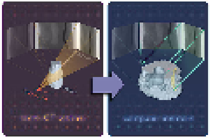
Bayesian Inference and Algorithms for Large Scale Computed Tomography

Ali Mohammad-Djafari
Laboratoire des Signaux et Systèmes (L2S)
UMR8506 CNRS-CentraleSupélec-UNIV PARIS SUD
SUPELEC, 91192 Gif-sur-Yvette, France
http://lss.centralesupelec.fr
Email: djafari@lss.supelec.fr
http://djafari.free.fr
http://publicationslist.org/djafari

Eusipco 2017

Contents

Seeing inside of a body: Computed Tomography



- $f(x,y)$ a section of a real 3D body $f(x,y,z)$
- $g_\phi(r)$ a line of observed radiographe $g_\phi(r,z)$

- Forward model:
  Line integrals or Radon Transform

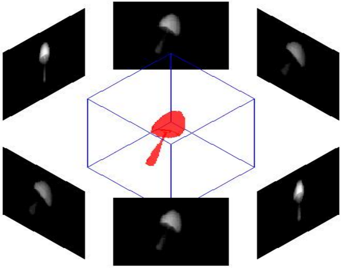
$$
\begin{aligned}
g_\phi(r) &= \int_{L_{r,\phi}} f(x,y)\,\mathrm{d}l + \epsilon_\phi(r) \\
&= \iint f(x,y)\,\delta(r - x\cos\phi - y\sin\phi)\,\mathrm{d}x\,\mathrm{d}y + \epsilon_\phi(r)
\end{aligned}
$$

- Inverse problem: Image reconstruction

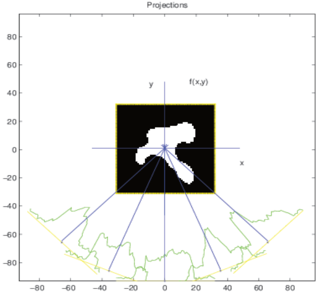
  Given the forward model $\mathcal{H}$ (Radon Transform) and
  a set of data $g_{\phi_i}(r), i = 1, \cdots, M$
  find $f(x,y)$

2D and 3D Computed Tomography

3D



2D



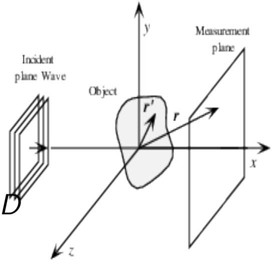
$$g_\phi(r_1,r_2)=\int_{\mathcal{L}_{r_1,r_2,\phi}} f(x,y,z)\,\mathrm{d}l \qquad g_\phi(r)=\int_{\mathcal{L}_{r,\phi}} f(x,y)\,\mathrm{d}l$$

Forward probelm: $f(x,y)$ or $f(x,y,z) \longrightarrow g_\phi(r)$ or $g_\phi(r_1,r_2)$

Inverse problem: $g_\phi(r)$ or $g_\phi(r_1,r_2) \longrightarrow f(x,y)$ or $f(x,y,z)$

Microwave or ultrasound imaging

Mesaurs: diffracted wave by the object $\phi_d(\mathbf{r}_i)$
Unknown quantity: $f(\mathbf{r}) = k_0^2(n^2(\mathbf{r}) - 1)$
Intermediate quantity : $\phi(\mathbf{r})$

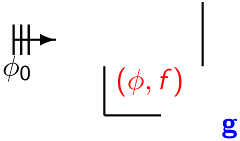


$$\phi_d(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \ \mathbf{r}_i \in S$$

$$\phi(\mathbf{r}) = \phi_0(\mathbf{r}) + \iint_D G_o(\mathbf{r}, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \ \mathbf{r} \in D$$

Born approximation $(\phi(\mathbf{r}') \simeq \phi_0(\mathbf{r}'))$ **):**

$$\phi_d(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')\phi_0(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \ \mathbf{r}_i \in S$$



Discretization :

$$\begin{cases} \phi_d = \mathbf{G}_m \mathbf{F} \phi \\ \phi = \phi_0 + \mathbf{G}_o \mathbf{F} \phi \end{cases} \longrightarrow \begin{cases} \phi_d = \mathbf{H}(\mathbf{f}) \\ \text{with} \quad \mathbf{F} = \mathrm{diag}(\mathbf{f}) \\ \mathbf{H}(\mathbf{f}) = \mathbf{G}_m \mathbf{F}(\mathbf{I} - \mathbf{G}_o \mathbf{F})^{-1} \phi_0 \end{cases}$$

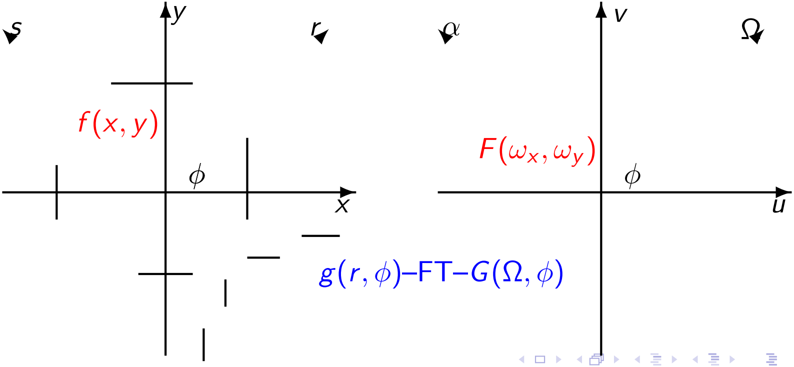
Fourier Synthesis in X ray Tomography

$$g(r,\phi) = \iint f(x,y)\,\delta(r - x\cos\phi - y\sin\phi)\,dx\,dy$$

$$G(\Omega,\phi) = \int g(r,\phi)\exp\left[-j\Omega r\right]\,dr$$

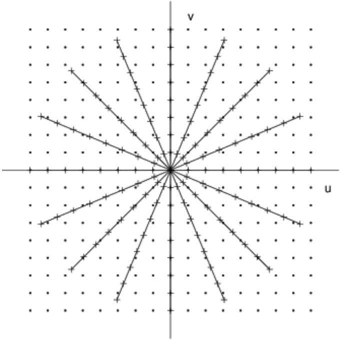
$$F(\omega_x,\omega_y) = \iint f(x,y)\exp\left[-j\omega_x x,\omega_y y\right]\,dx\,dy$$

$$F(\omega_x,\omega_y) = P(\Omega,\phi) \qquad \text{for} \quad \omega_x = \Omega\cos\phi \quad \text{and} \quad \omega_y = \Omega\sin\phi$$

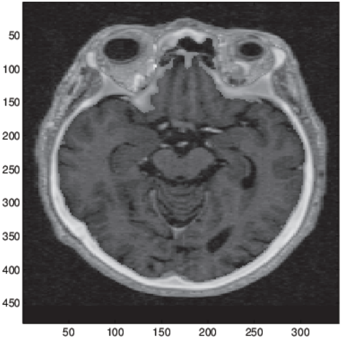


Fourier Synthesis in X ray tomography

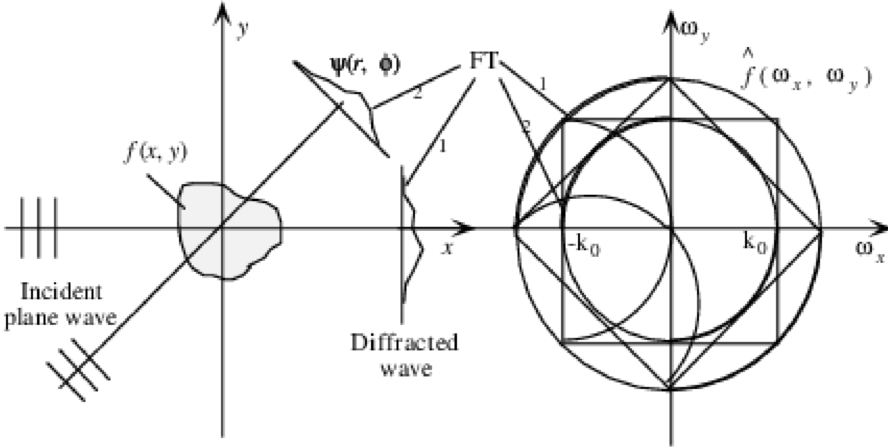
$$F(\omega_x, \omega_y) = \iint f(x, y) \exp\left[-j\omega_x x, \omega_y y\right] \,\mathrm{d}x \,\mathrm{d}y$$



? $\Longrightarrow$

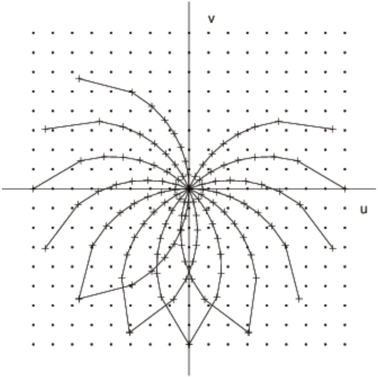


Fourier Synthesis in Diffraction tomography

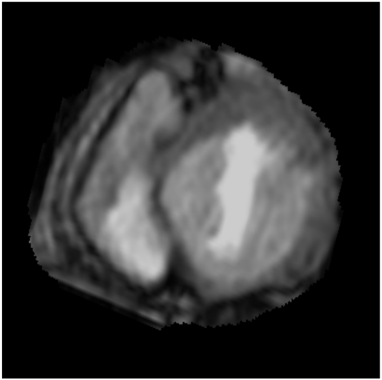


Fourier Synthesis in Diffraction tomography

$$F(\omega_x, \omega_y) = \iint f(x, y) \exp\left[-j\omega_x x, \omega_y y\right] \, \mathrm{d}x \, \mathrm{d}y$$

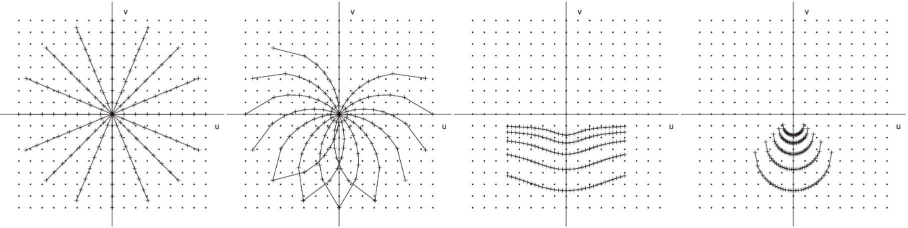


? $\Longrightarrow$



Fourier Synthesis in different imaging systems

$$F(\omega_x, \omega_y) = \iint f(x,y) \exp\left[-j\omega_x x, \omega_y y\right] \, dx \, dy$$



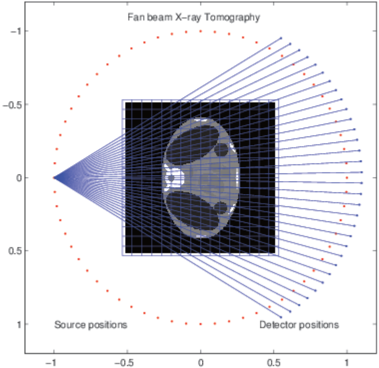
X ray Tomography | Diffraction | Eddy current | SAR & Radar

