

Résolution de problèmes inverses à l'aide de transformées redondantes et de techniques d'optimisation convexe.

Lotfi Chaâri, Nelly Pustelnik,
Caroline Chaux, Jean-Christophe Pesquet

LIGM - Université Paris-Est

Ecole d'été de Peyresq - 22 juillet 2009



Mots-Clés

Résolution de problèmes inverses à l'aide de transformées redondantes et de techniques d'optimisation convexe .

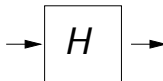
- ▶ restauration/reconstruction d'image
- ▶ minimisation de critères convexes : algorithmes proximaux
- ▶ *a priori* sur les coefficients de transformées redondantes
- ▶ régularisation à l'Analyse / régularisation à la Synthèse

Problème inverse

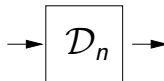


$$\bar{y} \in \mathbb{R}^L$$

image originale



$$H\bar{y} \in \mathbb{R}^M$$



$$z \in \mathbb{R}^M$$

image dégradée

Modèle de dégradation

$$z = \mathcal{D}_n(H\bar{y})$$

Les trames

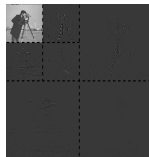
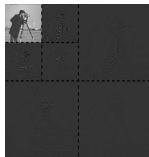
$$y \in \mathbb{R}^L \text{ et } x \in \mathbb{R}^K \text{ (} K \geq L \text{)}$$

F opérateur d'analyse de trame de \mathbb{R}^L vers \mathbb{R}^K

F^* opérateur de synthèse de trame

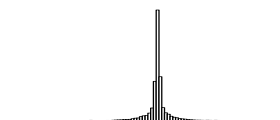
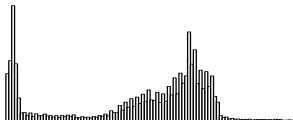
Condition de Trame : $\underline{\mu} \text{Id} \leq F^* \circ F \leq \overline{\mu} \text{Id}$ où $(\underline{\mu}, \overline{\mu}) \in (\mathbb{R}_+^*)^2$



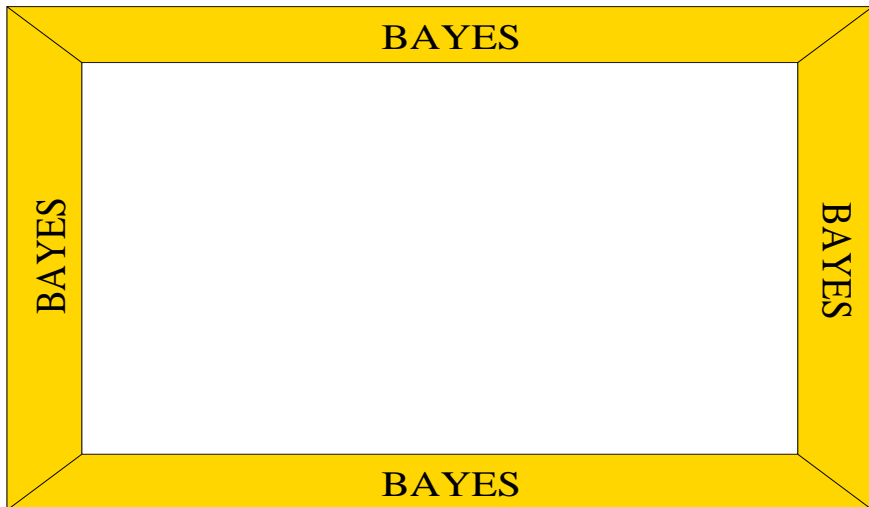
$$\begin{array}{c} \xrightarrow{F} \\ \xleftarrow{F^*} \end{array}$$


