



# Sensors, Measurement systems and Inverse problems

## DECONVOLUTION

Ali Mohammad-Djafari

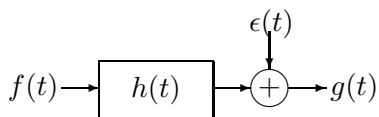
Laboratoire des Signaux et Systèmes,  
UMR8506 CNRS-SUPELEC-UNIV PARIS SUD 11  
SUPELEC, 91192 Gif-sur-Yvette, France  
<http://lss.supelec.free.fr>

Email: [djafari@lss.supelec.fr](mailto:djafari@lss.supelec.fr)  
<http://djafari.free.fr>

# Case study: Signal deconvolution

- ▶ Convolution, Identification and Deconvolution
- ▶ Forward and Inverse problems: Well-posedness and Ill-posedness
- ▶ Naïve methods of Deconvolution
- ▶ Classical methods: Wiener filtering
- ▶ Bayesian approach to deconvolution
- ▶ Simple and Blind Deconvolution
- ▶ Deterministic and probabilistic methods
- ▶ Joint source and canal estimation

# Convolution, Identification and deconvolution



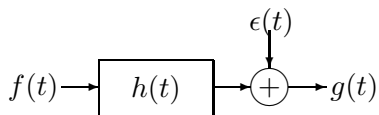
$$g(t) = \int f(t') h(t - t') dt' + \epsilon(t) = \int h(t') f(t - t') dt' + \epsilon(t)$$

- ▶ Convolution: Given  $f$  and  $h$  compute  $g$
- ▶ Identification: Given  $f$  and  $g$  estimate  $h$
- ▶ Deconvolution: Given  $g$  and  $h$  estimate  $f$
- ▶ Blind deconvolution: Given  $g$  estimate both  $h$  and  $f$

## Convolution: Given $f$ and $h$ compute $g$

- ▶ Direct computation:  $g = \text{conv}(h, f)$
- ▶ Fourier domain:  $g(t) = h(t) * f(t) \longrightarrow G(\omega) = H(\omega)F(\omega)$ 
  - ▶ Compute  $H(\omega)$ ,  $F(\omega)$  and  $G(\omega) = H(\omega)F(\omega)$
  - ▶ Compute  $g(t)$  by inverse FT of  $G(\omega)$
- ▶ Take care of dimensions and boarder effects.

## Convolution: Discretization



$$g(t) = \int f(t') h(t - t') dt' + \epsilon(t) = \int h(t') f(t - t') dt' + \epsilon(t)$$

- ▶ The signals  $f(t)$ ,  $g(t)$ ,  $h(t)$  are discretized with the same sampling period  $\Delta T = 1$ ,
- ▶ The impulse response is finite (FIR) :  $h(t) = 0$ , for  $t$  such that  $t < -q\Delta T$  or  $\forall t > p\Delta T$ .

$$g(m) = \sum_{k=-q}^p h(k) f(m - k) + \epsilon(m), \quad m = 0, \dots, M$$

# Convolution: Discretized matrix vector forms

$$\begin{bmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{bmatrix} = \begin{bmatrix} h(p) & \cdots & h(0) & \cdots & h(-q) & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & & \ddots & & \ddots & & & & \vdots \\ \vdots & & & & & & & & & \vdots \\ \vdots & & & & & & & & & \vdots \\ \vdots & & h(p) & \cdots & h(0) & \cdots & h(-q) & & & \vdots \\ \vdots & & \vdots & & \vdots & & \vdots & & & \vdots \\ \vdots & & \vdots & & \vdots & & \vdots & & & \vdots \\ \vdots & & \vdots & & \vdots & & \vdots & & & \vdots \\ \vdots & & \vdots & & \vdots & & \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 & h(p) & \cdots & h(0) & \cdots & h(-q) & 0 \end{bmatrix} \begin{bmatrix} f(-p) \\ \vdots \\ f(0) \\ f(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ f(M) \\ f(M+1) \\ \vdots \\ f(M+q) \end{bmatrix}$$

$$g = Hf + \epsilon$$

- ▶  $g$  is a  $(M + 1)$ -dimensional vector,
- ▶  $f$  has dimension  $M + p + q + 1$ ,
- ▶  $h = [h(p), \dots, h(0), \dots, h(-q)]$  has dimension  $(p + q + 1)$
- ▶  $H$  has dimensions  $(M + 1) \times (M + p + q + 1)$ .

## Convolution: Discretized matrix vector form

- ▶ If system is causal ( $q = 0$ ) we obtain

$$\begin{bmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{bmatrix} = \begin{bmatrix} h(p) & \cdots & h(0) & 0 & \cdots & \cdots & 0 \\ 0 & & & & & & \vdots \\ \vdots & & & & & & \vdots \\ \vdots & & h(p) & \cdots & h(0) & & \vdots \\ \vdots & & & & & & \vdots \\ \vdots & & & & & & 0 \\ 0 & \cdots & \cdots & 0 & h(p) & \cdots & h(0) \end{bmatrix} \begin{bmatrix} f(-p) \\ \vdots \\ f(0) \\ f(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ f(M) \end{bmatrix}$$

- ▶  $\mathbf{g}$  is a  $(M + 1)$ -dimensional vector,
- ▶  $\mathbf{f}$  has dimension  $M + p + 1$ ,
- ▶  $\mathbf{h} = [h(p), \cdots, h(0)]$  has dimension  $(p + 1)$
- ▶  $\mathbf{H}$  has dimensions  $(M + 1) \times (M + p + 1)$ .