Invers Problems: other examples and applications

- X ray, Gamma ray Computed Tomography (CT)
- Microwave and ultrasound tomography
- Positron emission tomography (PET)
- Magnetic resonance imaging (MRI)
- Photoacoustic imaging
- Radio astronomy
- Geophysical imaging
- Non Destructive Evaluation (NDE) and Testing (NDT) techniques in industry
- Hyperspectral imaging
- Earth observation methods (Radar, SAR, IR, ...)
- Survey and tracking in security systems

Computed tomography (CT)

A Multislice CT Scanner

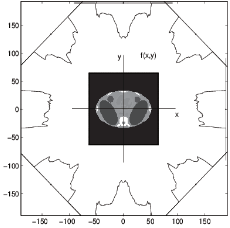


$$g(s_i) = \int_{L_i} f(\mathbf{r}) \, \mathrm{d}l_i + \epsilon(s_i)$$

Discretization

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

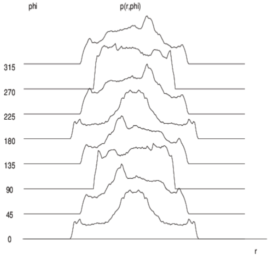
X ray Tomography



$$g(r,\phi) = -\ln\left(\frac{I}{I_0}\right) = \int_{L_{r,\phi}} f(x,y)\, \mathrm{d}l$$

$$g(r,\phi) = \iint_D f(x,y)\, \delta(r - x\cos\phi - y\sin\phi)\, \mathrm{d}x\, \mathrm{d}y$$

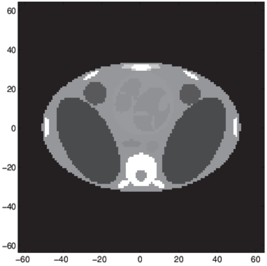
$f(x,y)$ → RT → $g(r,\phi)$



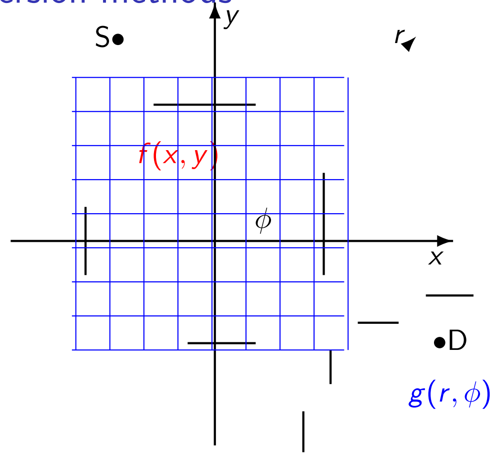
IRT

?

$\Longrightarrow$



Analytical Inversion methods



Radon:

$$g(r,\phi) = \iint_D f(x,y)\,\delta(r - x\cos\phi - y\sin\phi)\,\mathrm{d}x\,\mathrm{d}y$$

$$f(x,y) = \left(-\frac{1}{2\pi^2}\right)\int_0^{\pi}\int_{-\infty}^{+\infty} \frac{\frac{\partial}{\partial r} g(r,\phi)}{(r - x\cos\phi - y\sin\phi)}\,\mathrm{d}r\,\mathrm{d}\phi$$

Filtered Backprojection method

$$f(x,y) = \left(-\frac{1}{2\pi^2}\right) \int_0^{\pi} \int_{-\infty}^{+\infty} \frac{\frac{\partial}{\partial r} g(r,\phi)}{(r - x\cos\phi - y\sin\phi)} \, \mathrm{d}r \, \mathrm{d}\phi$$

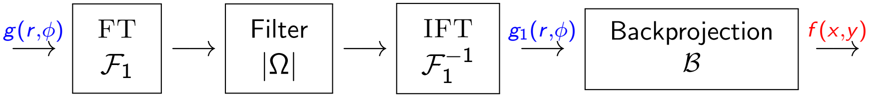
Derivation $\mathcal{D}$: $\quad \bar{g}(r,\phi) = \dfrac{\partial g(r,\phi)}{\partial r}$

Hilbert Transform $\mathcal{H}$: $\quad g_1(r',\phi) = \dfrac{1}{\pi} \displaystyle\int_0^{\infty} \frac{\bar{g}(r,\phi)}{(r - r')} \, \mathrm{d}r$

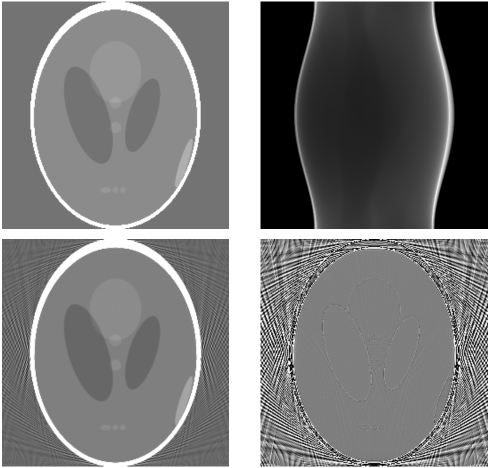
Backprojection $\mathcal{B}$: $\quad f(x,y) = \dfrac{1}{2\pi} \displaystyle\int_0^{\pi} g_1(r' = x\cos\phi + y\sin\phi, \phi) \, \mathrm{d}\phi$

$$f(x,y) = \mathcal{B}\,\mathcal{H}\,\mathcal{D}\, g(r,\phi) = \mathcal{B}\, \mathcal{F}_1^{-1}\, |\Omega|\, \mathcal{F}_1\, g(r,\phi)$$

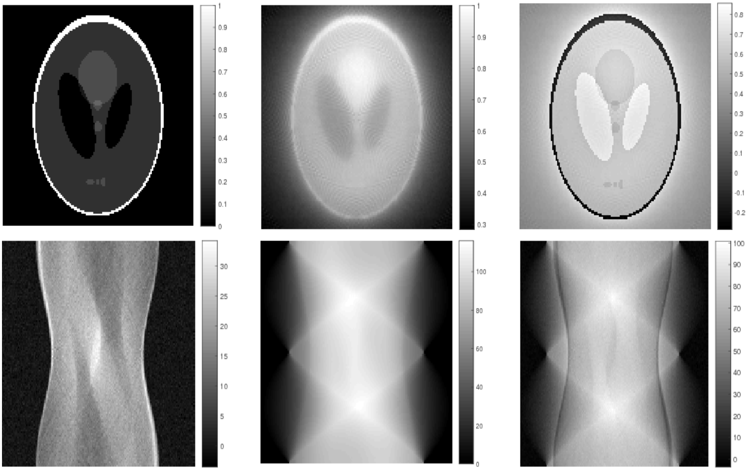
- Backprojection of filtered projections:



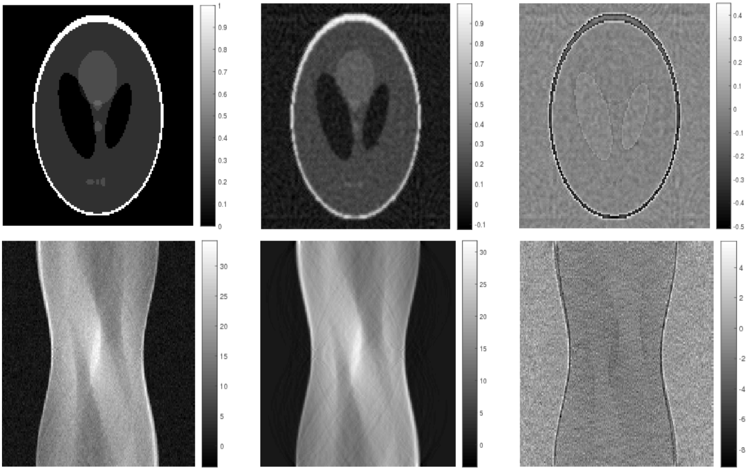
Examples of reconstruction by FBP using CTsim



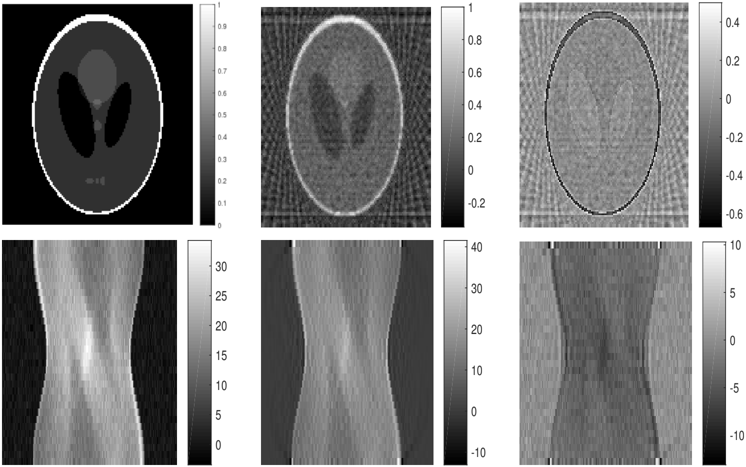
Examples of reconstruction by BP (128 projections)



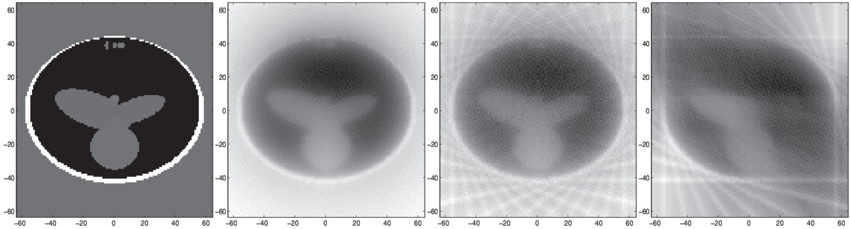
Examples of reconstruction by FBP (128 projections)



Examples of reconstruction by FBP (32 projections)



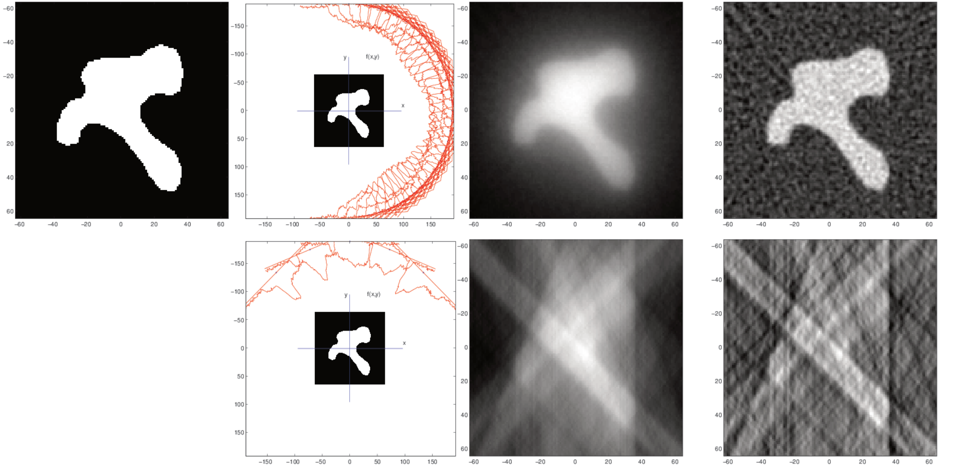
Limitations : Limited angle or noisy data



Original   64 proj.   16 proj.   8 proj. $[0, \pi/2]$

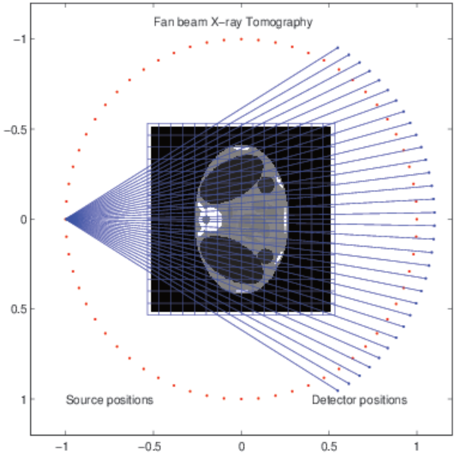
- Limited angle or noisy data
- Accounting for detector size
- Other measurement geometries: fan beam, ...

