

Comparaison entre les algorithmes semi-quadratiques et gradients conjugués préconditionnés pour la restauration d'image préservant les bords

Christian LABAT, Jérôme IDIER

IRCCyN-CNRS UMR 6597
ECN, 1 rue de la Noë, BP 92101, 44321 Nantes Cedex 3, France
prénom.nom@irccyn.ec-nantes.fr

Résumé – Notre contribution concerne la restauration d'image. L'approche adoptée consiste à minimiser un critère des moindres carrés pénalisés. Ici, nous nous intéressons à la recherche d'algorithmes efficaces pour mener une telle tâche. Plus précisément, nous étudions des liens entre algorithmes semi-quadratiques (SQ) et gradients conjugués préconditionnés (GCP) pour la minimisation de critères pénalisés. Nous proposons un théorème de convergence pour l'algorithme GCP associé à une recherche de pas spécifique. Enfin, nous menons une comparaison expérimentale entre algorithmes SQ et GCP dans le cadre de la déconvolution montrant la grande efficacité de ce dernier.

Abstract – Our contribution deals with image restoration. The adopted approach consists in minimizing a criterion of penalized least squares. Here, we are interested in research of effective algorithms to carry out such a task. More precisely, we study links between half-quadratic (HQ) algorithms and preconditioned conjugate gradient (PCG) algorithm for the minimization of penalized criteria. We propose a theorem of convergence for the PCG algorithm associated with a specific line search. Lastly, we carry out an experimental comparison between algorithms HQ and PCG within the framework of deconvolution showing the great effectiveness of this last.

1 Introduction

Notre contribution concerne la restauration d'image où une image solution $\hat{\mathbf{x}} \in \mathbb{R}^N$ est estimée à partir d'une observation bruitée $\mathbf{y} \in \mathbb{R}^M$. Suivant [1], nous proposons de minimiser un critère des moindres carrés pénalisé avec un terme préservant les bords de l'image:

$$\hat{\mathbf{x}} = \min_{\mathbf{x} \in \mathbb{R}^N} \mathcal{J}(\mathbf{x}) \quad (1)$$

$$\mathcal{J}(\mathbf{x}) = \mathcal{Q}(\mathbf{x}) + \lambda \Phi(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^N \quad (2)$$

où $\mathcal{Q}(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2$, avec $\mathbf{A} \in \mathbb{R}^{M \times N}$ la matrice d'observation, λ un hyperparamètre de régularisation et

$$\Phi(\mathbf{x}) = \sum_{c \in \mathcal{C}} \phi(\mathbf{v}_c^t \mathbf{x}),$$

avec \mathcal{C} ($|\mathcal{C}| = C$) un ensemble de cliques sur \mathcal{S} ($|\mathcal{S}| = N$), chaque vecteur $\mathbf{v}_c \in \mathbb{R}^N$ ayant un support restreint à la clique c . Pour illustrer, \mathcal{C} peut correspondre à l'ensemble des paires de pixels voisins $c = \{r, s\}$, avec $\mathbf{v}_c^t \mathbf{x} = x_r - x_s$. La fonction potentiel $\phi : \mathbb{R} \mapsto \mathbb{R}$ peut être C^2 convexe et issue du domaine de l'estimation robuste préservant les bords, c'est-à-dire qu'elle a un comportement affine vers l'infini [2]. Le critère \mathcal{J} est alors lui-même convexe, ce qui assure qu'il n'y a pas de multiples minima locaux. Le coût de calcul d'un tel estimateur peut être un facteur limitant l'utilisation de ce type de méthode, c'est pourquoi nous nous focalisons ici sur la recherche d'algorithmes rapides pour minimiser \mathcal{J} .

Soit $\mathbf{Q}_0 = 2\mathbf{A}^t \mathbf{A}$ et $\mathbf{V} = [\mathbf{v}_1^t | \dots | \mathbf{v}_C^t]$. Dans la suite, on suppose que la matrice $\mathbf{Q}_0 + \lambda \mathbf{V}^t \mathbf{V}$ est inversible.

2 Les algorithmes semi-quadratiques

2.1 Formulation "gradient-reliée"

La minimisation de (1) peut être conduite selon une approche semi-quadratique (SQ), ce qui fournit alors les constructions de Geman & Reynolds (GR) et Geman & Yang (GY) [1]. Dans [3], il est démontré que les algorithmes SQ GR et GY s'identifient à des méthodes *gradient-relieés* à pas constant unitaire.