## Convolution: Causal systems and causal input

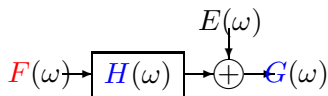
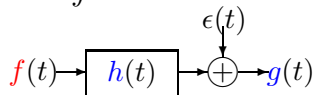
$$\begin{bmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{bmatrix} = \begin{bmatrix} h(0) & & & & & & \\ & h(1) & \cdots & & & & \\ & \vdots & & & & & \\ & h(p) & \cdots & h(0) & & & \\ & 0 & \cdots & & \cdots & & \\ & \vdots & & & & & \\ 0 & \cdots & 0 & h(p) & \cdots & h(0) & \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ f(M) \end{bmatrix}$$

- ▶  $\mathbf{g}$  is a  $(M + 1)$ -dimensional vector,
- ▶  $\mathbf{f}$  has dimension  $M + 1$ ,
- ▶  $\mathbf{h} = [h(p), \dots, h(0)]$  has dimension  $(p + 1)$
- ▶  $\mathbf{H}$  has dimensions  $(M + 1) \times (M + 1)$ .



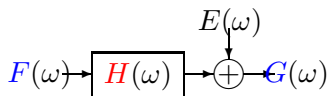
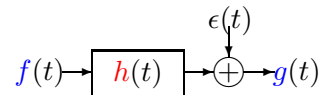
# Convolution, Identification, Deconvolution and Blind deconvolution problems

$$g(t) = \int f(t') h(t - t') dt' + \epsilon(t) = \int h(t') f(t - t') dt' + \epsilon(t)$$



$$G(\omega) = H(\omega) F(\omega) + E(\omega)$$

$$F(\omega) = \frac{G(\omega)}{H(\omega)} + \frac{E(\omega)}{H(\omega)}$$



$$G(\omega) = H(\omega) F(\omega) + E(\omega)$$

$$H(\omega) = \frac{G(\omega)}{F(\omega)} + \frac{E(\omega)}{F(\omega)}$$

- ▶ Convolution: Given  $h$  and  $f$  compute  $g$
- ▶ Identification: Given  $f$  and  $g$  estimate  $h$
- ▶ Simple Deconvolution: Given  $h$  and  $g$  estimate  $f$
- ▶ Blind Deconvolution: Given  $g$  estimate  $h$  and  $f$

## Deconvolution: Given $g$ and $h$ estimate $f$

- ▶ Direct computation:  $f = \text{deconv}(g, h)$
- ▶ Fourier domain: Inverse Filtering  $F(\omega) = \frac{G(\omega)}{H(\omega)}$ 
  - ▶ Compute  $H(\omega)$ ,  $G(\omega)$  and  $F(\omega) = \frac{G(\omega)}{H(\omega)}$
  - ▶ Compute  $g(t)$  by inverse FT of  $F(\omega)$
  
- ▶ Main difficulties: Divide by zero and noise amplification

## Identification: Given $g$ and $f$ estimate $h$

- ▶ Direct computation:

- ▶  $f(t) = \delta(t) \rightarrow g(t) = h(t) \rightarrow h(t) = g(t)$

- ▶  $f(t) = \begin{cases} 0 & t < 0 \\ 1 & t > 0 \end{cases} \rightarrow g(t) = \int_0^t h(t) dt \rightarrow h(t) = \frac{\partial g(t)}{\partial t}$

- ▶ Fourier domain: Inverse Filtering  $H(\omega) = \frac{G(\omega)}{F(\omega)}$

- ▶ Compute  $F(\omega)$ ,  $G(\omega)$  and  $H(\omega) = \frac{G(\omega)}{F(\omega)}$

- ▶ Compute  $h(t)$  by inverse FT of  $H(\omega)$

- ▶ Main difficulties: Divide by zero and noise amplification

# Convolution in 1D and 2D: Signal deconvolution and Image restoration

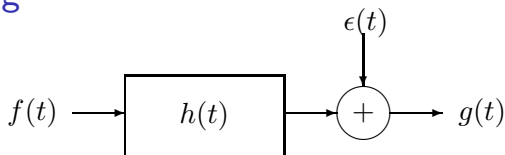
$$\begin{array}{c} \epsilon(t) \\ \downarrow \\ f(t) \longrightarrow \boxed{h(t)} \longrightarrow \oplus \longrightarrow g(t) \end{array}$$
$$g(t) = \iint f(t') h(t - t') dt' + \epsilon(t)$$

- ▶  $f(t)$ ,  $g(t)$  and  $\epsilon(t)$  are modelled as Gaussian random signal

$$\begin{array}{c} \epsilon(x, y) \\ \downarrow \\ f(x, y) \longrightarrow \boxed{h(x, y)} \longrightarrow \oplus \longrightarrow g(x, y) \end{array}$$
$$g(x, y) = \iint f(x', y') h(x - x', y - y') dx' dy' + \epsilon(x, y)$$

- ▶  $f(x, y)$ ,  $g(x, y)$  and  $\epsilon(x, y)$  are modelled as homogeneous and Gaussian random fields

# Wiener Filtering



$$E\{g(t)\} = h(t) * E\{f(t)\} + E\{\epsilon(t)\}$$

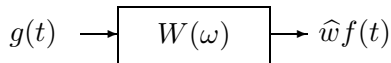
$$R_{gg}(\tau) = h(t) * h(t) * R_{ff}(\tau) + R_{\epsilon\epsilon}(\tau)$$

$$R_{gf}(\tau) = h(t) * R_{ff}(\tau)$$

$$S_{gg}(\omega) = |H(\omega)|^2 S_{ff}(\omega) + R_{\epsilon\epsilon}(\omega)$$

$$S_{gf}(\omega) = H(\omega) S_{ff}(\omega)$$

$$S_{fg}(\omega) = H^*(\omega) S_{ff}(\omega)$$



$$\hat{w}f(t) = w(t) * g(t)$$

# Wiener Filtering

$$EQM = E \{ [f(t) - \hat{w}f(t)]^2 \} = E \{ [f(t) - w(t) * g(t)]^2 \}$$

$$\frac{\partial EQM}{\partial f} = -2E \{ [f(t) - w(t) * g(t)] * g(t + \tau) \} = 0$$

$$E \{ [f(t) - w(t) * g(t)] g(t + \tau) \} = 0 \quad \forall t, \tau \rightarrow$$

$$R_{fg}(\tau) = w(t) * R_{gg}(\tau)$$

$$W(\omega) = \frac{S_{fg}(\omega)}{S_{gg}(\omega)} = \frac{H^*(\omega) S_{ff}(\omega)}{|H(\omega)|^2 S_{ff}(\omega) + S_{\epsilon\epsilon}(\omega)}$$

$$W(\omega) = \frac{H^*(\omega) S_{ff}(\omega)}{|H(\omega)|^2 S_{ff}(\omega) + S_{\epsilon\epsilon}(\omega)} = \frac{1}{H(\omega)} \frac{|H(\omega)|^2}{|H(\omega)|^2 + \frac{S_{\epsilon\epsilon}(\omega)}{S_{ff}(\omega)}}$$