$y = F^* x$

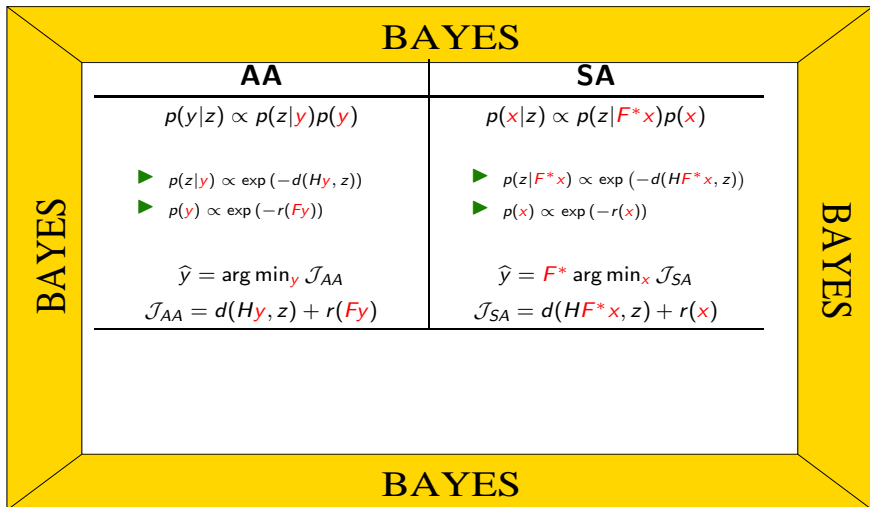
x



Régularisation à l'Analyse (AA)/ à la Synthèse (SA)



Régularisation à l'Analyse (AA)/ à la Synthèse (SA)



Régularisation à l'Analyse (AA)/ à la Synthèse (SA)

BAYES	
AA	SA
$p(y z) \propto p(z y)p(y)$	$p(x z) \propto p(z F^*x)p(x)$
<ul style="list-style-type: none"> ▶ $p(z y) \propto \exp(-d(Hy, z))$ ▶ $p(y) \propto \exp(-r(Fy))$ 	<ul style="list-style-type: none"> ▶ $p(z F^*x) \propto \exp(-d(HF^*x, z))$ ▶ $p(x) \propto \exp(-r(x))$
$\hat{y} = \arg \min_y \mathcal{J}_{AA}$	$\hat{y} = F^* \arg \min_x \mathcal{J}_{SA}$
$\mathcal{J}_{AA} = d(Hy, z) + r(Fy)$	$\mathcal{J}_{SA} = d(HF^*x, z) + r(x)$
<u>Exemple</u>	<u>Exemple</u>
<ul style="list-style-type: none"> ▶ $d(Hy, z) = \frac{1}{2\sigma^2} \ Hy - z\ ^2$ ▶ $r(Fy) = \kappa Fy _p$ 	<ul style="list-style-type: none"> ▶ $d(HF^*x, z) = \frac{1}{2\sigma^2} \ HF^*x - z\ ^2$ ▶ $r(x) = \kappa x _p$
BAYES	

Régularisation à l'Analyse (AA)/ à la Synthèse (SA)

BAYES

AA

$$\hat{y} = \arg \min_y \mathcal{J}_{AA}$$

$$\mathcal{J}_{AA} = d(Hy, z) + r(Fy)$$

SA

$$\hat{y} = F^* \arg \min_x \mathcal{J}_{SA}$$

$$\mathcal{J}_{SA} = d(HF^*x, z) + r(x)$$



* Equivalence entre les deux approches
quand la trame est une base orthogonale

* Comparaison des deux approches

[Elad *et al.* (2007), Carlavan *et al.* (2009)]

BAYES

BAYES

BAYES

Solution au problème de minimisation

AA	SA
$\hat{y} = \arg \min_y \mathcal{J}_{AA}$ $\mathcal{J}_{AA} = \underbrace{d(Hy, z)}_{(1)} + \underbrace{r(Fy)}_{(2)}$	$\hat{y} = F^* \arg \min_x \mathcal{J}_{SA}$ $\mathcal{J}_{SA} = \underbrace{d(HF^*x, z)}_{(1)} + \underbrace{r(x)}_{(2)}$