La forme GR se définit par

$$\mathbf{d}_{\text{GR}}^k = -(\mathbf{B}_{\text{GR}}(\mathbf{x}^k))^{-1} \nabla \mathcal{J}(\mathbf{x}^k) \quad (\text{direction GR}) \quad (3)$$

$$\mathbf{x}^k = \mathbf{x}^k + \mathbf{d}_{\text{GR}}^k \quad (\text{mise à jour GR})$$

où l'opérateur normal $\mathbf{B}_{\text{GR}}(\mathbf{u})$ est définie par

$$\mathbf{b}(\mathbf{u}) := \text{Vect} \begin{bmatrix} \phi'(|[\mathbf{V}\mathbf{u}]_c|) \\ |[\mathbf{V}\mathbf{u}]_c| \end{bmatrix},$$

$$\mathbf{B}_{\text{GR}}(\mathbf{u}) := \mathbf{Q}_0 + \lambda \mathbf{V}^t \text{Diag}\{\mathbf{b}(\mathbf{u})\} \mathbf{V},$$

et la forme GY se définit par

$$\mathbf{d}_{\text{GY}}^k = -(\mathbf{B}_{\text{GY}}^a)^{-1} \nabla \mathcal{J}(\mathbf{x}^k) \quad (\text{direction GY}) \quad (4)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}_{\text{GY}}^k \quad (\text{mise à jour GY})$$

où $a = 1/\phi''(0)$ et la matrice normale \mathbf{B}_{GY}^a est définie par

$$\mathbf{B}_{\text{GY}}^a := \mathbf{Q}_0 + \frac{\lambda}{a} \mathbf{V}^t \mathbf{V}.$$

Des résultats de convergence de ces algorithmes SQ se trouvent notamment dans [1].

2.2 Inversion des matrices normales semi-quadratiques

Le calcul exact de la direction de descente des algorithmes SQ (3)-(4) est généralement trop coûteux. [2, 4, 5] proposent

une version de cette direction de descente approchée par gradients conjugués (GC) ou GC préconditionnés (GCP) tronqués, dans le sens où un nombre de sous-itérations plus faible que la dimension de l'espace solution est utilisé. Ce type d'implémentation a pour avantage d'être efficace, mais au prix de la perte des résultats de convergence associés aux calculs des directions de descentes SQ exactes.

La partie suivante présente en quelque sorte une version duale de cette implémentation: au lieu d'utiliser les GCP au sein des SQ, nous étudions une forme d'algorithme GCP non linéaires où se retrouve une version scalaire des SQ tronqués.

3 L'algorithme des gradients conjugués préconditionnés non linéaire

3.1 La forme préconditionnée de Polak-Ribiere

La méthode des gradients conjugués (GC) est une mise à jour itérative de la forme $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$, où les directions de descentes $\{\mathbf{d}^k\}$ sont mutuellement conjuguées dans le cas d'un problème purement quadratique. Suivant [6], nous nous limitons ici à la forme de Polak-Ribiere des GC préconditionnée (PRGCP) [7]:

$$\begin{aligned} \mathbf{p}^k &= -(\mathbf{M}^k)^{-1} \nabla \mathcal{J}(\mathbf{x}^k) \quad (\text{préconditionnement}) \quad (5) \\ \beta_k &= \begin{cases} 0 & \text{si } k = 0 \\ \frac{(\nabla \mathcal{J}(\mathbf{x}^k) - \nabla \mathcal{J}(\mathbf{x}^{k-1}))^t \mathbf{p}^k}{(\nabla \mathcal{J}(\mathbf{x}^{k-1}))^t \mathbf{p}^{k-1}} & \text{si } k > 0 \end{cases} \\ \mathbf{d}^k &= \mathbf{p}^k + \beta_k \mathbf{d}^{k-1} \quad (\text{direction PRGCP}) \\ \mathbf{x}^{k+1} &= \mathbf{x}^k + \alpha_k \mathbf{d}^k \quad (\text{recherche du pas et mise à jour}) \end{aligned}$$

L'algorithme PRGCP possède deux degrés de liberté: le pas α_k , et la matrice de préconditionnement définie positive \mathbf{M}^k .