# Wiener Filtering

- ▶ Linear Estimation:  $\hat{w}f(x, y)$  is such that:
  - ▶  $\hat{w}f(x, y)$  depends on  $g(x, y)$  in a linear way:

$$\hat{w}f(x, y) = \iint g(x', y') w(x - x', y - y') dx' dy'$$

$w(x, y)$  is the impulse response of the Wiener filter

- ▶ minimizes MSE:  $E \{|f(x, y) - \hat{w}f(x, y)|^2\}$
- ▶ Orthogonality condition:

$$(f(x, y) - \hat{w}f(x, y)) \perp g(x', y') \quad \longrightarrow \quad E \{(f(x, y) - \hat{w}f(x, y)) g(x', y')\} = 0$$

$$\hat{w}f = g * w \quad \longrightarrow \quad E \{(f(x, y) - g(x, y) * w(x, y)) g(x + \alpha_1, y + \alpha_2)\} = 0$$

$$R_{fg}(\alpha_1, \alpha_2) = (R_{gg} * w)(\alpha_1, \alpha_2) \quad \longrightarrow \quad \text{TF} \quad \longrightarrow \quad S_{fg}(u, v) = S_{gg}(u, v) W(u, v)$$

$\Downarrow$

$$W(u, v) = \frac{S_{fg}(u, v)}{S_{gg}(u, v)}$$

# Wiener filtering

Signal	Image
$W(\omega) = \frac{S_{fg}(\omega)}{S_{gg}(\omega)}$	$W(u, v) = \frac{S_{fg}(u, v)}{S_{gg}(u, v)}$

## Particular Case:

$f(x, y)$  and  $b(x, y)$  are assumed to be centered and non correlated

$$S_{fg}(u, v) = H'(u, v) S_{ff}(u, v)$$

$$S_{gg}(u, v) = |H(u, v)|^2 S_{ff}(u, v) + S_{\epsilon\epsilon}(u, v)$$

$$W(u, v) = \frac{H'(u, v) S_{ff}(u, v)}{|H(u, v)|^2 S_{ff}(u, v) + S_{\epsilon\epsilon}(u, v)}$$

Signal	Image
$W(\omega) = \frac{1}{H(\omega)} \frac{ H(\omega) ^2}{ H(\omega) ^2 + \frac{S_{\epsilon\epsilon}(\omega)}{S_{ff}(\omega)}}$	$W(u, v) = \frac{1}{H(u, v)} \frac{ H(u, v) ^2}{ H(u, v) ^2 + \frac{S_{\epsilon\epsilon}(u, v)}{S_{ff}(u, v)}}$



# Convolution: Discretization for Identification

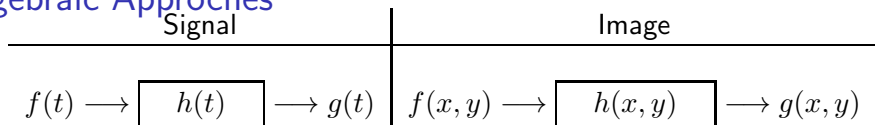
## Causal systems and causal input

$$\begin{bmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{bmatrix} = \begin{bmatrix} 0 & \cdot & 0 & f(0) \\ \cdot & \cdot & f(0) & f(1) \\ \cdot & \cdot & f(0) & f(1) \\ \cdot & \cdot & \cdot & \cdot \\ f(0) & f(1) & \cdot & \cdot \\ f(1) & \cdot & \cdot & f(M-p) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ f(M-p) & \cdot & \cdot & f(M) \end{bmatrix} \begin{bmatrix} h(p) \\ h(p-1) \\ \vdots \\ \vdots \\ h(1) \\ h(0) \end{bmatrix}$$

$$\mathbf{g} = \mathbf{F} \mathbf{h} + \boldsymbol{\epsilon}$$

- ▶  $\mathbf{g}$  is a  $(M + 1)$ -dimensional vector,
- ▶  $\mathbf{F}$  has dimension  $(M + 1) \times (p + 1)$ ,

# Algebraic Approaches



Discretization



$$g = Hf$$

- ▶ **Ideal case:**  $H$  invertible  $\longrightarrow \hat{f} = H^{-1}g$
- ▶  $M > N$  **Least Squares:**

$$g = Hf + \epsilon$$

$$e = \|g - Hf\|^2 = [g - Hf]'[g - Hf]$$

$$\hat{f} = \arg \min_f \{e\}$$

$$\nabla e = -2H'[g - Hf] = 0 \longrightarrow H'Hf = H'g$$

- ▶ If  $H'H$  is invertible  $\hat{f} = (H'H)^{-1}H'g$

# Algebraic Approches: Generalized Inversion

General case of  $[M, N]$  matrix  $\mathbf{H}$ :