Critère d'optimalité :

- ▶ (1) : partie convexe et différentiable de gradient β -Lipschitz

(AA : $\beta = \alpha \|H\|^2$ et SA : $\beta = \alpha \|HF^*\|^2$)

- ▶ Soit $f(\cdot) = d(\cdot, z)$ une fonction de gradient α -Lipschitz
- ▶ Soit $f_S = f \circ H \circ F^*$, alors $\nabla f_S = FH^*(\nabla f(HF^* \cdot))$ et

$$(\forall (x_1, x_2) \in (\mathbb{R}^K)^2) \quad \|\nabla f_S(x_1) - \nabla f_S(x_2)\| \leq \alpha \|HF^*\|^2 \|x_1 - x_2\|.$$

Solution au problème de minimisation

AA	SA
$\hat{y} = \arg \min_y \mathcal{J}_{AA}$ $\mathcal{J}_{AA} = \underbrace{d(Hy, z)}_{(1)} + \underbrace{r(Fy)}_{(2)}$	$\hat{y} = F^* \arg \min_x \mathcal{J}_{SA}$ $\mathcal{J}_{SA} = \underbrace{d(HF^*x, z)}_{(1)} + \underbrace{r(x)}_{(2)}$

Critère d'optimalité :

- ▶ (1) : partie convexe et différentiable de gradient β -Lipschitz
(AA : $\beta = \alpha\|H\|^2$ et SA : $\beta = \alpha\|HF^*\|^2$)
- ▶ (2) : partie convexe non nécessairement différentiable

Solution au problème de minimisation

AA	SA
$\hat{y} = \arg \min_y \mathcal{J}_{AA}$ $\mathcal{J}_{AA} = \underbrace{d(Hy, z)}_{(1)} + \underbrace{r(Fy)}_{(2)}$	$\hat{y} = F^* \arg \min_x \mathcal{J}_{SA}$ $\mathcal{J}_{SA} = \underbrace{d(HF^*x, z)}_{(1)} + \underbrace{r(x)}_{(2)}$

Critère d'optimalité :

- ▶ (1) : partie convexe et différentiable de gradient β -Lipschitz
(AA : $\beta = \alpha \|H\|^2$ et SA : $\beta = \alpha \|HF^*\|^2$)
- ▶ (2) : partie convexe non nécessairement différentiable

⇒ **Difficile** d'utiliser des algorithmes classiques d'optimisation

⇒ Algorithmes itératifs d'optimisation convexe mettant en œuvre les **opérateurs proximaux** [Daubechies *et al.* (2004), Combettes et Wajs (2005), Chaux *et al.* (2007), Combettes et Pesquet (2007), Bauschke et Combettes (2008), Combettes et Pesquet (2008)]

L'opérateur proximal

Définition [Moreau (1965)]

* $\Gamma_0(\mathcal{H})$: classe des fonctions propres semi-continues inférieurement de l'espace séparable de Hilbert \mathcal{H} vers $] -\infty, +\infty]$

* $\forall x \in \mathcal{H}$, la fonction $y \mapsto f(y) + \|x - y\|^2/2$ où $f \in \Gamma_0(\mathcal{H})$ atteint son minimum en un **point unique** noté $\text{prox}_f x$.