3.2 Recherche du pas avec les formes scalaires des algorithmes semi-quadratiques

Contrairement aux algorithmes SQ, les méthodes GC requièrent usuellement une recherche du pas non triviale afin de déterminer un pas α_k admissible. Dans [6], une procédure itérative de recherche du pas est présentée. Elle a pour avantage d'être simple et convergente dans le sens où la suite générée $\alpha_k^1, \dots, \alpha_k^i$ converge vers un minimum local de $f(\alpha) = \mathcal{J}(\mathbf{x}^k + \alpha \mathbf{d}^k)$. Cette procédure s'identifie en fait à la version scalaire (1D) de l'algorithme GR. Dans le même esprit, une alternative utilisant l'algorithme GY scalaire peut être envisagée. En considérant la formulation 1D gradient-reliée de ces algorithmes SQ appliqués à la minimisation de $f(\alpha)$, ces algorithmes s'écrivent

$$\begin{cases} \alpha_k^0 = 0 \\ \alpha_k^{i+1} = \alpha_k^i - \frac{(\mathbf{d}^k)^t \nabla \mathcal{J}(\mathbf{x}^k + \alpha_k^i \mathbf{d}^k)}{(\mathbf{d}^k)^t \mathbf{Q}_k^i \mathbf{d}^k}, \quad 0 \leq i \leq I-1 \\ \alpha_k = \alpha_k^I \end{cases} \quad (6)$$

avec $\mathbf{Q}_k^i = \mathbf{B}_{\text{GY}}^a$ pour GY1D et avec $\mathbf{Q}_k^i = \mathbf{B}_{\text{GR}}(\mathbf{x}^k + \alpha_k^i \mathbf{d}^k)$ pour GR1D. Lorsque $I = 1$, la formule obtenue est la même que celle proposée dans [8], pour une vaste famille de méthodes des gradients conjugués.

3.3 Théorème de convergence

Le théorème suivant concerne la convergence de l'algorithme PRGCP avec la recherche du pas (6). La preuve se trouve dans [9].

Théorème 1. *Supposons que*

$$\begin{cases} \phi : \mathbb{R} \mapsto \mathbb{R} \text{ est } \mathcal{C}^2, \text{ coercive et paire} \\ \forall t > 0, \phi'(t) > 0 \\ \psi(t) = \phi(\sqrt{t}) \text{ est concave pour } t > 0 \\ \sup_{t \in \mathbb{R}} |\phi''(t)| < \infty \end{cases}$$

Alors, la forme de Polak-Ribiere préconditionnée par une suite de matrices constantes $\mathbf{M}^k = \mathbf{M}_0$ est convergente dans le sens où

$$\liminf_{k \rightarrow \infty} \|\nabla \mathcal{J}(\mathbf{x}^k)\| = 0$$

lorsque la recherche du pas (6) est employée avec $\mathbf{Q}_k^i = \mathbf{B}_{\text{GY}}^a$ ou bien $\mathbf{Q}_k^i = \mathbf{B}_{\text{GR}}(\mathbf{x}^k + \alpha_k^i \mathbf{d}^k)$.

4 Liens entre algorithmes

4.1 Comparaison des pas

Si on choisit $\mathbf{Q}^k = \mathbf{M}^k$ et si on utilise une stratégie sans conjugaison, alors la formule du pas (6) avec $I = 1$ se simplifie en $\alpha_k = 1$. Les algorithmes GY et GR correspondent alors aux choix particuliers des matrices normales SQ pour \mathbf{M}^k . Par conséquent, la formule du pas (6) peut être interprétée comme une généralisation naturelle du pas constant lorsqu'on passe des algorithmes gradients-reliés aux algorithmes des gradients conjugués.

4.2 Intérêt du choix de la matrice de préconditionnement

La matrice de préconditionnement des algorithmes semi-quadratiques est imposée ($\mathbf{B}_{\text{GR}}(\mathbf{x}^k)$ et \mathbf{B}_{GY}^a), ce qui entraîne en général des difficultés d'inversion. Par contre, l'algorithme PRGCP permet de choisir une matrice de préconditionnement définie positive quelconque et constante (afin que le théorème 1 s'applique). Ainsi, il est possible d'utiliser une matrice de type circulante bloc circulante (CBC) comme préconditionneur lorsque le Hessien de \mathcal{J} est approximativement CBC, sous l'hypothèse classique de régularité de l'image recherchée. On peut alors tirer profit du fait que les matrices CBC se diagonalisent facilement dans la base de Fourier.

5 Comparaison expérimentale de la vitesse de convergence

5.1 Les problèmes traités

Nous considérons comme problème de restauration celui de la déconvolution en présence de bruit. Deux cas sont étudiés: convolution à partir d'un noyau gaussien (écart-type 2.24, taille 17×17) avec l'hypothèse de bords zéros à l'extérieur (Dirichlet), et débruitage qui n'est qu'un cas limite de déconvolution. L'image d'origine, de taille $N = 512 \times 512$ à valeur dans $[0, 255]$, est la même pour ces deux problèmes.