- ▶ if  $M = N$  and  $\text{rang}\{\mathbf{H}\} = N$  then  $\mathbf{H}^+ = \mathbf{H}^{-1}$
- ▶ if  $M > N$  and  $\text{rang}\{\mathbf{H}\} = N$  then  $\mathbf{H}^+ = (\mathbf{H}'\mathbf{H})^{-1}\mathbf{H}'$
- ▶ if  $M < N$  and  $\text{rang}\{\mathbf{H}\} = M$  then  $\mathbf{H}^+ = \mathbf{H}'(\mathbf{H}\mathbf{H}')^{-1}$
- ▶ if  $\text{rang}\{\mathbf{H}\} = K < \inf M, N$  then
  - ▶ Singular Value Decomposition (SVD)
  - ▶ Iterative methods
  - ▶ Recursive methods

# Regularization

$$J_\alpha(\mathbf{f}) = [\mathbf{H}\mathbf{f} - \mathbf{g}]'[\mathbf{H}\mathbf{f} - \mathbf{g}] + \alpha[\mathbf{D}\mathbf{f}]'[\mathbf{D}\mathbf{f}] = \|\mathbf{H}\mathbf{f} - \mathbf{g}\|^2 + \alpha\|\mathbf{D}\mathbf{f}\|^2$$

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ -1 & 1 & \ddots & & \vdots \\ 0 & -1 & 1 & \ddots & \vdots \\ & 0 & -1 & 1 & \ddots \\ 0 & & & 0 & -1 & 1 \end{bmatrix} \text{ or } \mathbf{D} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ -2 & 1 & \ddots & & \vdots \\ 1 & -2 & 1 & \ddots & \vdots \\ & 1 & -2 & 1 & \ddots \\ 0 & & & 1 & -2 & 1 \end{bmatrix}$$

$$\nabla J_\alpha = 2\mathbf{H}'[\mathbf{H}\mathbf{f} - \mathbf{g}]' + 2\alpha\mathbf{D}'\mathbf{D}\mathbf{f} = 0$$

$$[\mathbf{H}'\mathbf{H} + \alpha\mathbf{D}'\mathbf{D}]\hat{\mathbf{f}} = \mathbf{H}'\mathbf{g} \longrightarrow \hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \alpha\mathbf{D}'\mathbf{D}]^{-1}\mathbf{H}'\mathbf{g}$$

# Regularization Algorithms

$$\begin{aligned} & \text{minimize } J(\mathbf{f}) = Q(\mathbf{f}) + \lambda\Omega(\mathbf{f}) \\ \text{with } & Q(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 = [\mathbf{g} - \mathbf{H}\mathbf{f}]'[\mathbf{g} - \mathbf{H}\mathbf{f}] \\ & = \\ & \text{minimize } \Omega(\mathbf{f}) \text{ subj. to the constraint} \\ & e = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 = [\mathbf{g} - \mathbf{H}\mathbf{f}]'[\mathbf{g} - \mathbf{H}\mathbf{f}] < \epsilon \end{aligned}$$

## A priori Information:

- ▶ Smoothness

$$\begin{aligned} \Omega(\mathbf{f}) &= [\mathbf{D}\mathbf{f}]'[\mathbf{D}\mathbf{f}] = \|\mathbf{D}\mathbf{f}\|^2 \\ \hat{\mathbf{f}} &= [\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}]^{-1}\mathbf{H}'\mathbf{g} \end{aligned}$$

- ▶ Positivity:

$\Omega(\mathbf{f}) =$  a nonquadratique function of  $\mathbf{f}$

No explicite solution

## Regularization Algorithms: 3 main approaches

$$\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}]^{-1}\mathbf{H}'\mathbf{g}$$

Computation of  $\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}]^{-1}\mathbf{H}'\mathbf{g}$

- ▶ Circulant matrix approximation:  
when  $\mathbf{H}$  and  $\mathbf{D}$  are Toeplitz, they can be approximated by the circulant matrices

- ▶ Iterative methods:

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f}} \{ \|J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\mathbf{D}\mathbf{f}\|^2 \}$$

- ▶ Recursive methods:  
 $\hat{\mathbf{f}}$  at iteration  $k$  is computed as a function of  $\hat{\mathbf{f}}$  at previous iteration with one less data.

# Regularization algorithms: Circulant approximation

## 1D Deconvolution:

$$\mathbf{g} = \mathbf{H} \mathbf{f} + \epsilon$$

## $\mathbf{H}$ Toeplitz matrix

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f}} \{ \mathbf{f} \} J(\mathbf{f}) = Q(\mathbf{f}) + \lambda \Omega(\mathbf{f})$$

$$Q(\mathbf{f}) = \|\mathbf{g} - \mathbf{H} \mathbf{f}\|^2 = [\mathbf{g} - \mathbf{H} \mathbf{f}]' [\mathbf{g} - \mathbf{H} \mathbf{f}] \text{ and } \Omega(\mathbf{f}) = \|\mathbf{C} \mathbf{f}\|^2 = [\mathbf{C} \mathbf{f}]' [\mathbf{C} \mathbf{f}]$$

$\mathbf{C}$  a convolution matrix with the following impulse response

$$\mathbf{h}_1 = [1, -2, 1] \quad \longrightarrow \quad x(i) = x(i+1) - 2x(i) + x(i-1)$$

$$\Omega(\mathbf{f}) = \sum_{j=1}^N (x(i+1) - 2x(i) + x(i-1))^2 = \|\mathbf{C} \mathbf{f}\|^2 = \mathbf{f}' \mathbf{C}' \mathbf{C} \mathbf{f}$$

## Solution :

$$\hat{\mathbf{f}} = [\mathbf{H}' \mathbf{H} + \lambda \mathbf{C}' \mathbf{C}]^{-1} \mathbf{H}' \mathbf{g}$$