$(\forall x \in \mathcal{H}),$

$$\text{prox}_f x = \arg \min_{y \in \mathcal{H}} f(y) + \frac{1}{2} \|y - x\|^2$$

L'opérateur proximal

Définition [Moreau (1965)]

* $\Gamma_0(\mathcal{H})$: classe des fonctions propres semi-continues inférieurement de l'espace séparable de Hilbert \mathcal{H} vers $] -\infty, +\infty]$

* $\forall x \in \mathcal{H}$, la fonction $y \mapsto f(y) + \|x - y\|^2/2$ où $f \in \Gamma_0(\mathcal{H})$ atteint son minimum en un **point unique** noté $\text{prox}_f x$.

$$(\forall x \in \mathcal{H}), \quad \text{prox}_f x = \arg \min_{y \in \mathcal{H}} f(y) + \frac{1}{2} \|y - x\|^2$$

Exemples :

- ▶ Si $f = \iota_C \Rightarrow \text{prox}_{\iota_C} = P_C$ où P_C représente la projection sur un convexe fermé C .
- ▶ Si $f = |\cdot|_1 \Rightarrow \text{prox}_{|\cdot|_1}$ correspond à un seuillage doux
- ▶ Si $f = \omega |\cdot|_p \Rightarrow$ **Forme explicite**

Algorithme forward-backward

[Daubechies et al.(2004), Combettes et Wajs (2005)]

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>

Algorithme forward-backward

[Daubechies et al.(2004), Combettes et Wajs (2005)]

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>
<p>1: Choisir $y_0 \in \mathbb{R}^L$</p>	<p>1: Choisir $x_0 \in \mathbb{R}^K$</p>

Algorithme forward-backward

[Daubechies et al.(2004), Combettes et Wajs (2005)]

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>
<ol style="list-style-type: none"> 1: Choisir $y_0 \in \mathbb{R}^L$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ H\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $y_{n+\frac{1}{2}} = y_n - \gamma_n \nabla f_A(y_n)$ 6: $p_n = \text{prox}_{\gamma_n r \circ F}(y_{n+\frac{1}{2}})$ 7: $y_{n+1} = y_n + \lambda_n (p_n - y_n)$ 8: end for 	<ol style="list-style-type: none"> 1: Choisir $x_0 \in \mathbb{R}^K$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ HF^*\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $x_{n+\frac{1}{2}} = x_n - \gamma_n \nabla f_S(x_n)$ 6: $p_n = \text{prox}_{\gamma_n r}(x_{n+\frac{1}{2}})$ 7: $x_{n+1} = x_n + \lambda_n (p_n - x_n)$ 8: end for

Algorithme forward-backward

[Daubechies et al.(2004), Combettes et Wajs (2005)]

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>
<ol style="list-style-type: none"> 1: Choisir $y_0 \in \mathbb{R}^L$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ H\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $y_{n+\frac{1}{2}} = y_n - \gamma_n \nabla f_A(y_n)$ 6: $p_n = \text{prox}_{\gamma_n r \circ F}(y_{n+\frac{1}{2}})$ 7: $y_{n+1} = y_n + \lambda_n (p_n - y_n)$ 8: end for 9: Après convergence, on a $\hat{y} = y_{n+1}$. 	<ol style="list-style-type: none"> 1: Choisir $x_0 \in \mathbb{R}^K$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ HF^*\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $x_{n+\frac{1}{2}} = x_n - \gamma_n \nabla f_S(x_n)$ 6: $p_n = \text{prox}_{\gamma_n r}(x_{n+\frac{1}{2}})$ 7: $x_{n+1} = x_n + \lambda_n (p_n - x_n)$ 8: end for 9: Après convergence, on a $\hat{y} = F^* x_{n+1}$.

Algorithme forward-backward

[Daubechies et al.(2004), Combettes et Wajs (2005)]

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>
<ol style="list-style-type: none"> 1: Choisir $y_0 \in \mathbb{R}^L$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ H\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $y_{n+\frac{1}{2}} = y_n - \gamma_n \nabla f_A(y_n)$ 6: $p_n = \text{prox}_{\gamma_n r \circ F}(y_{n+\frac{1}{2}})$ 7: $y_{n+1} = y_n + \lambda_n (p_n - y_n)$ 8: end for 9: Après convergence, on a $\hat{y} = y_{n+1}$. 	<ol style="list-style-type: none"> 1: Choisir $x_0 \in \mathbb{R}^K$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ HF^*\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $x_{n+\frac{1}{2}} = x_n - \gamma_n \nabla f_S(x_n)$ 6: $p_n = \text{prox}_{\gamma_n r}(x_{n+\frac{1}{2}})$ 7: $x_{n+1} = x_n + \lambda_n (p_n - x_n)$ 8: end for 9: Après convergence, on a $\hat{y} = F^* x_{n+1}$.