La pénalisation retenue utilise pour \mathcal{C} l'ensemble des paires de pixels voisins et pour potentiel la fonction convexe hyperbolique $\phi(t) = \sqrt{\delta^2 + t^2}$, où δ correspond à un seuil entre le régime quadratique et affine de ϕ [1].

Pour chacun des deux problèmes, le critère \mathcal{J} convexe ainsi défini dépend des hyperparamètres λ et δ : nous avons choisi de les régler de manière à avoir l'image restaurée la plus proche visuellement de l'image d'origine. Ainsi, nous avons retenu $\delta = 13$ pour les deux problèmes, $\lambda = 10$ pour le débruitage et $\lambda = 0.2$ pour la déconvolution.

Finalement, nous avons deux critères à minimiser pour lesquels le même critère d'arrêt global est utilisé

$$\eta = \|\nabla \mathcal{J}(\mathbf{x}^k)\|/N < 10^{-6}. \quad (7)$$

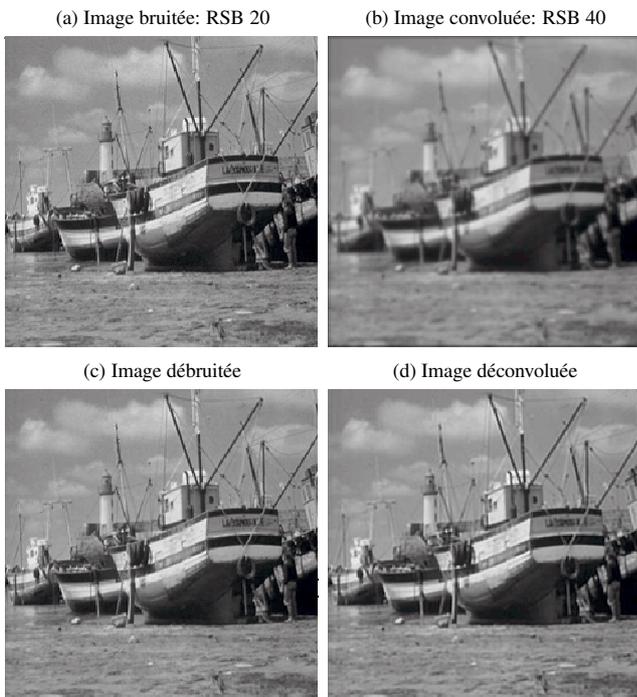


FIG. 1: (a)-(b) Images simulées. (c)-(d) Restauration par minimisation de critère pénalisé.

5.2 Les versions algorithmiques testées

À présent, nous comparons les vitesses de convergence des familles d'algorithmes précédemment décrites, où la notation {P} signifie avec ou sans préconditionnement CBC:

- 1 Les algorithmes GR et GY approchés avec inversion des matrices normales par une forme tronquée des GC ou GCP linéaires: GR+GC{P} et GY+GC{P}.
- 2 L'algorithme PRGC ou PRGCP non linéaire avec détermination du pas
 - 2.1 à l'aide de la recherche scalaire `fminbnd` de Matlab: GC{P}+fminbnd
 - 2.2 à l'aide des formes scalaires de GR ou GY (6): GC{P}+GR1D et GC{P}+GY1D.

Toutes ces familles d'algorithmes présentent la même structure itérative: des itérations globales au sein desquelles des sous-itérations sont effectuées, que ce soit des GC linéaires pour la

première famille d'algorithmes ou bien une recherche du pas pour la deuxième famille d'algorithmes. Le critère d'arrêt (7) porte sur les itérations globales, mais il nous faut aussi définir un critère d'arrêt pour les sous-itérations.

Pour la première famille d'algorithmes, nous reprenons la proposition de [4]: les GC{P} linéaires sont utilisés avec le critère d'arrêt $\|r^i\|/\|r^0\| < 10^{-6}$, où r^i est le résidu de l'équation normale après i itérations.

La fonction `fminbnd` (nombre d'or et interpolation parabolique) de Matlab est utilisée avec son critère d'arrêt par défaut.

Les formes scalaires de GR ou GY (6) sont utilisées avec I itérations, où I est fixé au préalable.