## Regularization algorithms: Circulant approximation

Main Idea : expand the vectors  $\mathbf{f}$ ,  $\mathbf{h}$  and  $\mathbf{g}$  by the zeros to obtain  $\mathbf{g}_e = \mathbf{H}_e \mathbf{f}_e$  with  $\mathbf{H}_e$  a circulant matrix

$$\mathbf{f}_e(i) = \begin{cases} x(i) & i = 1, \dots, Nx \\ 0 & i = Nx + 1, \dots, P \geq Nx + Nh - 1 \end{cases}$$

$$y_e(i) = \begin{cases} y(i) & i = 1, \dots, Ny \\ 0 & i = Ny + 1, \dots, P \end{cases}$$

$$h_e(i) = \begin{cases} h(i) & i = 1, \dots, Nh \\ 0 & i = Nh + 1, \dots, P \end{cases}$$

$$y_e(k) = \sum_{i=0}^{Nh-1} x_e(k-i)h_e(i) \quad \longrightarrow \quad \mathbf{g}_e = \mathbf{H}_e \mathbf{f}_e$$

with  $\mathbf{H}_e$  a circulant matrix which can be diagonalized by FFT



## Regularization algorithms: Circulant approximation

$$\mathbf{H}_e = \mathbf{F}\mathbf{\Lambda}\mathbf{F}^{-1} \text{ with } \mathbf{F}[k, l] = \exp\left\{j2\pi\frac{kl}{P}\right\} \quad \mathbf{F}^{-1}[k, l] = \frac{1}{P} \exp\left\{-j2\pi\frac{kl}{P}\right\}$$

$$\mathbf{\Lambda} = \text{diag}[\lambda_1, \dots, \lambda_P] \text{ and } [\lambda_1, \dots, \lambda_P] = \text{TFD} [h_1, \dots, h_{Nh}, 0, \dots, 0]$$

$$c = [1, -2, 1] \quad c_e(i) = \begin{cases} c(i) & i = 1, \dots, 3 \\ 0 & i = 4, \dots, P \end{cases}$$

$$\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda\mathbf{C}'\mathbf{C}]^{-1}\mathbf{H}'\mathbf{g} \longrightarrow \mathbf{F}\hat{\mathbf{f}}_e = [\mathbf{\Lambda}'_h\mathbf{\Lambda}_h + \lambda\mathbf{\Lambda}'_c\mathbf{\Lambda}_c]^{-1}\mathbf{\Lambda}'_h\mathbf{F}\mathbf{g}$$

$$\text{TFD} \{\mathbf{f}_e\} = [\mathbf{\Lambda}'_h\mathbf{\Lambda}_h + \lambda\mathbf{\Lambda}'_c\mathbf{\Lambda}_c]^{-1}\mathbf{\Lambda}'_h \text{TFD} \{\mathbf{g}\}$$

$$\hat{\mathbf{f}}(\omega) = \frac{1}{H(\omega)} \frac{|H(\omega)|^2}{|H(\omega)|^2 + \lambda|C(\omega)|^2} y(\omega)$$

**Link with Wiener filter:**  $C(\omega) = N(\omega)/X(\omega)$

# Image Restoration

$C$  Convolution matrix with the following impulse response:

$$H_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\Omega(\mathbf{f}) = \sum \sum (f(i+1, j) + f(i-1, j) + f(i+1, j+1) + f(i-1, j+1) - 4f(i, j))^2$$

$$x_e(k, l) = \begin{cases} x(k, l) & k = 1, \dots, K \quad l = 1, \dots, L \\ 0 & k = K+1, \dots, P \quad l = L+1, \dots, P \end{cases}$$

# Regularization: Iterative methods: Gradient based

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f}} \{J(\mathbf{f}) = Q(\mathbf{f}) + \lambda\Omega(\mathbf{f})\}$$

Let note :  $\mathbf{g}^k = \nabla J(\mathbf{f}^k)$  gradient,  $\mathbf{H}^k = \nabla^2 J(\mathbf{f}^k)$  Hessian.

## First order gradient methods

- ▶ fixed step:

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha \mathbf{g}^{(k)} \quad \alpha \text{ fixe}$$

- ▶ Optimal or steepest descent step:

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha^{(k)} \mathbf{g}^{(k)}$$

$$\alpha^{(k)} = -\frac{\mathbf{g}^{(k)t} \mathbf{g}^{(k)}}{\mathbf{g}^{(k)t} \mathbf{H}^k \mathbf{g}^{(k)}} = \frac{\|\mathbf{g}^k\|^2}{\|\mathbf{g}^k\|_H^2}$$

# Regularization: Iterative methods: Conjugate Gradient

## ► Conjugate Gradient (CG)

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)} \quad \alpha^{(k)} = -\frac{\mathbf{d}^{(k)t} \mathbf{g}^{(k)}}{\mathbf{d}^{(k)t} \mathbf{H}^k \mathbf{d}^{(k)}}$$

$$\mathbf{d}^{(k+1)} = \mathbf{d}^{(k)} + \beta^{(k)} \mathbf{g}^{(k)} \quad \beta^{(k)} = -\frac{\mathbf{g}^{(k)t} \mathbf{g}^{(k)}}{\mathbf{g}^{(k-1)t} \mathbf{g}^{(k-1)}}$$

## ► Newton method

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + (\mathbf{H}^{(k)})^{-1} \mathbf{g}^{(k)}$$

- Advantages :  $\Omega(\mathbf{f})$  can be any convex function
- Limitations : Computational cost

## Regularization: Recursive algorithms

$$\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}]^{-1}\mathbf{H}'\mathbf{g}$$

**Main idea:** Express  $\mathbf{f}_{i+1}$  as a function of  $\mathbf{f}_i$

$$\mathbf{f}_{i+1} = (\mathbf{H}'_{i+1}\mathbf{H}_{i+1} + \alpha\mathbf{D})^{-1}\mathbf{H}'_{i+1}\mathbf{g}_{i+1}$$

$$\mathbf{f}_i = (\mathbf{H}_i^t\mathbf{H}_i + \alpha\mathbf{D})^{-1}\mathbf{H}_i^t\mathbf{g}_i$$