Limitations : Limited angle or noisy data



Original Data Backprojection Filtered Backprojection

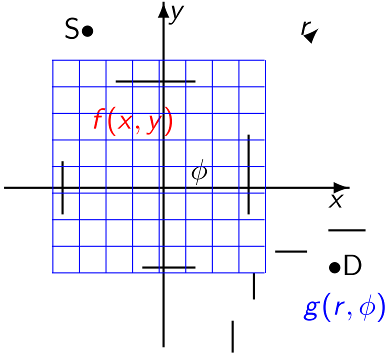
CT as a linear inverse problem

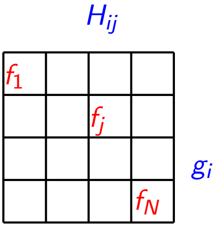


$$g(s_i) = \int_{L_i} f(\mathbf{r}) \, \mathrm{d}l_i + \epsilon(s_i) \longrightarrow \text{Discretization} \longrightarrow \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- **g**, **f** and **H** are huge dimensional

Algebraic methods: Discretization





$$f(x,y) = \sum_j f_j\, b_j(x,y)$$

$$b_j(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

$$g(r,\phi) = \int_L f(x,y)\, \mathrm{d}l$$

$$g_i = \sum_{j=1}^{N} H_{ij}\, f_j + \epsilon_i$$

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

Inversion: Deterministic methods

Data matching

- Observation model
  $g_i = h_i(\mathbf{f}) + \epsilon_i, \quad i = 1, \ldots, M \longrightarrow \mathbf{g} = \mathbf{H}(\mathbf{f}) + \boldsymbol{\epsilon}$
- Misatch between data and output of the model $\Delta(\mathbf{g}, \mathbf{H}(\mathbf{f}))$

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{\Delta(\mathbf{g}, \mathbf{H}(\mathbf{f}))\}$$

- Examples:

– LS $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \|\mathbf{g} - \mathbf{H}(\mathbf{f})\|^2 = \sum_i |g_i - h_i(\mathbf{f})|^2$

– L_p $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \|\mathbf{g} - \mathbf{H}(\mathbf{f})\|^p = \sum_i |g_i - h_i(\mathbf{f})|^p, \; 1 < p < 2$

– KL $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \sum_i g_i \ln \frac{g_i}{h_i(\mathbf{f})}$

- In general, does not give satisfactory results for inverse problems.

Deterministic Inversion Algorithms

Least Squares Based Methods

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\} \quad \text{with} \quad J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2$$

$$\nabla J(\mathbf{f}) = -2\mathbf{H}^t(\mathbf{g} - \mathbf{H}\mathbf{f})$$

Gradient based algorithms:

- Initialize: $\mathbf{f}^{(0)}$
- Iterate: $\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} - \alpha \nabla J(\mathbf{f}^{(k)})$

At each iteration: $\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha \mathbf{H}^t \left(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)}\right)$

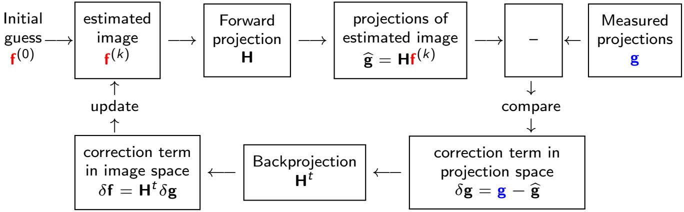
we have to do the following operations:

- Compute $\widehat{\mathbf{g}} = \mathbf{H}\mathbf{f}$ (Forward projection)
- Compute $\delta\mathbf{g} = \mathbf{g} - \widehat{\mathbf{g}}$ (Error or residual)
- Distribute $\delta\mathbf{f} = \mathbf{H}^t \delta\mathbf{g}$ (Backprojection of error)
- Update $\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \delta\mathbf{f}$

Gradient based algorithms

Operations at each iteration: $\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha \mathbf{H}^t \left(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)} \right)$

- Compute $\widehat{\mathbf{g}} = \mathbf{H}\mathbf{f}$ (Forward projection)
- Compute $\delta\mathbf{g} = \mathbf{g} - \widehat{\mathbf{g}}$ (Error or residual)
- Distribute $\delta\mathbf{f} = \mathbf{H}^t \delta\mathbf{g}$ (Backprojection of error)
- Update $\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \delta\mathbf{f}$



Gradient based algorithms

- Fixed step gradient:

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

- Steepest descent gradient:

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

  with $\alpha^{(k)} = \arg\min_{\alpha} \{ J(\mathbf{f} + \alpha \delta \mathbf{f}) \}$

- Conjugate Gradient

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

  The successive directions $\mathbf{d}^{(k)}$ have to be conjugate to each other.

Algebraic Reconstruction Techniques

- Main idea: Use the data as they arrive

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha^{(k)} [\mathbf{H}^t]_{i*} \left(g_i - [\mathbf{H}\mathbf{f}^{(k)}]_i \right)$$

which can also be written as:

$$\begin{aligned} \mathbf{f}^{(k+1)} &= \mathbf{f}^{(k)} + \frac{\left(g_i - [\mathbf{H}\mathbf{f}^{(k)}]_i \right)}{\mathbf{h}_{i*}^t \mathbf{h}_{i*}} \mathbf{h}_{i*}^t \\ &= \mathbf{f}^{(k)} + \sum_i \frac{\left(g_i - \sum_j H_{ij} f_j^{(k)} \right)}{\sum_i Hij^2} H_{ij} \end{aligned}$$

Algebraic Reconstruction Techniques

- Main idea: Use the data as they arrive

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha^{(k)}[\mathbf{H}^t]_{i*} \left(g_i - [\mathbf{H}\mathbf{f}^{(k)}]_i\right)$$

which can also be written as:

$$\begin{aligned}\mathbf{f}^{(k+1)} &= \mathbf{f}^{(k)} + \frac{\left(g_i - [\mathbf{H}\mathbf{f}^{(k)}]_i\right)}{\mathbf{h}_{i*}^t \mathbf{h}_{i*}} \mathbf{h}_{i*}^t \\ &= \mathbf{f}^{(k)} + \sum_i \frac{\left(g_i - \sum_j H_{ij} f_j^{(k)}\right)}{\sum_i Hij^2} H_{ij}\end{aligned}$$

- Main idea: Update each pixel at each time

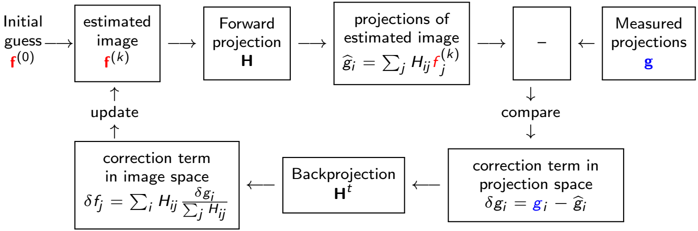
$$f_j^{(k+1)} = f_j^{(k)} + \frac{\left(g_i - \sum_j H_{ij} f_j^{(k)}\right)}{\sum_i Hij^2} H_{ij}$$

Algebraic Reconstruction Techniques (ART)

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \sum_i \frac{\left(g_i - \sum_j H_{ij} f_j^{(k)}\right)}{\sum_i Hij^2} H_{ij}$$

or

$$f_j^{(k+1)} = f_j^{(k)} + \frac{\left(g_i - \sum_j H_{ij} f_j^{(k)}\right)}{\sum_i Hij^2} H_{ij}$$

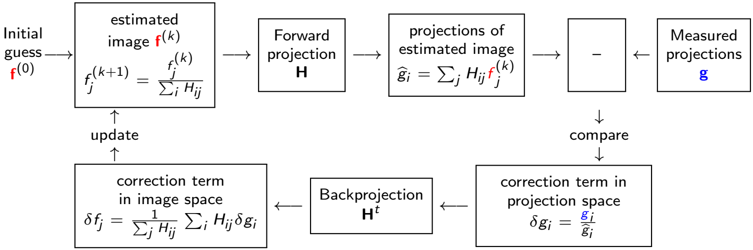


Algebraic Reconstruction using KL distance

- $\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$ with $J(\mathbf{f}) = \sum_i g_i \ln \frac{g_i}{\sum_j H_{ij} f_j}$

$$f_j^{(k+1)} = \frac{f_j^{(k)}}{\sum_i H_{ij}} \sum_i H_{ij} \frac{g_i}{\sum_j H_{ij} f_j^{(k)}}$$

Interestingly, this is the OSEM (Ordered subset Expectation-Maximization) algorithm which is based on Maximum Likelihood and proposed first by Shepp & Vardi.