5.3 Résultats

Nous comparons expérimentalement l'efficacité des familles d'algorithmes 1, 2.1 et 2.2 pour la minimisation des critères définis précédemment, ceci selon deux types d'évaluations complémentaires:

- le nombre d'itérations globales et le nombre moyen de sous-itérations qui donnent une idée de la complexité de l'algorithme considéré,
- le temps mis pour effectuer la minimisation qui dépend bien sûr de la plateforme d'implémentation.

Les Tab. 1 et 2 récapitulent, pour un algorithme donné, le premier critère d'évaluation sous la forme (It1, It2), où:

- It1 est le nombre d'itérations globales,
- It2 est le nombre moyen de sous-itérations.

Les expériences ont été menées avec Matlab 6.5 sur un PC P4 2.8GHz RAM 1Go. Les algorithmes ont été rangés par ordre de durée de minimisation croissant.

Nous avons testé la famille d'algorithme 2.2 avec $I \in \{1, 2, 3, 4, 5, 10, 15, 20\}$. Seules les versions $I = 1$ ont été systématiquement reportées dans Tab. 1 et 2, les autres versions n'ayant été reportées que si elles sont plus rapides.

	Sans precond.		Avec precond.	
	Iter.	Tps (s)	Iter.	Tps (s)
GC+GR1D	(11, 2)	9.2	(9, 2)	10.62
"	(12, 1)	7.95	(13, 1)	12.57
GC+GY1D	(11, 4)	12.19	(9, 4)	13.17
"	(14, 1)	9.67	(20, 1)	19.5
GC+fminbnd	(12, 11.8)	20.85	(10, 12.9)	21.35
GR+GC	(10, 21.5)	62.76	(10, 12.3)	82.23
GY+GC	(20, 20.5)	91.83	(20, 7.3)	86.14

TAB. 1: Comparaison des algorithmes sur le problème de débruitage.

	Sans precond.		Avec precond.	
	Iter.	Tps (s)	Iter.	Tps (s)
GC+GR1D	(90, 1)	140.1	(43, 1)	79.16
GC+GY1D	(89, 2)	151.3	(45, 2)	89.91
"	(93, 1)	145.7	(53, 1)	98.85
GC+fminbnd	(94, 9.3)	220.1	(47, 10.4)	129.4
GR+GC	(13, 1135.5)	20485	(13, 161.8)	3598
GY+GC	(38, 498.7)	26490	(38, 66.9)	4331

TAB. 2: Comparaison des algorithmes sur le problème de déconvolution.

L'analyse des Tab. 1 et 2 fait ressortir en premier lieu que la famille d'algorithmes 2 surpasse largement la famille d'algorithmes 1 tant du point de vue du nombre d'itérations que

du temps de calcul. Si le préconditionnement CBC est très favorable pour le problème de déconvolution, il devient presque systématiquement contre productif pour le problème de débruitage. Ceci est une conséquence du meilleur conditionnement de ce dernier. Cependant, l'utilisation d'un préconditionneur à base de transformée en cosinus pourrait être plus efficace que celui utilisé à base de transformée de Fourier.

Au sein de chaque famille d'algorithmes, les versions GR font systématiquement mieux que les versions GY. Ainsi, par souci de clarté nous avons exclu les versions GY de la Fig. 2.

Enfin, il apparaît clairement qu'un nombre très faible de réitérations des formes scalaires SQ fournit à chaque fois la version la plus rapide parmi la famille d'algorithmes 2.2, le cas non réitéré ($I = 1$) arrivant même souvent en tête.