Calcul de $\|HF^*\|^2$: Algorithme

- ▶ Le calcul de $\|HF^*\|^2$ n'est pas toujours explicite, on propose donc un **algorithme itératif** permettant de calculer cette norme.
- ▶ L'algorithme est basé sur le raisonnement suivant :
 - ▶ Soit $B = HF^*$

Calcul de $\|HF^*\|^2$: Algorithme

- ▶ Le calcul de $\|HF^*\|^2$ n'est pas toujours explicite, on propose donc un **algorithme itératif** permettant de calculer cette norme.
- ▶ L'algorithme est basé sur le raisonnement suivant :
 - ▶ Soit $B = HF^*$
 - ▶ On effectue une **décomposition en valeurs propres** de B^*B , telle que $B^*B = U\Lambda U^*$ où $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_K\}$ et $U = [u_1, \dots, u_K] \in \mathbb{R}^{K \times K}$ est une matrice orthogonale.

Calcul de $\|HF^*\|^2$: Algorithme

- ▶ Le calcul de $\|HF^*\|^2$ n'est pas toujours explicite, on propose donc un **algorithme itératif** permettant de calculer cette norme.
- ▶ L'algorithme est basé sur le raisonnement suivant :
 - ▶ Soit $B = HF^*$
 - ▶ On effectue une **décomposition en valeurs propres** de B^*B , telle que $B^*B = U\Lambda U^*$ où $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_K\}$ et $U = [u_1, \dots, u_K] \in \mathbb{R}^{K \times K}$ est une matrice orthogonale.
 - ▶ On sait que $\|HF^*\|^2 = \lambda_{i_0}$ avec $i_0 \in \arg \max_{1 \leq k \leq K} \lambda_k$.

Calcul de $\|HF^*\|^2$: Algorithme

- ▶ Le calcul de $\|HF^*\|^2$ n'est pas toujours explicite, on propose donc un **algorithme itératif** permettant de calculer cette norme.
- ▶ L'algorithme est basé sur le raisonnement suivant :
 - ▶ Soit $B = HF^*$
 - ▶ On effectue une **décomposition en valeurs propres de B^*B** , telle que $B^*B = U\Lambda U^*$ où $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_K\}$ et $U = [u_1, \dots, u_K] \in \mathbb{R}^{K \times K}$ est une matrice orthogonale.
 - ▶ On sait que $\|HF^*\|^2 = \lambda_{i_0}$ avec $i_0 \in \arg \max_{1 \leq k \leq K} \lambda_k$.
 - ▶ On peut montrer que

$$\lim_{n \rightarrow +\infty} \frac{\|B^n x_0\|^2}{\|B^{n-1} x_0\|^2} = \lambda_{i_0} = \|HF^*\|^2$$

Calcul de $\|HF^*\|^2$: Algorithme



$$\lim_{n \rightarrow +\infty} \frac{\|B^n x_0\|^2}{\|B^{n-1} x_0\|^2} = \lambda_{i_0} = \|HF^*\|^2$$