Inversion: Regularization theory

Inverse problems = Ill posed problems

$\longrightarrow$ Need for prior information

Functional space (Tikhonov):

$$g = \mathcal{H}(f) + \epsilon \longrightarrow J(f) = ||g - \mathcal{H}(f)||_2^2 + \lambda ||\mathcal{D}f||_2^2$$

Finite dimensional space (Philips & Towmey): $\mathbf{g} = \mathbf{H}(\mathbf{f}) + \boldsymbol{\epsilon}$

- Minimum norme LS (MNLS): $J(\mathbf{f}) = ||\mathbf{g} - \mathbf{H}(\mathbf{f})||^2 + \lambda ||\mathbf{f}||_2^2$
- Classical Quadratic regularization: $J(\mathbf{f}) = ||\mathbf{g} - \mathbf{H}(\mathbf{f})||^2 + \lambda ||\mathbf{D}\mathbf{f}||_2^2$
- L1 or TV regularization: $J(\mathbf{f}) = ||\mathbf{g} - \mathbf{H}(\mathbf{f})||^2 + \lambda ||\mathbf{D}\mathbf{f}||_1$
- More general regularization:

$$J(\mathbf{f}) = \Delta_1(\mathbf{g}, \mathbf{H}(\mathbf{f})) + \lambda \Delta_2(\mathbf{f}, \mathbf{f}_0)$$

Limitations:

- Errors are considered implicitly white and Gaussian
- Limited prior information on the solution
- Lack of tools for the determination of the hyperparameters

Bayesian estimation approach

$$\mathcal{M}: \qquad \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- Observation model $\mathcal{M}$ + Hypothesis on the noise $\boldsymbol{\epsilon} \longrightarrow$
  $$p(\mathbf{g}|\mathbf{f};\mathcal{M}) = p_{\boldsymbol{\epsilon}}(\mathbf{g} - \mathbf{H}\mathbf{f})$$
- A priori information $\qquad p(\mathbf{f}|\mathcal{M})$
- Bayes : $$p(\mathbf{f}|\mathbf{g};\mathcal{M}) = \frac{p(\mathbf{g}|\mathbf{f};\mathcal{M})\, p(\mathbf{f}|\mathcal{M})}{p(\mathbf{g}|\mathcal{M})}$$

Link with regularization :

Maximum A Posteriori (MAP) :

$$\begin{aligned} \widehat{\mathbf{f}} &= \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\max_{\mathbf{f}} \{p(\mathbf{g}|\mathbf{f})\; p(\mathbf{f})\} \\ &= \arg\min_{\mathbf{f}} \{-\ln p(\mathbf{g}|\mathbf{f}) - \ln p(\mathbf{f})\} \end{aligned}$$

with $\quad Q(\mathbf{g},\mathbf{H}\mathbf{f}) = -\ln p(\mathbf{g}|\mathbf{f}) \quad$ and $\quad \lambda\Omega(\mathbf{f}) = -\ln p(\mathbf{f})$

But, Bayesian inference is not only limited to MAP

Case of linear models and Gaussian priors

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

- Hypothesis on the noise: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_\epsilon^2 \mathbf{I})$

$$p(\mathbf{g}|\mathbf{f}) \propto \exp\left[-\frac{1}{2\sigma_\epsilon^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2\right]$$

- Hypothesis on $\mathbf{f}$: $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \sigma_f^2 \mathbf{I})$

$$p(\mathbf{f}) \propto \exp\left[-\frac{1}{2\sigma_f^2}\|\mathbf{f}\|^2\right]$$

- A posteriori:

$$p(\mathbf{f}|\mathbf{g}) \propto \exp\left[-\frac{1}{2\sigma_\epsilon^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 - \frac{1}{2\sigma_f^2}\|\mathbf{f}\|^2\right]$$

- MAP : $\widehat{\mathbf{f}} = \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$

  with $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{f}\|^2, \qquad \lambda = \frac{\sigma_\epsilon^2}{\sigma_f^2}$

- Advantage : characterization of the solution

$$\mathbf{f}|\mathbf{g} \sim \mathcal{N}(\widehat{\mathbf{f}}, \widehat{\mathbf{P}}) \text{ with } \widehat{\mathbf{f}} = \widehat{\mathbf{P}}\mathbf{H}^t\mathbf{g}, \quad \widehat{\mathbf{P}} = \left(\mathbf{H}^t\mathbf{H} + \lambda\mathbf{I}\right)^{-1}$$

MAP estimation with other priors:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\} \quad \text{with} \quad J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\Omega(\mathbf{f})$$

Separable priors:

- Gaussian: $p(f_j) \propto \exp\left[-\alpha|f_j|^2\right] \longrightarrow \Omega(\mathbf{f}) = \|\mathbf{f}\|^2 = \alpha \sum_j |f_j|^2$
- Gamma: $p(f_j) \propto f_j^\alpha \exp\left[-\beta f_j\right] \longrightarrow \Omega(\mathbf{f}) = \alpha \sum_j \ln f_j + \beta f_j$
- Beta:
  $p(f_j) \propto f_j^\alpha (1 - f_j)^\beta \longrightarrow \Omega(\mathbf{f}) = \alpha \sum_j \ln f_j + \beta \sum_j \ln(1 - f_j)$
- Generalized Gaussian:
  $p(f_j) \propto \exp\left[-\alpha|f_j|^p\right], \quad 1 < p < 2 \longrightarrow \quad \Omega(\mathbf{f}) = \alpha \sum_j |f_j|^p,$

Markovian models:

$$p(f_j|\mathbf{f}) \propto \exp\left[-\alpha \sum_{i \in N_j} \phi(f_j, f_i)\right] \longrightarrow \quad \Omega(\mathbf{f}) = \alpha \sum_j \sum_{i \in N_j} \phi(f_j, f_i),$$

Main advantages of the Bayesian approach

- MAP = Regularization
- Posterior mean ? Marginal MAP ?
- More information in the posterior law than only its mode or its mean
- Meaning and tools for estimating hyper parameters
- Meaning and tools for model selection
- More specific and specialized priors, particularly through the hidden variables
- More computational tools:
  - Expectation-Maximization for computing the maximum likelihood parameters
  - MCMC for posterior exploration
  - Variational Bayes for analytical computation of the posterior marginals
  - ...

Full Bayesian approach

$$\mathcal{M}: \quad \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon$$

- Forward & errors model: $\longrightarrow p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1; \mathcal{M})$
- Prior models $\longrightarrow p(\mathbf{f}|\boldsymbol{\theta}_2; \mathcal{M})$
- Hyperparameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \longrightarrow p(\boldsymbol{\theta}|\mathcal{M})$
- Bayes: $\longrightarrow p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M}) = \frac{p(\mathbf{g}|\mathbf{f},\boldsymbol{\theta};\mathcal{M})\; p(\mathbf{f}|\boldsymbol{\theta};\mathcal{M})\; p(\boldsymbol{\theta}|\mathcal{M})}{p(\mathbf{g}|\mathcal{M})}$
- Joint MAP: $(\widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}) = \arg\max_{(\mathbf{f},\boldsymbol{\theta})} \{p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M})\}$
- Marginalization: $\begin{cases} p(\mathbf{f}|\mathbf{g}; \mathcal{M}) & = \int p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M})\, \mathrm{d}\mathbf{f} \\ p(\boldsymbol{\theta}|\mathbf{g}; \mathcal{M}) & = \int p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M})\, \mathrm{d}\boldsymbol{\theta} \end{cases}$
- Posterior means: $\begin{cases} \widehat{\mathbf{f}} & = \int \mathbf{f}\; p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M})\, \mathrm{d}\mathbf{f}\, \mathrm{d}\boldsymbol{\theta} \\ \widehat{\boldsymbol{\theta}} & = \int \boldsymbol{\theta}\; p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}; \mathcal{M})\, \mathrm{d}\mathbf{f}\, \mathrm{d}\boldsymbol{\theta} \end{cases}$
- Evidence of the model:

$$p(\mathbf{g}|\mathcal{M}) = \iint p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) p(\mathbf{f}|\boldsymbol{\theta}; \mathcal{M}) p(\boldsymbol{\theta}|\mathcal{M})\, \mathrm{d}\mathbf{f}\, \mathrm{d}\boldsymbol{\theta}$$

MAP estimation with different prior models

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ p(\mathbf{f}) = \mathcal{N}\left(\mathbf{0}, \sigma_f^2 (\mathbf{D}^t\mathbf{D})^{-1}\right) \end{cases} = \begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ \mathbf{f} = \mathbf{C}\mathbf{f} + \mathbf{z} \quad \text{with} \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_f^2 \mathbf{I}) \\ \mathbf{D}\mathbf{f} = \mathbf{z} \text{ with } \mathbf{D} = (\mathbf{I} - \mathbf{C}) \end{cases}$$

$$p(\mathbf{f}|\mathbf{g}) = \mathcal{N}(\widehat{\mathbf{f}}, \widehat{\mathbf{P}}_f) \quad \text{with} \quad \widehat{\mathbf{f}} = \widehat{\mathbf{P}}_f \mathbf{H}^t \mathbf{g}, \quad \widehat{\mathbf{P}}_f = \left(\mathbf{H}^t\mathbf{H} + \lambda \mathbf{D}^t\mathbf{D}\right)^{-1}$$

$$J(\mathbf{f}) = -\ln p(\mathbf{f}|\mathbf{g}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda \|\mathbf{D}\mathbf{f}\|^2$$

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ p(\mathbf{f}) = \mathcal{N}\left(\mathbf{0}, \sigma_f^2 (\mathbf{W}\mathbf{W}^t)\right) \end{cases} = \begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ \mathbf{f} = \mathbf{W}\mathbf{z} \quad \text{with} \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_f^2 \mathbf{I}) \end{cases}$$

$$p(\mathbf{z}|\mathbf{g}) = \mathcal{N}(\widehat{\mathbf{z}}, \widehat{\mathbf{P}}_z) \text{ with } \widehat{\mathbf{z}} = \widehat{\mathbf{P}}_z \mathbf{W}^t \mathbf{H}^t \mathbf{g}, \; \widehat{\mathbf{P}}_z = \left(\mathbf{W}^t\mathbf{H}^t\mathbf{H}\mathbf{W} + \lambda \mathbf{I}\right)^{-1}$$

$$J(\mathbf{z}) = -\ln p(\mathbf{z}|\mathbf{g}) = \|\mathbf{g} - \mathbf{H}\mathbf{W}\mathbf{z}\|^2 + \lambda \|\mathbf{z}\|^2 \longrightarrow \widehat{\mathbf{f}} = \mathbf{W}\widehat{\mathbf{z}}$$

$\mathbf{z}$ decomposition coefficients

MAP estimation and Compressed Sensing

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ \mathbf{f} = \mathbf{W}\mathbf{z} \end{cases}$$

- $\mathbf{W}$ a code book matrix, $\mathbf{z}$ coefficients
- Gaussian:

$$\begin{array}{l} p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \sigma_z^2\mathbf{I}) \propto \exp\left[-\frac{1}{2\sigma_z^2}\sum_j |z_j|^2\right] \\ J(\mathbf{z}) = -\ln p(\mathbf{z}|\mathbf{g}) = \|\mathbf{g} - \mathbf{H}\mathbf{W}\mathbf{z}\|^2 + \lambda \sum_j |z_j|^2 \end{array}$$

- Generalized Gaussian (sparsity, $\beta = 1$):

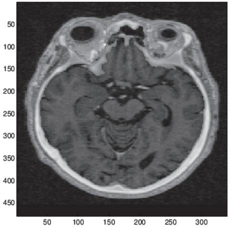
$$\begin{array}{l} p(\mathbf{z}) \propto \exp\left[-\lambda \sum_j |z_j|^\beta\right] \\ J(\mathbf{z}) = -\ln p(\mathbf{z}|\mathbf{g}) = \|\mathbf{g} - \mathbf{H}\mathbf{W}\mathbf{z}\|^2 + \lambda \sum_j |z_j|^\beta \end{array}$$

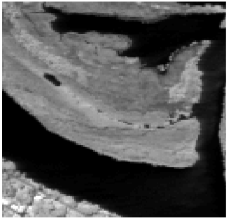
- $\mathbf{z} = \arg\min_{\mathbf{z}} \{J(\mathbf{z})\} \longrightarrow \widehat{\mathbf{f}} = \mathbf{W}\widehat{\mathbf{z}}$

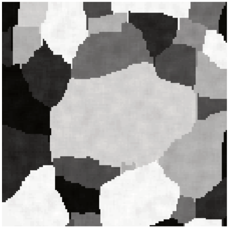
Two main steps in the Bayesian approach

- Prior modeling
  - Separable:
    Gaussian, Generalized Gaussian, Gamma,
    mixture of Gaussians, mixture of Gammas, ...
  - Markovian: Gauss-Markov, GGM, ...
  - Separable or Markovian with hidden variables
    (contours, region labels)
- Choice of the estimator and computational aspects
  - MAP, Posterior mean, Marginal MAP
  - MAP needs optimization algorithms
  - Posterior mean needs integration methods
  - Marginal MAP needs integration and optimization
  - Approximations:
    - Gaussian approximation (Laplace)
    - Numerical exploration MCMC
    - Variational Bayes (Separable approximation)

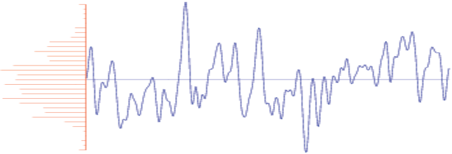
Which images I am looking for?





