \ldots, J\}$ *// Processing of block number* j_k *(chosen, here, according to a quasi cyclic rule)*
    Let $\mathbf{P}_{j_k}(\mathbf{x}_k)$ *be a SPD matrix* *// Construction of the preconditioner* $\mathbf{P}_{j_k}(\mathbf{x}_k)$
    Let $\nabla_{j_k} \mathcal{F}(\mathbf{x}_k)$ *be the Gradient* *// Calculation of Gradient*
    $\tilde{\mathbf{x}}_k^{(j_k)} = \mathbf{x}_k^{(j_k)} - \gamma_k \mathbf{P}_{j_k}(\mathbf{x}_k)^{-1} \nabla_{j_k} \mathcal{F}(\mathbf{x}_k)$ *// Updating of block* j_k *according to a Gradient step*
    $\mathbf{x}_{k+1}^{(j_k)} \in \operatorname{prox}_{\gamma_k^{-1} \mathbf{P}_{j_k}(\mathbf{x}_k), \mathcal{R}_{j_k}} \left(\tilde{\mathbf{x}}_k^{(j_k)} \right)$ *// Updating of block* j_k *according to a Proximal step*
    $\mathbf{x}_{k+1}^{\bar{j}_k} = \mathbf{x}_k^{\bar{j}_k}$ *where* $\bar{j} = \{1, \ldots, J\} \setminus \{j\}$ *// Other blocks are kept unchanged*
end for

Influence of the initialization



Performance versus different initializations (noisy, overestimated case): error index $\mathbf{E}_1$, overfactoring error index $\mathbf{E}_2$

Visual results: noiseless case

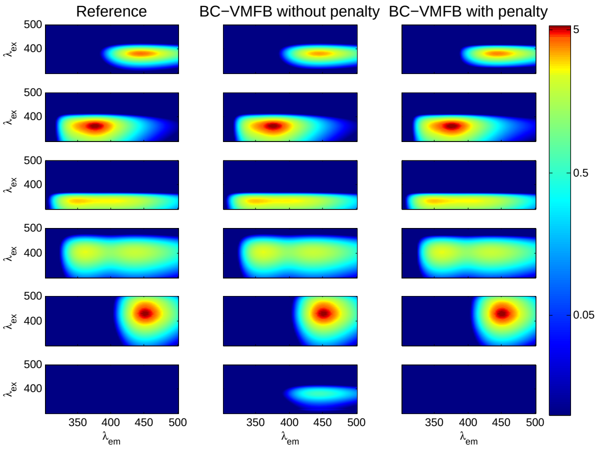


Figure: FEEM of reference (left) - FEEM reconstructed using BC-VMFB without regularization (middle) and with regularization $\alpha = 0.05$ (right).

Visual results: noiseless case

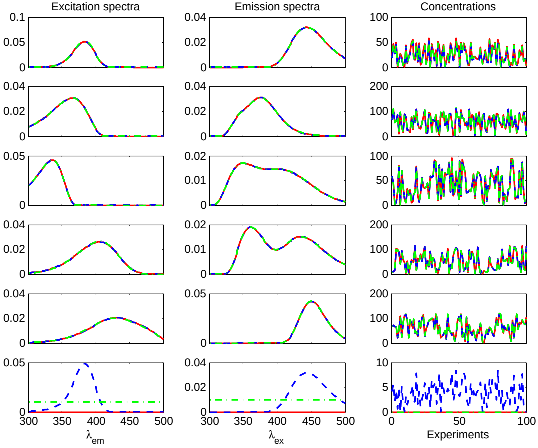


Figure: $\widehat{R} = 6$ - reference spectra / BC-VMFB without penalty / BC-VMFB with penalty $\alpha = 0.05$.

Visual results: noisy case

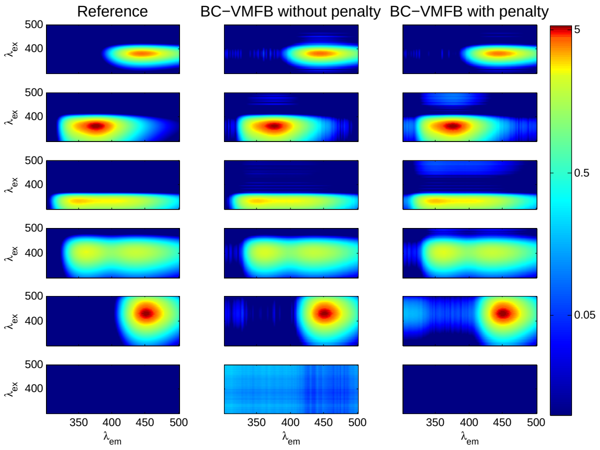


Figure: FEEM of reference (left) - FEEM reconstructed using BC-VMFB without regularization (middle) and with regularization $\alpha = 0.05$ (right).