Algorithme

Choisir aléatoirement $x_0 \in \mathbb{R}^K$, et soit $\rho_0 = \epsilon + 1$, $n = 1$ et $\rho_n = 1$

while $\frac{|\rho_n - \rho_{n-1}|}{\rho_n} \leq \epsilon$ **do**

 Soit $x_n = B^* B x_{n-1}$ où $B = HF^*$

 Soit $\rho_n = \frac{\|x_n\|}{\|x_{n-1}\|}$

end while

Après convergence, on a $\|HF^*\|^2 = \rho_n$

Calcul de $\text{prox}_{r \circ F}$: Forme explicite

AA	SA
$\hat{y} = \arg \min_y f_A(y) + r(Fy)$ <p>avec $f_A(y) = d(Hy, z)$</p>	$\hat{y} = F^* \arg \min_x [f_S(x) + r(x)]$ <p>$f_S(x) = d(HF^*x, z)$</p>
<ol style="list-style-type: none"> 1: Choisir $y_0 \in \mathbb{R}^L$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ H\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $y_{n+\frac{1}{2}} = y_n - \gamma_n \nabla f_A(y_n)$ 6: $p_n = \text{prox}_{\gamma_n r \circ F}(y_{n+\frac{1}{2}})$ 7: $y_{n+1} = y_n + \lambda_n (p_n - y_n)$ 8: end for 9: Après convergence, on a $\hat{y} = y_{n+1}$. 	<ol style="list-style-type: none"> 1: Choisir $x_0 \in \mathbb{R}^K$ 2: for $n \in \mathbb{N}$ do 3: Soit $\gamma_n \in]0, 2/(\alpha \ HF^*\ ^2)[$ 4: Soit $\lambda_n \in]0, 1]$ 5: $x_{n+\frac{1}{2}} = x_n - \gamma_n \nabla f_S(x_n)$ 6: $p_n = \text{prox}_{\gamma_n r}(x_{n+\frac{1}{2}})$ 7: $x_{n+1} = x_n + \lambda_n (p_n - x_n)$ 8: end for 9: Après convergence, on a $\hat{y} = F^* x_{n+1}$.

Calcul de $\text{prox}_{r \circ F}$: Forme explicite

Proposition [Combettes et Pesquet 2007]

Soit \mathcal{G} un espace d'Hilbert, soit $\varphi \in \Gamma_0(\mathcal{G})$, et $T: \mathcal{H} \rightarrow \mathcal{G}$ un opérateur linéaire borné. Supposons que la composition de T et T^* satisfasse $T \circ T^* = \chi \text{Id}$, pour $\chi \in]0, +\infty[$. Alors $\varphi \circ T \in \Gamma_0(\mathcal{H})$ et

$$\text{prox}_{\varphi \circ T} = \text{Id} + \frac{T^*}{\chi} \circ (\text{prox}_{\chi\varphi} - \text{Id}) \circ T.$$

Dans notre application :

- ▶ Si $\varphi = r$, $T = F$ et $\chi = 1$ (base orthogonale) \rightarrow ok
- ▶ Si $\varphi = r$, $T = F^*$ et $\chi = \mu$ (trame ajustée) \rightarrow ok
- ▶ Si $\varphi = r$, $T = F$ même si la trame est ajustée, la proposition n'est pas applicable.

Calcul de $\text{prox}_{r \circ F}$: Algorithme de décomposition Dual-Primal

- Par **définition de l'opérateur proximal**, on peut écrire :

$$\text{prox}_{r \circ F}(y) = \min_{p \in \mathbb{R}^L} \frac{1}{2} \|p - y\|^2 + r(Fp) \quad (1)$$

- Le **problème dual** associé à l'équation (1) est :

$$\min_{v \in \mathbb{R}^K} \frac{1}{2} \|y - F^*v\|^2 + r^*(v) \quad (2)$$

Calcul de $\text{prox}_{r \circ F}$: Algorithme de décomposition Dual-Primal

- ▶ Définition : $\text{prox}_{r \circ F}(y) = \min_{p \in \mathbb{R}^L} \frac{1}{2} \|p - y\|^2 + r(Fp)$
- ▶ Dualisation : $\min_{v \in \mathbb{R}^K} \frac{1}{2} \|y - F^*v\|^2 + r^*(v)$