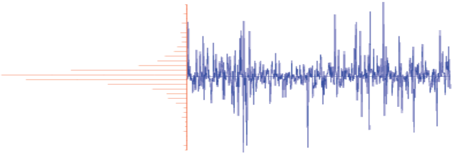


Which image I am looking for?



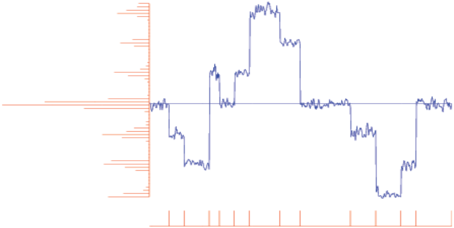
Gaussian

$p(f_j|f_{j-1}) \propto \exp\left[-\alpha|f_j - f_{j-1}|^2\right]$



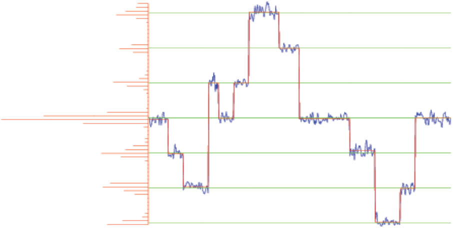
Generalized Gaussian

$p(f_j|f_{j-1}) \propto \exp\left[-\alpha|f_j - f_{j-1}|^p\right]$



Piecewize Gaussian

$p(f_j|q_j, f_{j-1}) = \mathcal{N}\left((1-q_j)f_{j-1}, \sigma_f^2\right)$



Mixture of GM

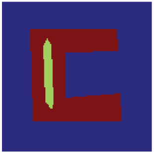
$p(f_j|z_j = k) = \mathcal{N}\left(m_k, \sigma_k^2\right)$

Gauss-Markov-Potts prior models for images

"In NDT applications of CT, the objects are, in general, composed of a finite number of materials, and the voxels corresponding to each materials are grouped in compact regions"

How to model this prior information?

$f(\mathbf{r})$



$z(\mathbf{r}) \in \{1, ..., K\}$

$$p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(m_k, v_k)$$

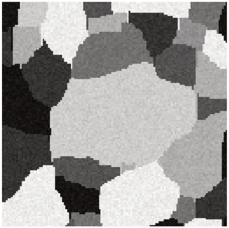
$$p(f(\mathbf{r})) = \sum_k P(z(\mathbf{r}) = k)\,\mathcal{N}(m_k, v_k) \quad \text{Mixture of Gaussians}$$

$$p(z(\mathbf{r})|z(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) \propto \exp\left[\gamma \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

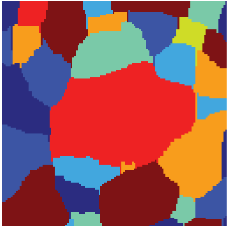
Four different cases

To each pixel of the image is associated 2 variables $f(\mathbf{r})$ and $z(\mathbf{r})$

- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ iid :
  Mixture of Gaussians
- $\mathbf{f}|\mathbf{z}$ Gauss-Markov, $\mathbf{z}$ iid :
  Mixture of Gauss-Markov
- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ Potts-Markov :
  Mixture of Independent Gaussians
  (MIG with Hidden Potts)
- $\mathbf{f}|\mathbf{z}$ Markov, $\mathbf{z}$ Potts-Markov :
  Mixture of Gauss-Markov
  (MGM with hidden Potts)

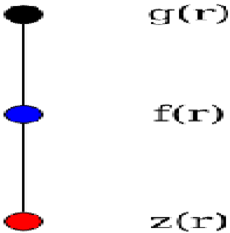


$f(\mathbf{r})$

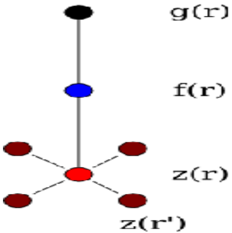


$z(\mathbf{r})$

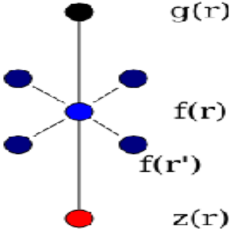
Four different cases



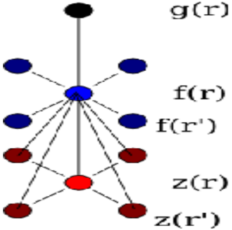
Case 1: Mixture of Gaussians



Case 2: Mixture of Gauss-Markov

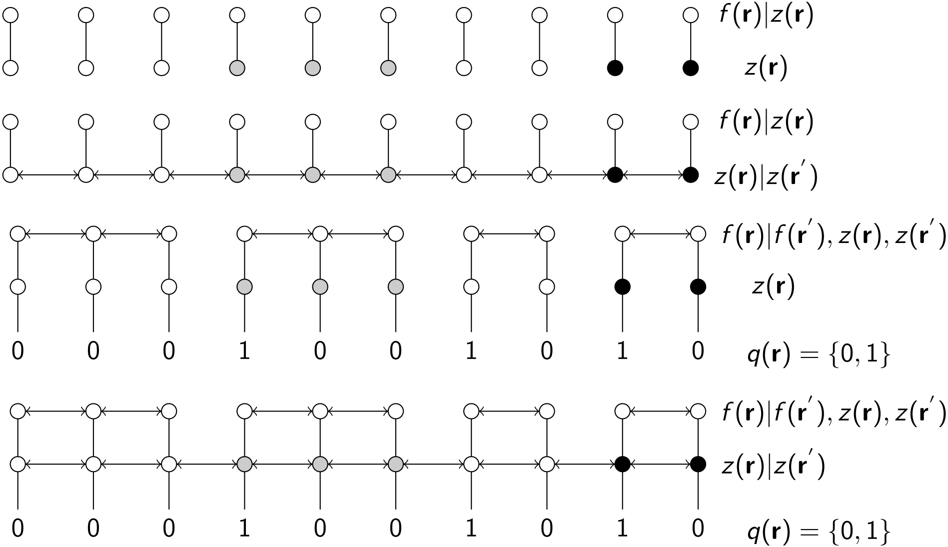


Case 3: MIG with Hidden Potts



Case 4: MGM with hidden Potts

Four different cases



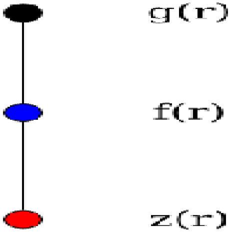
Case 1: $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ iid

Independent Mixture of Independent Gaussiens (IMIG):

$$p(f(\mathbf{r})|z(\mathbf{r})=k)=\mathcal{N}(m_k,v_k), \quad \forall \mathbf{r}\in\mathcal{R}$$

$$p(f(\mathbf{r}))=\sum_{k=1}^K \alpha_k \mathcal{N}(m_k,v_k), \text{with } \sum_k \alpha_k=1.$$

$$p(\mathbf{z})=\prod_{\mathbf{r}} p(z(\mathbf{r})=k)=\prod_{\mathbf{r}} \alpha_k=\prod_k \alpha_k^{n_k}$$



Noting $\quad \mathcal{R}_k=\{\mathbf{r}: z(\mathbf{r})=k\}, \quad \mathcal{R}=\cup_k \mathcal{R}_k,$

$$m_z(\mathbf{r})=m_k, v_z(\mathbf{r})=v_k, \alpha_z(\mathbf{r})=\alpha_k, \forall \mathbf{r}\in\mathcal{R}_k$$

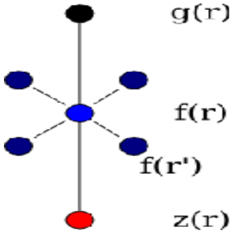
we have:

$$p(\mathbf{f}|\mathbf{z}) = \prod_{\mathbf{r}\in\mathcal{R}} \mathcal{N}(m_z(\mathbf{r}),v_z(\mathbf{r}))$$

$$p(\mathbf{z}) = \prod_{\mathbf{r}} \alpha_z(\mathbf{r})=\prod_k \alpha_k^{\sum_{\mathbf{r}\in\mathcal{R}} \delta(z(\mathbf{r})-k)}=\prod_k \alpha_k^{n_k}$$

Case 2: $\mathbf{f}|\mathbf{z}$ Gauss-Markov, $\mathbf{z}$ iid

Independent Mixture
of Gauss-Markov (IMGM):



$$p(f(\mathbf{r})|z(\mathbf{r}), z(\mathbf{r}'), f(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r}))$$

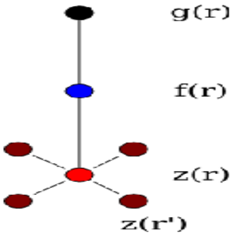
$$= \mathcal{N}(\mu_z(\mathbf{r}), v_z(\mathbf{r})), \forall \mathbf{r} \in \mathcal{R}$$

$$\begin{aligned}
\mu_z(\mathbf{r}) &= \frac{1}{|\mathcal{V}(\mathbf{r})|} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \mu_z^*(\mathbf{r}') \\
\mu_z^*(\mathbf{r}') &= \delta(z(\mathbf{r}') - z(\mathbf{r}))\, f(\mathbf{r}') + (1 - \delta(z(\mathbf{r}') - z(\mathbf{r}))\, m_z(\mathbf{r}') \\
&= (1 - c(\mathbf{r}'))\quad f(\mathbf{r}') \;+\quad c(\mathbf{r}')\quad m_z(\mathbf{r}')
\end{aligned}$$

$$\begin{aligned}
p(\mathbf{f}|\mathbf{z}) &\propto \prod_{\mathbf{r}} \mathcal{N}(\mu_z(\mathbf{r}), v_z(\mathbf{r})) &\propto \prod_k \alpha_k \mathcal{N}(m_k \mathbf{1}, \mathbf{\Sigma}_k) \\
p(\mathbf{z}) &= \prod_{\mathbf{r}} v_z(\mathbf{r}) &= \prod_k \alpha_k^{n_k}
\end{aligned}$$

with $\mathbf{1}_k = 1, \forall \mathbf{r} \in \mathcal{R}_k$ and $\mathbf{\Sigma}_k$ a covariance matrix $(n_k \times n_k)$.