$\Downarrow$

$$\mathbf{f}_{i+1} = (\mathbf{H}_i^t\mathbf{H}_i + \mathbf{h}_{i+1}\mathbf{h}'_{i+1} + \alpha\mathbf{D})^{-1}(\mathbf{H}_i^t\mathbf{g}_i - \mathbf{h}_{i+1}\mathbf{g}_i + 1)$$

**Noting:**

$$\mathbf{P}_i = (\mathbf{H}_i^t\mathbf{H}_i + \alpha\mathbf{D})^{-1} \quad \text{and} \quad \mathbf{P}'_{i+1} = \mathbf{P}'_i + \mathbf{h}_{i+1}\mathbf{h}'_{i+1}$$

$\Downarrow$

$$\mathbf{f}_{i+1} = \mathbf{f}_i + \mathbf{P}_{i+1}\mathbf{h}_{i+1}(\mathbf{g}_{i+1} - \mathbf{h}'_{i+1}\mathbf{f}_i)$$

$$\mathbf{P}_{i+1} = \mathbf{P}_i - \mathbf{P}_i\mathbf{h}_{i+1}(\mathbf{h}'_{i+1}\mathbf{P}_i\mathbf{H}_{i+1} + \alpha)^{-1}\mathbf{h}'_{i+1}\mathbf{P}_i$$

# Identification and Deconvolution

Deconvolution

$$g = H f + \epsilon$$

$$J(f) = \|g - H f\|^2 + \lambda_f \|D_f f\|^2$$

$$\nabla J(f) = -2H'(g - H f) + 2\lambda_f D_f' D_f f$$

$$\hat{f} = [H'H + \lambda_f D_f' D_f]^{-1} H' g$$

$$\hat{f}(\omega) = \frac{H^*(\omega)}{|H(\omega)|^2 + \lambda_f |D_f(\omega)|^2} g(\omega)$$

$$\hat{f}(\omega) = \frac{H^*(\omega)}{|H(\omega)|^2 + \frac{S_{\epsilon\epsilon}(\omega)}{S_{ff}(\omega)}} g(\omega)$$

$$p(g|f) = \mathcal{N}(H f, \Sigma_\epsilon)$$

$$p(f) = \mathcal{N}(0, \Sigma_f)$$

$$p(f|g) = \mathcal{N}(\hat{f}, \hat{\Sigma}_f)$$

$$\hat{\Sigma}_f = [H'H + \lambda_f D_f' D_f]^{-1}$$

$$\hat{f} = [H'H + \lambda_f D_f' D_f]^{-1} H' g$$

Identification

$$g = F h + \epsilon$$

$$J(h) = \|g - F h\|^2 + \lambda_h \|D_h h\|^2$$

$$\nabla J(h) = -2F'(g - F h) + 2\lambda_h D_h' h$$

$$\hat{h} = [F'F + \lambda_h D_h' D_h]^{-1} F' g$$

$$\hat{h}(\omega) = \frac{F^*(\omega)}{|F(\omega)|^2 + \lambda_h |D_h(\omega)|^2} g(\omega)$$

$$\hat{h}(\omega) = \frac{F^*(\omega)}{|F(\omega)|^2 + \frac{S_{\epsilon\epsilon}(\omega)}{S_{hh}(\omega)}} g(\omega)$$

$$p(g|h) = \mathcal{N}(F h, \Sigma_\epsilon)$$

$$p(h) = \mathcal{N}(0, \Sigma_h)$$

$$p(h|g) = \mathcal{N}(\hat{h}, \hat{\Sigma}_h)$$

$$\hat{\Sigma}_h = [F'F + \lambda_h D_h' D_h]^{-1}$$

$$\hat{h} = [F'F + \lambda_h D_h' D_h]^{-1} F' g$$

# Blind Deconvolution: Regularization

Deconvolution

$$\mathbf{g} = \mathbf{H} \mathbf{f} + \epsilon$$

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H} \mathbf{f}\|^2 + \lambda_f \|\mathbf{D}_f \mathbf{f}\|^2$$

- ▶ Joint Criterion

$$J(\mathbf{f}, \mathbf{h}) = \|\mathbf{g} - \mathbf{H} \mathbf{f}\|^2 + \lambda_f \|\mathbf{D}_f \mathbf{f}\|^2 + \lambda_h \|\mathbf{D}_h \mathbf{h}\|^2$$

- ▶ iterative algorithm

Identification

$$\mathbf{g} = \mathbf{F} \mathbf{h} + \epsilon$$

$$J(\mathbf{h}) = \|\mathbf{g} - \mathbf{F} \mathbf{h}\|^2 + \lambda_h \|\mathbf{D}_h \mathbf{h}\|^2$$

Deconvolution

$$\nabla_{\mathbf{f}} J(\mathbf{f}, \mathbf{h}) = -2\mathbf{H}'(\mathbf{g} - \mathbf{H} \mathbf{f}) + 2\lambda_f \mathbf{D}_f' \mathbf{D}_f \mathbf{f}$$

$$\hat{\mathbf{f}} = [\mathbf{H}' \mathbf{H} + \lambda_f \mathbf{D}_f' \mathbf{D}_f]^{-1} \mathbf{H}' \mathbf{g}$$

$$\hat{f}(\omega) = \frac{1}{H(\omega)} \frac{|H(\omega)|^2}{|H(\omega)|^2 + \lambda_f |D_f(\omega)|^2} g(\omega)$$

Identification

$$\nabla_{\mathbf{h}} J(\mathbf{f}, \mathbf{h}) = -2\mathbf{F}'(\mathbf{g} - \mathbf{F} \mathbf{h}) + 2\lambda_h \mathbf{D}_h' \mathbf{D}_h \mathbf{h}$$

$$\hat{\mathbf{h}} = [\mathbf{F}' \mathbf{F} + \lambda_h \mathbf{D}_h' \mathbf{D}_h]^{-1} \mathbf{F}' \mathbf{g}$$

$$\hat{h}(\omega) = \frac{1}{F(\omega)} \frac{|F(\omega)|^2}{|F(\omega)|^2 + \lambda_h |D_h(\omega)|^2} g(\omega)$$