Algorithme de décomposition Dual-Primal [Combettes *et al.* 2009]

- 1: Choisir $\epsilon \in]0, \min\{1, 1/\bar{\mu}\}[$ et soit $v_0 \in \mathbb{R}^K$
- 2: **for** $n \in \mathbb{N}$ **do**
- 3: $p_n = y - F^*v_n$
- 4: Soit $\gamma_n \in [\epsilon, 2/\bar{\mu} - \epsilon]$ et $\lambda_n \in [\epsilon, 1]$
- 5: $v_{n+1} = v_n + \lambda_n(\gamma_n F p_n - \gamma_n \text{prox}_{\frac{1}{\gamma_n} r}(v_n + F p_n))$
- 6: **end for**

Convergence [Combettes *et al.* 2009]

La suite $(v_n)_{n \in \mathbb{N}}$ converge vers une solution v du problème dual, et $\text{prox}_{r \circ F}(y) = y - F^*v$.

Comparaison AA vs. SA : résultats

- ▶ Problématique : restauration d'une image dégradée par un **flou uniforme** et un **bruit additif Gaussien**
- ▶ Critères à minimiser :
 - ▶ AA : $\hat{y} = \arg \min_y \frac{1}{2\sigma^2} \|Hy - z\|^2 + \kappa |Fy|_1$.
 - ▶ SA : $\hat{y} = F^* \arg \min_x \left[\frac{1}{2\sigma^2} \|HF^*x - z\|^2 + \kappa |x|_1 \right]$.
- ▶ Algorithme :
 - ▶ AA : forward-backward ($\beta = \frac{1}{\sigma^2} \|H\|^2$) avec boucle interne de l'algorithme de décomposition Dual-Primal pour calculer $\text{prox}_{\kappa|F \cdot|_1}$
 - ▶ SA : forward-backward ($\beta = \frac{1}{\sigma^2} \|HF^*\|^2$)
- ▶ Choix de la trame : Ondelettes non décimées ou GenLOT [Gauthier *et al.* (2009)]

Comparaison AA vs. SA : résultats

SNR = 16.60 dB

Image originale



Image dégradée



Image restaurée avec AA

Image restaurée avec SA

SNR = 19.23 dB

SNR = 19.03 dB

Comparaison AA vs. SA : résultats

SNR = 17.47 dB

Image originale



Image dégradée

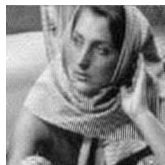


Image restaurée avec AA

Image restaurée avec SA

SNR = 19.92 dB

SNR = 19.93 dB

Comparaison AA vs. SA

- ▶ Suivant le type de trame et d'image considéré, une approche peut être meilleure que l'autre en SNR mais une règle générale ne peut être établie.
- ▶ L'algorithme associé à SA reste néanmoins plus simple à implémenter (pas de boucle interne).

Comparaison AA vs. SA

- ▶ Suivant le type de trame et d'image considéré, une approche peut être meilleure que l'autre en SNR mais une règle générale ne peut être établie.
- ▶ L'algorithme associé à SA reste néanmoins plus simple à implémenter (pas de boucle interne).
- ▶ Autre approche que l'algorithme de décomposition Dual-Primal dans le cas d'union de bases orthogonales.
 - ▶ Utilisation de l'algorithme PPRA [Combettes et Pesquet (2008)]
 - ▶ Algorithme permettant de minimiser une somme de J fonctions
 - ▶ Nécessité la connaissance des opérateurs proximaux associés à chacune des J fonctionnelles

Algorithme Parallèle PRoximal (PPRA)

Algorithme PPRA [Combettes et Pesquet (2008)]