Case 3: $\mathbf{f}|\mathbf{z}$ Gauss iid, $\mathbf{z}$ Potts



Gauss iid as in Case 1:

$$\begin{aligned} p(\mathbf{f}|\mathbf{z}) &= \prod_{\mathbf{r}\in\mathcal{R}} \mathcal{N}(m_z(\mathbf{r}), v_z(\mathbf{r})) \\ &= \prod_k \prod_{\mathbf{r}\in\mathcal{R}_k} \mathcal{N}(m_k, v_k) \end{aligned}$$

Potts-Markov:

$$p(z(\mathbf{r})|z(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) \propto \exp\left[\gamma \sum_{\mathbf{r}'\in\mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

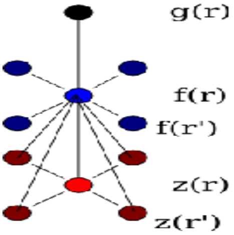
$$p(\mathbf{z}) \propto \exp\left[\gamma \sum_{\mathbf{r}\in\mathcal{R}} \sum_{\mathbf{r}'\in\mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

Case 4: $\mathbf{f}|\mathbf{z}$ Gauss-Markov, $\mathbf{z}$ Potts

Gauss-Markov as in Case 2:

$p(f(\mathbf{r})|z(\mathbf{r}), z(\mathbf{r}'), f(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) =$

$\mathcal{N}(\mu_z(\mathbf{r}), v_z(\mathbf{r})), \forall \mathbf{r} \in \mathcal{R}$



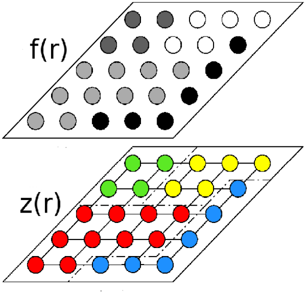
$$\begin{aligned} \mu_z(\mathbf{r}) &= \frac{1}{|\mathcal{V}(\mathbf{r})|} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \mu_z^*(\mathbf{r}') \\ \mu_z^*(\mathbf{r}') &= \delta(z(\mathbf{r}') - z(\mathbf{r}))\, f(\mathbf{r}') + (1 - \delta(z(\mathbf{r}') - z(\mathbf{r}))\, m_z(\mathbf{r}') \end{aligned}$$

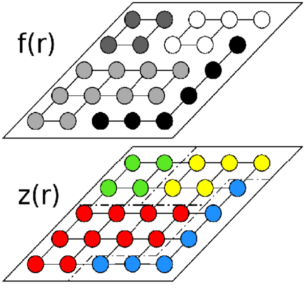
$$p(\mathbf{f}|\mathbf{z}) \propto \prod_{\mathbf{r}} \mathcal{N}(\mu_z(\mathbf{r}), v_z(\mathbf{r})) \propto \prod_k \alpha_k \mathcal{N}(m_k \mathbf{1}, \mathbf{\Sigma}_k)$$

Potts-Markov as in Case 3:

$$p(\mathbf{z}) \propto \exp \left[\gamma \sum_{\mathbf{r} \in \mathcal{R}} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}')) \right]$$

Summary of the two proposed models





$\mathbf{f}|\mathbf{z}$ Gaussian iid
$\mathbf{z}$ Potts-Markov

(MIG with Hidden Potts)

$\mathbf{f}|\mathbf{z}$ Markov
$\mathbf{z}$ Potts-Markov

(MGM with hidden Potts)

Bayesian Computation

$$p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f},\mathbf{z},v_\epsilon)\, p(\mathbf{f}|\mathbf{z},\mathbf{m},\mathbf{v})\, p(\mathbf{z}|\gamma,\boldsymbol{\alpha})\, p(\boldsymbol{\theta})$$

$$\boldsymbol{\theta} = \{v_\epsilon, (\alpha_k, m_k, v_k), k = 1, \cdot, K\} \qquad p(\boldsymbol{\theta}) \quad \text{Conjugate priors}$$

- Direct computation and use of $p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g};\mathcal{M})$ is too complex
- Possible approximations :
  - Gauss-Laplace (Gaussian approximation)
  - Exploration (Sampling) using MCMC methods
  - Separable approximation (Variational techniques)
- Main idea in Variational Bayesian methods:
  Approximate

  $$p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g};\mathcal{M}) \quad \text{by} \quad q(\mathbf{f},\mathbf{z},\boldsymbol{\theta}) = q_1(\mathbf{f})\; q_2(\mathbf{z})\; q_3(\boldsymbol{\theta})$$

  - Choice of approximation criterion : $KL(q:p)$
  - Choice of appropriate families of probability laws for $q_1(\mathbf{f})$, $q_2(\mathbf{z})$ and $q_3(\boldsymbol{\theta})$

MCMC based algorithm

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{z}, \boldsymbol{\theta})\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta})\, p(\mathbf{z})\, p(\boldsymbol{\theta})$$

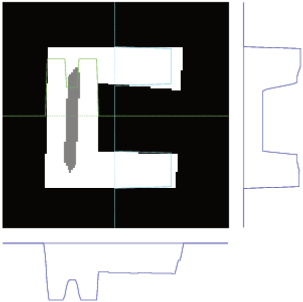
General scheme:

$$\widehat{\mathbf{f}} \sim p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g})$$

- Sample $\mathbf{f}$ from $p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta})\, p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}})$
  Needs optimisation of a quadratic criterion.
- Sample $\mathbf{z}$ from $p(\mathbf{z}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}})\, p(\mathbf{z})$
  Needs sampling of a Potts Markov field.
- Sample $\boldsymbol{\theta}$ from $p(\boldsymbol{\theta}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}}, \sigma_\epsilon^2 \mathbf{I})\, p(\widehat{\mathbf{f}}|\widehat{\mathbf{z}}, (m_k, v_k))\, p(\boldsymbol{\theta})$
  Conjugate priors $\longrightarrow$ analytical expressions.

Application of CT in NDT

Reconstruction from only 2 projections



$$g_1(x) = \int f(x,y)\,dy$$

$$g_2(y) = \int f(x,y)\,dx$$

- Given the marginals $g_1(x)$ and $g_2(y)$ find the joint distribution $f(x,y)$.
- Infinite number of solutions : $f(x,y) = g_1(x)\,g_2(y)\,\Omega(x,y)$ $\Omega(x,y)$ is a Copula:

$$\int \Omega(x,y)\,dx = 1 \quad \text{and} \quad \int \Omega(x,y)\,dy = 1$$

Application in CT

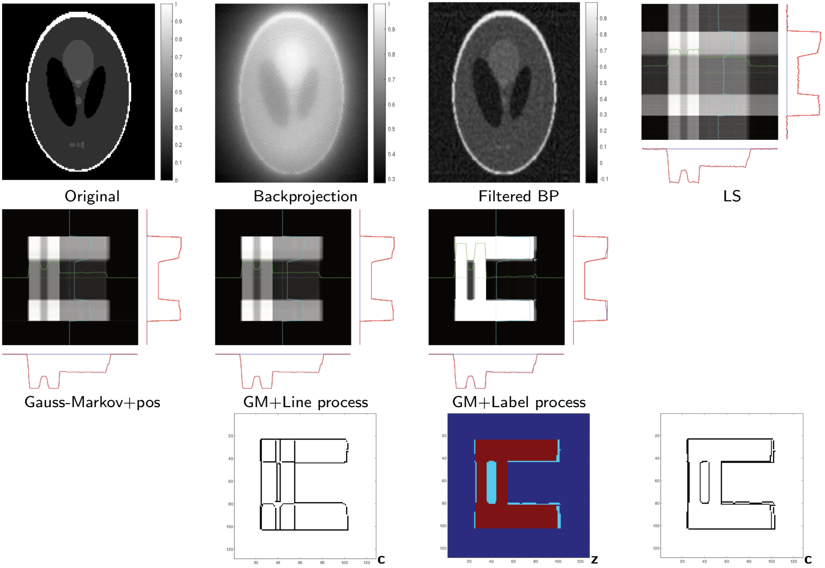
$\mathbf{g}\vert\mathbf{f}$	$\mathbf{f}\vert\mathbf{z}$	$\mathbf{z}$	$\mathbf{q}$
$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$	iid Gaussian	iid	$q(\mathbf{r}) \in \{0,1\}$
$\mathbf{g}\vert\mathbf{f} \sim \mathcal{N}(\mathbf{H}\mathbf{f}, \sigma_\epsilon^2\mathbf{I})$	or	or	$1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$
Gaussian	Gauss-Markov	Potts	binary

Forward model | Gauss-Markov-Potts Prior Model | Auxilary

Unsupervised Bayesian estimation:

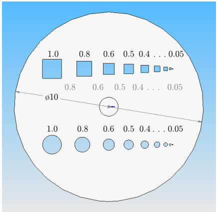
$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{z}, \boldsymbol{\theta})\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta})\, p(\boldsymbol{\theta})$$

Results: 2D case

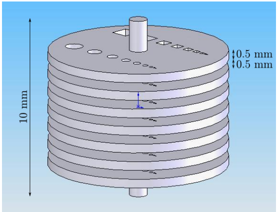


Some results in 3D case

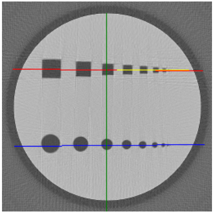
(Results obtained with collaboration with CEA)



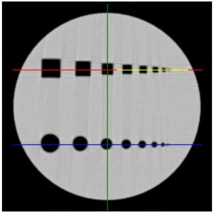
M. Defrise



Phantom

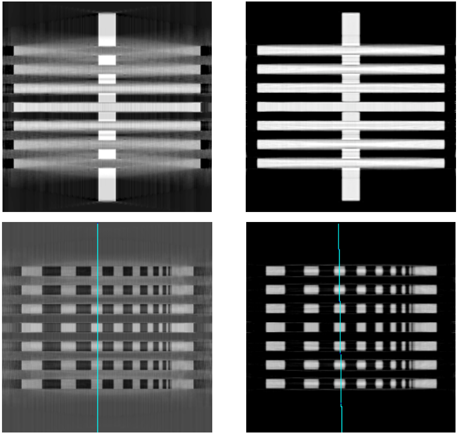


FeldKamp



Proposed method

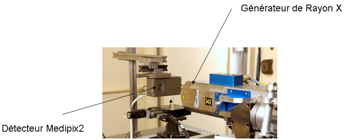
Some results in 3D case



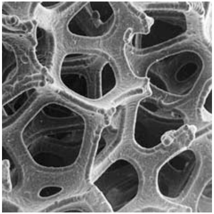
FeldKamp Proposed method

Some results in 3D case

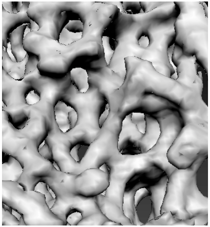
Experimental setup

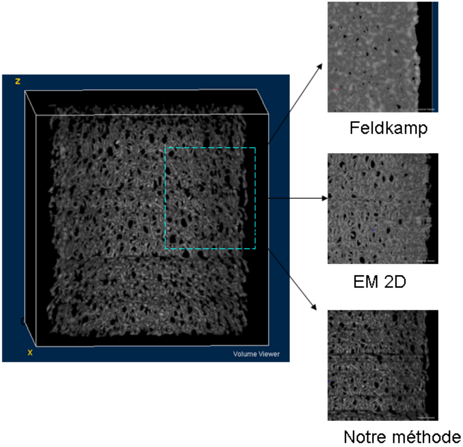


A photograpy of metalique esponge

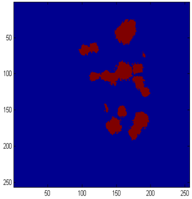


Reconstruction by proposed method

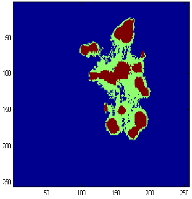




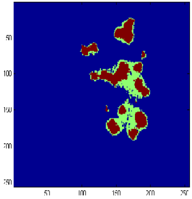
Application: liquid evaporation in metalic esponge



Time 0



Time 1



Time 2

Conclusions

- Gauss-Markov-Potts are useful prior models for images incorporating regions and contours
- Bayesian computation needs often pproximations (Laplace, MCMC, Variational Bayes)
- Application in different CT systems (X ray, Ultrasound, Microwave, PET, SPECT) as well as other inverse problems

Work in Progress and Perspectives :

- Efficient implementation in 2D and 3D cases using GPU
- Evaluation of performances and comparison with MCMC methods
- Application to other linear and non linear inverse problems: (PET, SPECT or ultrasound and microwave imaging)

Classical Linear Inverse Problems

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

- Regularization (L2, L1, TV, ...)