# Blind Deconvolution: Bayesian approach

Deconvolution	Identification
$\mathbf{g} = \mathbf{H} \mathbf{f} + \epsilon$	$\mathbf{g} = \mathbf{F} \mathbf{h} + \epsilon$
$p(\mathbf{g} \mathbf{f}) = \mathcal{N}(\mathbf{H} \mathbf{f}, \Sigma_\epsilon)$	$p(\mathbf{g} \mathbf{h}) = \mathcal{N}(\mathbf{F} \mathbf{h}, \Sigma_\epsilon)$
$p(\mathbf{f}) = \mathcal{N}(0, \Sigma_f)$	$p(\mathbf{h}) = \mathcal{N}(0, \Sigma_h)$
$p(\mathbf{f} \mathbf{g}) = \mathcal{N}(\hat{\mathbf{f}}, \hat{\Sigma}_f)$	$p(\mathbf{h} \mathbf{g}) = \mathcal{N}(\hat{\mathbf{h}}, \hat{\Sigma}_h)$
$\hat{\Sigma}_f = [\mathbf{H}'\mathbf{H} + \lambda_f \mathbf{D}'_f \mathbf{D}_f]^{-1}$	$\hat{\Sigma}_h = [\mathbf{F}'\mathbf{F} + \lambda_h \mathbf{D}'_h \mathbf{D}_h]^{-1}$
$\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda_f \mathbf{D}'_f \mathbf{D}_f]^{-1} \mathbf{H}'\mathbf{g}$	$\hat{\mathbf{h}} = [\mathbf{F}'\mathbf{F} + \lambda_h \mathbf{D}'_h \mathbf{D}_h]^{-1} \mathbf{F}'\mathbf{g}$

► Joint posterior law:

$$p(\mathbf{f}, \mathbf{h}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{h}) p(\mathbf{f}) p(\mathbf{h})$$

$$p(\mathbf{f}, \mathbf{h}|\mathbf{g}) \propto \exp \{-J(\mathbf{f}, \mathbf{h})\}$$

with

$$J(\mathbf{f}, \mathbf{h}) = \|\mathbf{g} - \mathbf{H} \mathbf{f}\|^2 + \lambda_f \|\mathbf{D}_f \mathbf{f}\|^2 + \lambda_h \|\mathbf{D}_h \mathbf{h}\|^2$$

► iterative algorithm



# Blind Deconvolution: Bayesian Joint MAP criterion

- ▶ Joint posterior law:

$$p(\mathbf{f}, \mathbf{h} | \mathbf{g}) \propto p(\mathbf{g} | \mathbf{f}, \mathbf{h}) p(\mathbf{f}) p(\mathbf{h})$$

$$p(\mathbf{f}, \mathbf{h} | \mathbf{g}) \propto \exp \{-J(\mathbf{f}, \mathbf{h})\}$$

with

$$J(\mathbf{f}, \mathbf{h}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda_f \|\mathbf{D}_f \mathbf{f}\|^2 + \lambda_h \|\mathbf{D}_h \mathbf{h}\|^2$$

- ▶ iterative algorithm

Deconvolution	Identification
$p(\mathbf{g}   \mathbf{f}, \mathbf{H}) = \mathcal{N}(\mathbf{H}\mathbf{f}, \Sigma_\epsilon)$	$p(\mathbf{g}   \mathbf{h}, \mathbf{F}) = \mathcal{N}(\mathbf{F}\mathbf{h}, \Sigma_\epsilon)$
$p(\mathbf{f}) = \mathcal{N}(0, \Sigma_f)$	$p(\mathbf{h}) = \mathcal{N}(0, \Sigma_h)$
$p(\mathbf{f}   \mathbf{g}, \mathbf{H}) = \mathcal{N}(\hat{\mathbf{f}}, \hat{\Sigma}_f)$	$p(\mathbf{h}   \mathbf{g}, \mathbf{F}) = \mathcal{N}(\hat{\mathbf{h}}, \hat{\Sigma}_h)$
$\hat{\Sigma}_f = [\mathbf{H}'\mathbf{H} + \lambda_f \mathbf{D}'_f \mathbf{D}_f]^{-1}$	$\hat{\Sigma}_h = [\mathbf{F}'\mathbf{F} + \lambda_h \mathbf{D}'_h \mathbf{D}_h]^{-1}$
$\hat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda_f \mathbf{D}'_f \mathbf{D}_f]^{-1} \mathbf{H}'\mathbf{g}$	$\hat{\mathbf{h}} = [\mathbf{F}'\mathbf{F} + \lambda_h \mathbf{D}'_h \mathbf{D}_h]^{-1} \mathbf{F}'\mathbf{g}$

# Blind Deconvolution: Marginalization and EM algorithm

- ▶ Joint posterior law:

- ▶ Marginalization  $p(\mathbf{f}, \mathbf{h}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{h}) p(\mathbf{f}) p(\mathbf{h})$

$$p(\mathbf{h}|\mathbf{g}) = \int p(\mathbf{f}, \mathbf{h}|\mathbf{g}) d\mathbf{f}$$

$$\hat{\mathbf{h}} = \arg \max_{\mathbf{h}} \{p(\mathbf{h}|\mathbf{g})\} \longrightarrow \hat{\mathbf{f}} = \arg \max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g}, \hat{\mathbf{h}})\}$$

- ▶ Expression of  $p(\mathbf{h}|\mathbf{g})$  and its maximization are complexes
- ▶ Expectation-Maximization Algorithm

$$\ln p(\mathbf{f}, \mathbf{h}|\mathbf{g}) \propto J(\mathbf{f}, \mathbf{h}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda_f \|\mathbf{D}_f \mathbf{f}\|^2 + \lambda_h \|\mathbf{D}_h \mathbf{h}\|^2$$