- 1: Soit $\gamma \in]0, +\infty[$
- 2: Soit $(\omega_j)_{1 \leq j \leq J} \in]0, 1]^J$ telle que $\sum_{j=1}^J \omega_j = 1$.
- 3: Soit $(u_{j,0})_{1 \leq j \leq J} \in \mathcal{H}^J$ et $\xi_0 = \sum_{j=1}^J \omega_j u_{j,0}$.
- 4: **for** $n \in \mathbb{N}$ **do**
- 5: **for** $j = 1, \dots, J$ **do**
- 6: $p_{j,n} = \text{prox}_{\gamma/\omega_j f_j} u_{j,n}$
- 7: **end for**
- 8: $p_n = \sum_{j=1}^J \omega_j p_{j,n}$
- 9: $\lambda_n \in]0, 2[$
- 10: **for** $j = 1, \dots, J$ **do**
- 11: $u_{j,n+1} = u_{j,n} + \lambda_n (2 p_n - \xi_n - p_{j,n})$
- 12: **end for**
- 13: $\xi_{n+1} = \xi_n + \lambda_n (p_n - \xi_n)$
- 14: **end for**

Convergence [Combettes et Pesquet (2008)]

Sous certaines hypothèses, la suite $(\xi_n)_{n \in \mathbb{N}}$ converge vers le minimiseur de $\sum_{j=1}^J f_j$.

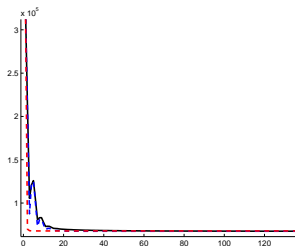
Algorithme Parallèle PProximal (PPRA)

- ▶ Dans le cas de la restauration d'une image dégradée par un **bruit additif Gaussien** et un **flou uniforme**

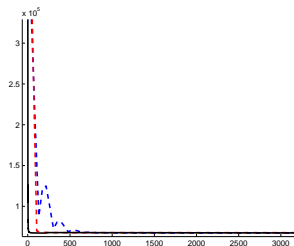
$$\hat{y} = \arg \min_y \frac{1}{2\sigma^2} \|Hy - z\|^2 + \kappa |Fy|_1. \quad (1)$$

- ▶ Calcul de l'opérateur proximal associé à $\frac{1}{2\sigma^2} \|H \cdot - z\|^2$
 - ok [Combettes et Pesquet (2008)]
- ▶ Calcul de l'opérateur proximal associé à $\kappa |Fy|_1$
 - forme explicite si F est associé à une base orthogonale.
 - calcul itératif en passant par le dual.
 - **forme explicite dans le cas où F correspond à une union de μ bases orthogonales**, dans ce cas $\kappa |Fy|_1 = \kappa \sum_{i=1}^{\mu} |F_i y|_1$ où $F_i^* \circ F_i = \text{Id}$. Par conséquent on sait calculer les μ opérateurs proximaux associés à $\kappa |F_i \cdot|_1$.

Algorithme Parallèle PRoximal (PPRA) : vitesse de convergence



Nombre d'itérations



Temps CPU

FIGURE: Comparaison des temps de convergence pour AA et union de 2 bases. FB(Algorithme de décomposition Dual-Primal) en rouge, PPRA(Algorithme de décomposition Dual-Primal) en bleu et PPRA en noir

Conclusion

- ▶ Possibilité de gérer les transformations redondantes à l'Analyse et à la Synthèse avec les algorithmes proximaux
- ▶ Dans le cas où le critère est composé d'un terme d'attache aux données et d'un terme de régularisation :
 - ▶ AA est plus pénalisante en temps de calcul sauf dans le cas où la trame considérée est une union de bases orthogonales.
 - ▶ Pas de règle générale sur la performance qualitative d'une approche par rapport à l'autre.
- ▶ Dans le cas où le critère est composé de plus d'un terme de régularisation : par exemple $\ell_1 + TV$, l'approche SA sera limitée aux trames ajustées contrairement à AA qui pourra gérer tout type de trames.

Merci de votre attention