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$$

$$R(\mathbf{f}) = \left\{ \|\mathbf{f}\|_2^2, \|\mathbf{f}\|_2^2, \|\mathbf{f}\|_\beta^\beta, \|\mathbf{D}\mathbf{f}\|_2^2, \|\mathbf{D}\mathbf{f}\|_1, \|\mathbf{D}\mathbf{f}\|_\beta^\beta, \ldots \right\}$$

- Bayesian MAP

$$\begin{cases} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \propto \exp\left[\frac{-1}{2v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_1\right] \\ p(\mathbf{f}) \propto \exp\left[\frac{-1}{2v_f}\|\mathbf{f}\|_2^2\right], \quad \exp\left[\frac{-1}{2v_f}\|\mathbf{D}\mathbf{f}\|_2\right], \quad \exp\left[\frac{-1}{2v_f}\|\mathbf{f}\|_1\right], \ldots \end{cases}$$

$$p(\mathbf{f}|\mathbf{g}) \propto \exp\left[\frac{-1}{2v_f} J(\mathbf{f})\right] \rightarrow \widehat{\mathbf{f}}_{MAP} = \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$$

Variable splitting or How to account for all uncertainties

Go beyound the classical forward model:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon} \longrightarrow \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} + \boldsymbol{\epsilon}, \quad \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} + \mathbf{u} + \boldsymbol{\epsilon}$$

Variable splitting: Different interpretations

$$Case1: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} \end{array} \right. \qquad Case4: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = \mathbf{H}(\mathbf{f} + \mathbf{f}_0) + \boldsymbol{\xi} \end{array} \right.$$

$$Case2: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \mathbf{u} + \boldsymbol{\xi} \end{array} \right. \qquad Case5: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \mathbf{u} + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} \end{array} \right.$$

$$Case3: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = (\mathbf{H} + \delta\mathbf{H})\mathbf{f} + \boldsymbol{\xi}, \end{array} \right. \qquad Case6: \left\{ \begin{array}{l} \mathbf{g} = \mathbf{g}_0 + \boldsymbol{\epsilon}, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \mathbf{u} + \boldsymbol{\xi}, \\ \mathbf{u} = \delta\mathbf{H}\,\mathbf{f} + \boldsymbol{\zeta}. \end{array} \right.$$

State of the art Regularization methods for the simple case

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

Regularization (or MAP):

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$$

$$R(\mathbf{f}) = \left\{ \|\mathbf{f}\|_2^2,\ \|\mathbf{f}\|_1,\ \|\mathbf{f}\|_\beta^\beta,\ \|\mathbf{D}\mathbf{f}\|_2^2,\ \|\mathbf{D}\mathbf{f}\|_1,\ \|\mathbf{D}\mathbf{f}\|_\beta^\beta \right\}$$

Optimization algorithms:

- Gradient based (Steepest descent, CG, ...)

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha^{(k)} [\mathbf{H}'(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)}) - \lambda \mathbf{f}^{(k)}]$$

- Augmented Lagrangian (ADMM):
  Minimize $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$ subject to $\mathbf{g} = \mathbf{H}\mathbf{f}$
- Bregman convex optimization (ISTA, FISTA,...):
  Minimize $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$ subject to $R(\mathbf{f}) \leq q\|\mathbf{f}\|^2$

Augmented Lagrangian (AL)

- Minimize $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$ subject to $\mathbf{g} = \mathbf{H}\mathbf{f}$
- Lagrangian:

$$\mathcal{L}(\mathbf{f}, \boldsymbol{\mu}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{D}\mathbf{f}) + \boldsymbol{\mu}'(\mathbf{H}\mathbf{f} - \mathbf{g}),$$

which gives the following algorithm:

$$\begin{cases} \mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha_1^{(k)}[2\mathbf{H}'(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)}) - \lambda \mathbf{D}'\nabla R(\mathbf{f}^{(k)}) - \mathbf{H}'\boldsymbol{\mu}] \\ \boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \alpha_2^{(k)}(\mathbf{H}\mathbf{f}^{(k)} - \mathbf{g}). \end{cases}$$

- Particular case of $R(\mathbf{f}) = \|\mathbf{f}\|_1$

$$\begin{cases} \mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha_1^{(k)} ST_{\boldsymbol{\mu}}[2\mathbf{H}'(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)})] \\ \boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \alpha_2^{(k)}(\mathbf{H}\mathbf{f}^{(k)} - \mathbf{g}). \end{cases}$$

Augmented Lagrangian (ADMM)

- Synthesis criterion and AL:

$$\min\ J(\mathbf{z}) = \|\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}\|_2^2 + \lambda R(\mathbf{z}) \text{ s.t. } \mathbf{H}\mathbf{D}\mathbf{z} = \mathbf{g},$$

for which the solution is obtained as the stationary point of the AL:

$$\mathcal{L}(\mathbf{z}, \boldsymbol{\mu}) = \|\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}\|_2^2 + \lambda R(\mathbf{z}) + \boldsymbol{\mu}'(\mathbf{H}\mathbf{D}\mathbf{z} - \mathbf{g}),$$

which gives the following algorithm:

$$\begin{cases} \mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} + \alpha_1^{(k)}[\mathbf{D}'\mathbf{H}'(\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}^{(k)}) - \lambda \nabla R(\mathbf{z}^{(k)}) - \mathbf{D}'\mathbf{H}'\boldsymbol{\mu}] \\ \boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \alpha_2^{(k)}(\mathbf{H}\mathbf{D}\mathbf{f}^{(k)} - \mathbf{g}). \end{cases}$$

At the end of the iterations, we can compute $\widehat{\mathbf{f}} = \mathbf{D}\widehat{\mathbf{z}}$.

Bregman convex optimization (ISTA, FISTA? ...)

Bregman convex optimization (ISTA, FISTA,...):
Minimize $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda R(\mathbf{f})$ subject to $R(\mathbf{f}) \leq q\|\mathbf{f}\|^2$
MinMax, Duality, ...
Particular case: $R(\mathbf{f}) = |\mathbf{D}\mathbf{f}|_1$:

Minimize $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + q\|\mathbf{D}\mathbf{f}\|_2^2$ subject to $\mathbf{D}\mathbf{f} = \mathbf{z}$

$$\mathcal{L}(\mathbf{f}, \mathbf{z}, \boldsymbol{\mu}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + q\|\mathbf{z}\|_1 + \boldsymbol{\mu}'(\mathbf{D}\mathbf{f} - \mathbf{z}),$$

$$\begin{cases} \mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha_1^{(k)} ST_{\boldsymbol{\mu}}[2\mathbf{H}'(\mathbf{g} - \mathbf{H}\mathbf{f}^{(k)})] \\ \boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \alpha_2^{(k)}(\mathbf{H}\mathbf{f}^{(k)} - \mathbf{g}). \end{cases}$$

State of the Art Bayesian methods for the simple case

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

Supervized Gaussian case:

$$\left\{ \begin{array}{l} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mathbf{0}, v_f \mathbf{I}) \end{array} \right. \rightarrow \left\{ \begin{array}{l} p(\mathbf{f}|\mathbf{g}) = \mathcal{N}(\mathbf{f}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}}) \\ \widehat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \lambda \mathbf{I}]^{-1} \mathbf{H}'\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = v_\epsilon [\mathbf{H}'\mathbf{H} + \lambda \mathbf{I}]^{-1}, \quad \lambda = \frac{v_\epsilon}{v_f} \end{array} \right.$$

Unsupervised Gaussian

$$\left\{ \begin{array}{l} p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|v_f) = \mathcal{N}(\mathbf{f}|\mathbf{0}, v_f \mathbf{I}) \\ p(v_\epsilon) = \mathcal{IG}(v_f|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_f) = \mathcal{IG}(v_f|\alpha_{f_0}, \beta_{f_0}) \end{array} \right. \rightarrow \left\{ \begin{array}{l} p(\mathbf{f}|\mathbf{g}, v_\epsilon, v_f) = \mathcal{N}(\mathbf{f}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}}) \\ \widehat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \widehat{\lambda} \mathbf{I}]^{-1} \mathbf{H}'\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = \hat{v}_\epsilon [\mathbf{H}'\mathbf{H} + \widehat{\lambda} \mathbf{I}]^{-1}, \ \widehat{\lambda} = \frac{\hat{v}_\epsilon}{\hat{v}_f} \\ p(v_\epsilon|\mathbf{g}, \mathbf{f}) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon) \\ p(v_f|\mathbf{g}) = \mathcal{IG}(v_f|\widetilde{\alpha}_f, \widetilde{\beta}_f) \\ \widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon, \widetilde{\alpha}_f, \widetilde{\beta}_f \end{array} \right.$$

Different inference tools: JMAP, Gibbs sampling MCMC, VBA

State of the Art Bayesian methods for the simple case

$$\begin{cases} p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|v_f) = \mathcal{N}(\mathbf{f}|\mathbf{0}, v_f \mathbf{I}) \\ p(v_\epsilon) = \mathcal{IG}(v_f|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_f) = \mathcal{IG}(v_f|\alpha_{f_0}, \beta_{f_0}) \end{cases} \rightarrow \begin{cases} p(\mathbf{f}|\mathbf{g}, v_\epsilon, v_f) = \mathcal{N}(\mathbf{f}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}}) \\ \widehat{\mathbf{f}} = [\mathbf{H}'\mathbf{H} + \widehat{\lambda}\mathbf{I}]^{-1}\mathbf{H}'\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = \hat{v}_\epsilon[\mathbf{H}'\mathbf{H} + \widehat{\lambda}\mathbf{I}]^{-1}, \ \widehat{\lambda} = \frac{\hat{v}_\epsilon}{\hat{v}_f} \\ p(v_\epsilon|\mathbf{g}, \mathbf{f}) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon) \\ p(v_f|\mathbf{g}, \mathbf{f}) = \mathcal{IG}(v_f|\widetilde{\alpha}_f, \widetilde{\beta}_f) \end{cases}$$

$$p(\mathbf{f}, v_\epsilon, v_\xi|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, v_\epsilon, v_\xi)\right]$$

- JMAP: Alternate optimization with respect to $\mathbf{f}, v_\epsilon, v_f$:

$$J(\mathbf{f}, v_\epsilon, v_f) = \frac{1}{2v_\epsilon}\|\mathbf{g}-\mathbf{H}\mathbf{f}\|_2^2+(\alpha_{\epsilon_0}+1)\ln v_\epsilon+\beta_{\epsilon_0}/v_\epsilon+(\alpha_{f_0}+1)\ln v_f+\beta_{f_0}/v_f$$

- Gibbs sampling MCMC:

$$\mathbf{f} \sim p(\mathbf{f}, v_\epsilon, v_\xi|\mathbf{g}) \rightarrow v_\epsilon \sim p(v_\epsilon|\mathbf{g}, \mathbf{f}) \rightarrow v_f \sim p(v_f|\mathbf{g}, \mathbf{f})$$

- Variational Bayesian Approximation: Approximate $p(\mathbf{f}, v_\epsilon, v_\xi|\mathbf{g})$ by a separable one $q(\mathbf{f}, v_\epsilon, v_\xi) = q_1(\mathbf{f}, v_\epsilon, v_\xi)q_2(v_\epsilon)q_3(v_\xi)$ minimizing $\mathrm{KL}(q|p)$.