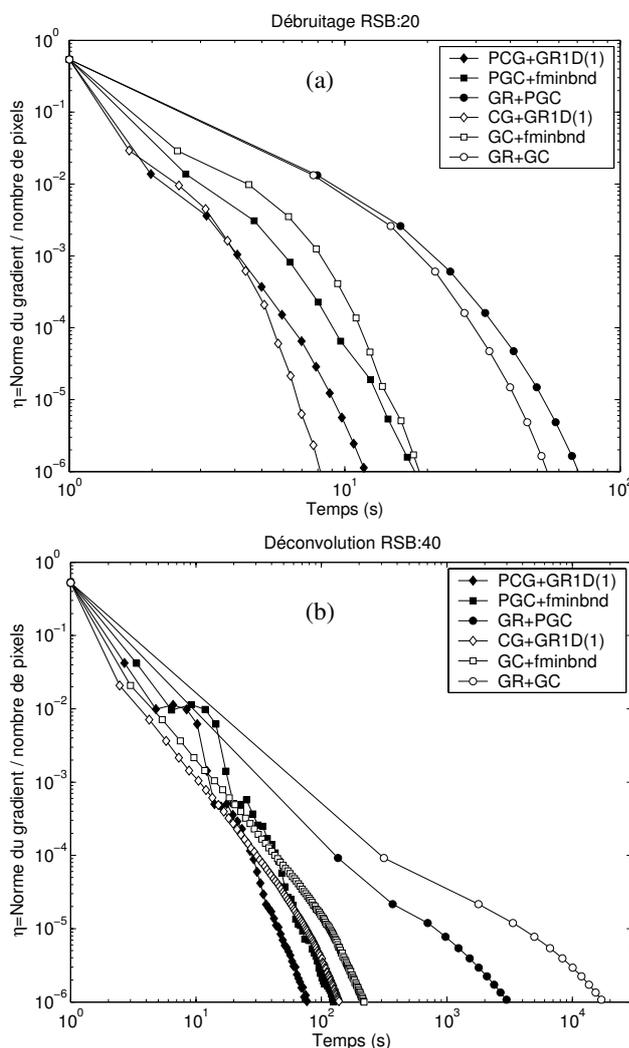


FIG. 2: Comparaison des taux de convergence: $\|\nabla \mathcal{J}(x^k)\|/N$ en fonction du temps. (a) Problème de débruitage. (b) Problème de déconvolution.

6 Conclusion

La famille d'algorithmes PRGC{P} non linéaires avec un pas déterminé par une forme scalaire SQ présente plusieurs intérêts majeurs par rapport aux algorithmes SQ avec inversion

tronquée par GC{P} linéaires:

- la convergence de la première famille est assurée par le Théorème 1, alors que la convergence de la seconde famille est à l'heure actuelle un problème ouvert, et
- la première famille apparaît expérimentalement comme largement plus efficace à tel point qu'il nous semble raisonnable de penser que pour les problèmes étudiés, les ordres de grandeurs des écarts temporels observés seront préservés indépendamment de l'implémentation.

Pour finir, le Théorème 1 assurant la convergence de la famille d'algorithme PRGC{P} non linéaires pour un pas déterminé par une forme scalaire SQ présente aussi des avantages par rapport aux résultats de convergence existants portant sur une recherche du pas qui vérifie des conditions telles que celles de Wolfe:

- d'efficacité, car il permet d'éviter de faire des sous-itérations inutiles puisque la convergence est assurée pour un nombre de sous-itérations I quelconque, donc aussi petit que souhaité; et
- de simplicité, car il dépend du seul paramètre naturel I alors que le réglage des paramètres des conditions de Wolfe est délicat.

- [1] J. Idier, «Convex half-quadratic criteria and interacting auxiliary variables for image restoration», *IEEE Trans. Image Processing*, vol. 10, n° 7, pp. 1001–1009, July 2001.
- [2] P. Charbonnier, L. Blanc-Féraud, G. Aubert et M. Barlaud, «Deterministic edge-preserving regularization in computed imaging», *IEEE Trans. Image Processing*, vol. 6, n° 2, pp. 298–311, Feb. 1997.
- [3] M. Allain, J. Idier et Y. Goussard, «On global and local convergence of half-quadratic algorithms», to appear, *IEEE Trans. Image Processing*, 2005.
- [4] M. Nikolova et M. Ng, «Fast image reconstruction algorithms combining half-quadratic regularization and preconditioning», in *Proc. IEEE ICIP*, Thessaloniki, Greece, Oct. 2001, pp. 277–280.
- [5] M. Nikolova et M. Ng, «Analysis of half-quadratic minimization methods for signal and image recovery», to appear, *SIAM Journal on Scientific computing*, 2005.
- [6] J. A. Fessler et S. D. Booth, «Conjugate-gradient preconditioning methods for shift-variant PET image reconstruction», *IEEE Trans. Image Processing*, vol. 8, n° 5, pp. 668–699, May 1999.
- [7] W. H. Press, S. A. Teukolsky, W. T. Vetterling et B. P. Flannery, *Numerical recipes in C, the art of scientific computing*, Cambridge Univ. Press, New York, 2nd edition, 1992.
- [8] J. Sun et J. Zhang, «Global convergence of conjugate gradient methods without line search», *Ann. Operations Research*, vol. 103, pp. 161–173, 2001.
- [9] C. Labat et J. Idier, «Extended convergence properties of preconditioned conjugate gradient methods without line search and application to image restoration», Rapport technique, IRCCyN, 2005.