- ▶ Iterative algorithm
- ▶ Expectation: Compute

$$Q(\mathbf{h}, \mathbf{h}^{k-1}) = \mathbb{E}_{p(\mathbf{f}, \mathbf{h}^{k-1}|\mathbf{g})} \{J(\mathbf{f}, \mathbf{h})\} = \langle \ln p(\mathbf{f}, \mathbf{h}|\mathbf{g}) \rangle_{p(\mathbf{f}, \mathbf{h}^{k-1}|\mathbf{g})}$$

- ▶ Maximization:

$$\mathbf{h}^k = \arg \max_{\mathbf{h}} \{Q(\mathbf{h}, \mathbf{h}^{k-1})\}$$

# Blind Deconvolution: Variational Bayesian Approximation algorithm

- ▶ Joint posterior law:

$$p(\mathbf{f}, \mathbf{h} | \mathbf{g}) \propto p(\mathbf{g} | \mathbf{f}, \mathbf{h}) p(\mathbf{f}) p(\mathbf{h})$$

- ▶ Approximation:  $p(\mathbf{f}, \mathbf{h} | \mathbf{g})$  by  $q(\mathbf{f}, \mathbf{h} | \mathbf{g}) = q_1(\mathbf{f}) q_2(\mathbf{h})$
- ▶ Criterion of approximation: Kullback-Leiler

$$\text{KL}(q|p) = \int q \ln \frac{q}{p} = \int q_1 q_2 \ln \frac{q_1 q_2}{p}$$

$$\begin{aligned} \text{KL}(q_1 q_2 | p) &= \int q_1 \ln q_1 + \int q_2 \ln q_2 - \int q \ln p \\ &= -\mathcal{H}(q_1) - \mathcal{H}(q_2) + \langle -\ln p(\mathbf{f}, \mathbf{h} | \mathbf{g}) \rangle_q \end{aligned}$$

- ▶ When the expression of  $q_1$  and  $q_2$  are obtained, use them.

# Variational Bayesian Approximation algorithm

- ▶ Kullback-Leibler criterion

$$\begin{aligned}\text{KL}(q_1 q_2|p) &= \int q_1 \ln q_1 + \int q_2 \ln q_2 + \int q \ln p \\ &= -\mathcal{H}(q_1) - \mathcal{H}(q_2) + \langle -\ln p((\mathbf{f}, \mathbf{h}|\mathbf{g})) \rangle_q\end{aligned}$$

- ▶ Free energy

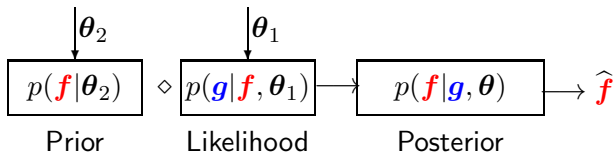
$$\mathcal{F}(q_1 q_2) = - \langle \ln p((\mathbf{f}, \mathbf{h}|\mathbf{g})) \rangle_{q_1 q_2}$$

- ▶ Equivalence between optimization of  $\text{KL}(q_1 q_2|p)$  and  $\mathcal{F}(q_1 q_2)$
- ▶ Alternate optimization:

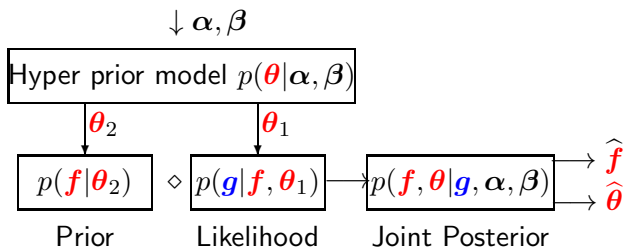
$$\begin{aligned}\hat{q}_1 &= \arg \min_{q_1} \{\text{KL}(q_1 q_2|p)\} = \arg \min_{q_1} \{\mathcal{F}(q_1 q_2)\} \\ \hat{q}_2 &= \arg \min_{q_2} \{\text{KL}(q_1 q_2|p)\} = \arg \min_{q_2} \{\mathcal{F}(q_1 q_2)\}\end{aligned}$$

# Summary of Bayesian estimation for Deconvolution

- Simple Bayesian Model and Estimation for Deconvolution

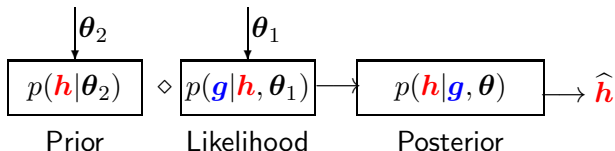


- Full Bayesian Model and Hyperparameter Estimation for Deconvolution

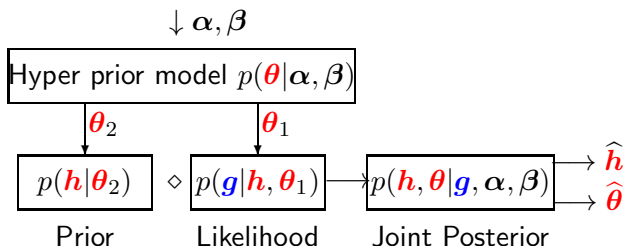


# Summary of Bayesian estimation for Identification

- ▶ Simple Bayesian Model and Estimation for Identification

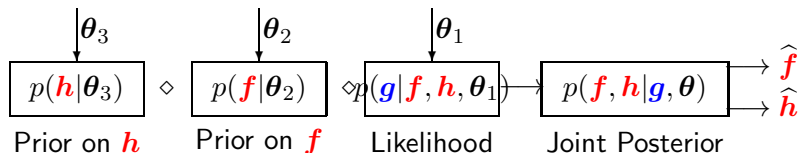


- ▶ Full Bayesian Model and Hyperparameter Estimation for Identification



# Summary of Bayesian estimation for Blind Deconvolution

## Known hyperparameters $\theta$



## Unknown hyperparameters $\theta$