Visual results: noisy case

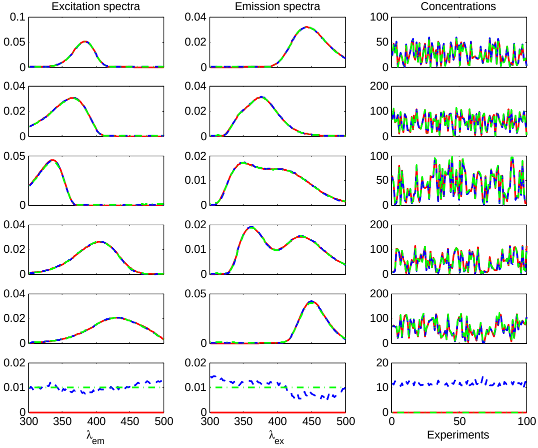
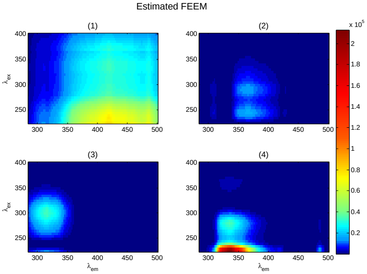
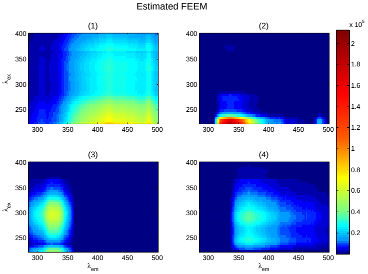


Figure: $\widehat{R} = 6$ - reference spectra / BC-VMFB without penalty / BC-VMFB with penalty $\alpha = 0.05$.

Results: what about the rank?



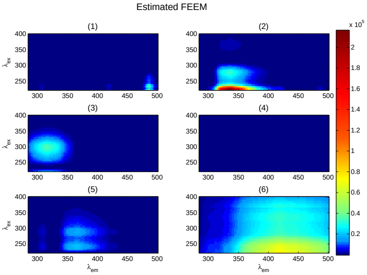
penalized BC-VMFB algorithm

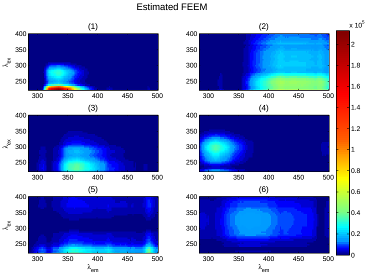


Bro's *N*-way algorithm

Case $\widehat{R} = 4$

Results: what about the rank?

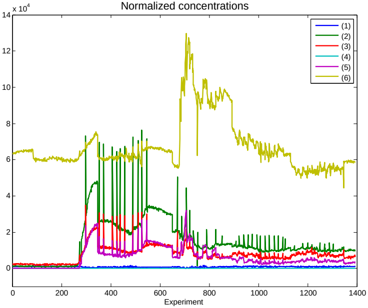


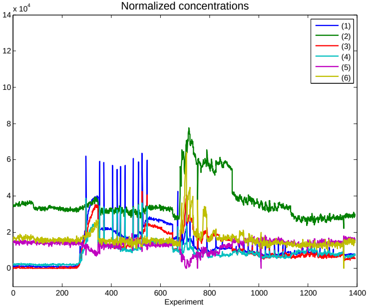


penalized BC-VMFB algorithm

Bro's N-way algorithm

$$\text{Case } \widehat{R} = 6$$



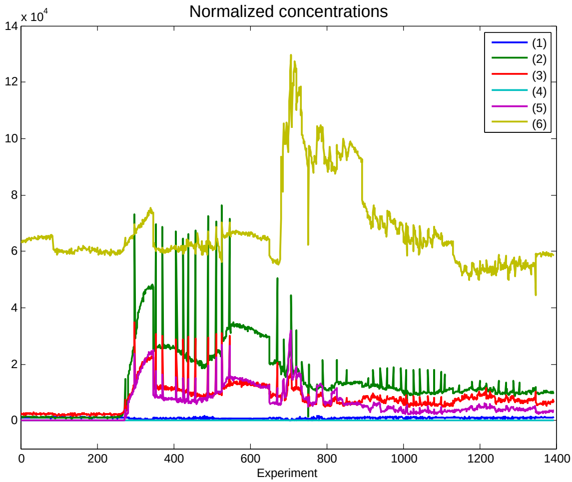


penalized BC-VMFB algorithm

Bro's N-way algorithm

Case $\widehat{R} = 6$

Concentrations estimated by BC-VMFB



Case $\widehat{R} = 6$

Concentrations estimated by BC-VMFB