Hierarchical models for more robustness

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon, \\ \mathbf{f} = \mathbf{D}\mathbf{z} + \zeta, \quad \mathbf{z} \text{ sparse DE} \end{cases}$$

Supervized Gaussian case:

$$\begin{cases} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}) \rightarrow \\ p(\mathbf{z}) = \mathcal{DE}(\mathbf{f}|\gamma) \propto \exp\left[-\gamma\|\mathbf{z}\|_1\right] \end{cases} \begin{cases} p(\mathbf{f}, \mathbf{z}|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, \mathbf{z})\right] \\ J(\mathbf{f}, \mathbf{z}) = \frac{1}{2v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \\ \qquad \frac{1}{2v_\xi}\|\mathbf{f} - \mathbf{D}\mathbf{z}\|_2^2 + \\ \qquad \gamma\|\mathbf{z}\|_1 \end{cases}$$

Unsupervized Gaussian case:

$$\begin{cases} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}) \\ p(\mathbf{z}) \propto \exp\left[-\gamma\|\mathbf{z}\|_1\right] \rightarrow \\ p(\gamma) = \mathcal{IG}(\gamma|\alpha_{\gamma 0}, \beta_{\gamma 0}) \\ p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases} \begin{cases} p(\mathbf{f}, \mathbf{z}, \gamma, v_\epsilon, v_\xi|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, \mathbf{z}, \gamma, v_\epsilon, v_\xi)\right] \\ J(\mathbf{f}, \mathbf{z}\ v_\epsilon, v_\xi, \gamma) = \frac{1}{2v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \\ \qquad \frac{1}{2v_\xi}\|\mathbf{f} - \mathbf{D}\mathbf{z}\|_2^2 + \gamma\|\mathbf{z}\|_1 \\ \qquad (\alpha_{\gamma 0} + 1)\ln\gamma + \beta_{\gamma 0}/\gamma \\ \qquad (\alpha_{\epsilon_0} + 1)\ln v_\epsilon + \beta_{\epsilon_0}/v_\epsilon \\ \qquad (\alpha_{\xi_0} + 1)\ln v_\xi + \beta_{\xi_0}/v_\xi \end{cases}$$

Alternate optimization of this criterion gives ADMM like algorithms

Main advantage: direct updates of the hyperparameters

Hierarchical models for more robustness

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon, \\ \mathbf{f} = \mathbf{D}\mathbf{z} + \zeta, \quad \mathbf{z} \text{ sparse Student} \rightarrow \begin{cases} p(z_j|v_{zj}) = \mathcal{N}(z_j|0, v_{zj}), \\ p(v_{zj}) = \mathcal{IG}(v_{zj}|\alpha_{z_0}, \beta_{z_0}) \end{cases} \end{cases}$$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}) \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{V}_z) \rightarrow \\ p(\mathbf{v}_z) = \prod_j \mathcal{IG}(v_{zj}|\alpha_{z_0}, \beta_{z_0}) \\ p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases}
\begin{cases} p(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, v_\epsilon, v_\xi|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, v_\epsilon, v_\xi)\right] \\ J(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, v_\epsilon, v_\xi) = \\ \frac{1}{2v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \frac{1}{2v_\xi}\|\mathbf{f} - \mathbf{D}\mathbf{z}\|_2^2 + \|\mathbf{V}_z^{\frac{-1}{2}}\mathbf{z}\|_2^2 + \\ \sum_j (\alpha_{z_0} + 1) \ln v_{zj} + \beta_{z_0}/v_{zj} \\ (\alpha_{\epsilon_0} + 1) \ln v_\epsilon + \beta_{\epsilon_0}/v_\epsilon \\ (\alpha_{\xi_0} + 1) \ln v_\xi + \beta_{\xi_0}/v_\xi \end{cases}$$

Main advantages:

- Quadratic optimization with respect to $\mathbf{f}$ and $\mathbf{z}$
- Direct updates of the hyperparameters v_ϵ and v_ξ

Non stationary noise and sparsity enforcing prior in the same framework

$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, & \boldsymbol{\epsilon} \text{ non stationary} \rightarrow \begin{cases} p(\epsilon_i|v_{\epsilon_i}) = \mathcal{N}(\epsilon_i|0, v_{\epsilon_i}), \\ p(v_{\epsilon_i}) = \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon 0}, \beta_{\epsilon 0}) \end{cases} \\ \mathbf{f} = \mathbf{D}\mathbf{z} + \zeta, & \mathbf{z} \text{ sparse Student} \rightarrow \begin{cases} p(z_j|v_{z_j}) = \mathcal{N}(z_j|0, v_{z_j}), \\ p(v_{z_j}) = \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \end{cases} \end{cases}$$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, \mathbf{V}_\epsilon) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}) \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{V}_z) \rightarrow \\ p(\mathbf{v}_z) = \prod_j \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \\ p(\mathbf{v}_\epsilon) = \prod_i \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon 0}, \beta_{\epsilon 0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases} \quad \begin{cases} p(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, \mathbf{v}_\epsilon, v_\xi|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, \mathbf{v}_\epsilon, v_\xi)\right] \\ J(\mathbf{f}, \mathbf{z}, \mathbf{v}_z, v_\epsilon, v_\xi) = \|\mathbf{V}_\epsilon^{\frac{-1}{2}}(\mathbf{g} - \mathbf{H}\mathbf{f})\|_2^2 + \\ \qquad \frac{1}{2v_\xi}\|\mathbf{f} - \mathbf{D}\mathbf{z}\|_2^2 + \|\mathbf{V}_z^{\frac{-1}{2}}\mathbf{z}\|_2^2 \\ \qquad \sum_j (\alpha_{z_0} + 1) \ln v_{z_j} + \beta_{z_0}/v_{z_j} \\ \qquad \sum_j (\alpha_{\epsilon_0} + 1) \ln v_{\epsilon_i} + \beta_{\epsilon 0}/v_{\epsilon_i} \\ \qquad (\alpha_{\xi_0} + 1) \ln v_\xi + \beta_{\xi_0}/v_\xi \end{cases}$$

Main advantages:

- Quadratic optimization with respect to **f** and **z**
- Direct updates of the hyperparameters

Variable splitting or How to account for all uncertainties

Standard case:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \rightarrow \begin{cases} \mathbf{g} = \mathbf{g}_0 + \epsilon, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} \end{cases}$$

Error splitting:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} + \epsilon \rightarrow \begin{cases} \mathbf{g} = \mathbf{g}_0 + \epsilon, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \boldsymbol{\xi} \end{cases}$$

or even

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{u} + \boldsymbol{\xi} + \epsilon \rightarrow \begin{cases} \mathbf{g} = \mathbf{g}_0 + \epsilon, \\ \mathbf{g}_0 = \mathbf{H}\mathbf{f} + \mathbf{u} + \boldsymbol{\xi} \end{cases}$$

Error terme variable splitting

$$\left\{\begin{array}{l} p(\mathbf{g}|\mathbf{g}_0, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{g}_0, v_\epsilon \mathbf{I}),\ \ p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon 0}, \beta_{\epsilon 0}), \\ p(\mathbf{g}_0|\mathbf{f}, \mathbf{f}_0, \mathbf{v}_\xi) = \mathcal{N}(\mathbf{g}_0|\mathbf{H}(\mathbf{f}+\mathbf{f}_0), \mathbf{V}_\xi), \\ \mathbf{V}_\xi = \text{diag}\left[\mathbf{v}_\xi\right], \\ p(\mathbf{v}_\xi) = \prod_{i=1}^M p(v_{\xi i}) = \prod_{i=1}^M \mathcal{IG}(v_{\xi i}|\alpha_{\xi 0}, \beta_{\xi 0}), \\ p(\mathbf{f}|\mathbf{v}_f) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{V}_f), \quad \mathbf{V}_f = \text{diag}\left[\mathbf{v}_f\right] \\ p(\mathbf{v}_f) = \prod_{j=1}^N p(v_{f_j}) = \prod_{j=1}^N \mathcal{IG}(v_{f j}|\alpha_{f_0}, \beta_{f_0}), \\ p(\mathbf{f}_0) = \mathcal{N}(\mathbf{f}_0|\mathbf{0}, v_u \mathbf{I}), \end{array}\right.$$

which results in:

$$p(\mathbf{f}, \mathbf{g}_0, \mathbf{f}_0, v_\epsilon, \mathbf{v}_\xi, \mathbf{v}_f|\mathbf{g}) \propto \exp\left[-J(\mathbf{f}, \mathbf{g}_0, \mathbf{f}_0, v_\epsilon, \mathbf{v}_\xi, \mathbf{v}_f)\right] \quad \text{with}$$

$$\begin{array}{l} J(\mathbf{f}, \mathbf{g}_0, \mathbf{f}_0, v_\epsilon, \mathbf{v}_\xi, \mathbf{v}_f, v_u) = \\ \quad \frac{1}{2v_\epsilon}\|\mathbf{g}-\mathbf{g}_0\|_2^2 + \frac{1}{2}\|\mathbf{V}_\xi^{-1/2}(\mathbf{g}_0 - \mathbf{H}(\mathbf{f}+\mathbf{f}_0))\|_2^2 \\ \quad + \frac{1}{2}\|\mathbf{V}_f^{-1/2}\mathbf{f}\|_2^2 + \frac{1}{2v_u}\|\mathbf{f}_0\|_2^2 + (\alpha_{\epsilon_0}+1)\ln v_\epsilon + \frac{\beta_{\epsilon_0}}{v_\epsilon} \\ \quad + \sum_{i=1}^M \left[(\alpha_{\xi_0}+1)\ln v_{\xi_i} + \frac{\beta_{\xi_0}}{v_{\xi_i}}\right] \\ \quad + \sum_{j=1}^N \left[(\alpha_{f_0}+1)\ln v_{f_j} + \frac{\beta_{f_0}}{v_{f_j}}\right] \end{array}$$