

Étude comparative d’algorithmes rapides de reconstruction d’image adaptés pour le *compressed sensing*

Yoann LE MONTAGNER^{1,2}, Elsa ANGELINI², Jean-Christophe OLIVO-MARIN¹

¹Institut Pasteur, Unité d’Analyse d’Images Quantitative, France

²Institut Télécom, Télécom ParisTech, CNRS LTCI, France

lemontag@telecom-paristech.fr, elsa.angelini@telecom-paristech.fr, jcolivo@pasteur.fr

Résumé – Ce papier présente et compare trois algorithmes d’optimisation adaptés à la reconstruction d’images échantillonnées dans l’espace de Fourier, dans un contexte de *compressed sensing*. Nous privilégions ici une approche empirique, cherchant à comparer les performances de ces algorithmes sur des données réelles (images biologiques). Nous mettons en évidence des disparités importantes en termes de temps de calcul, et discutons de quelques différences fines que l’on peut observer parmi les images reconstruites.

Abstract – This paper presents and compares three optimization algorithms well-suited to reconstruct Fourier sampled images within a *compressed sensing* framework. Our approach is mainly empirical, as we compare the algorithms on real data (biological images). We highlight large discrepancies in terms of computation time, and discuss some subtle differences that can be observed on the reconstructed images.

1 Présentation des algorithmes

1.1 Généralités

L’imagerie par *compressed sensing* (CS) requiert que l’on soit capable de résoudre efficacement un problème de minimisation convexe de la forme suivante :

$$\min_{x \in \mathbb{R}^N} \|x\|_{\text{TV}} \quad \text{t.q.} \quad \|\Phi x - y\|_{l_2} \leq \epsilon \quad (1)$$

$$\min_{x \in \mathbb{R}^N} \|\Psi x\|_{l_1} \quad \text{t.q.} \quad \|\Phi x - y\|_{l_2} \leq \epsilon \quad (2)$$

où x représente l’image à reconstruire, Φ est la matrice d’échantillonnage, et Ψ la matrice de passage vers la base dans laquelle le signal x est supposé parcimonieux (cf. [4, 5]). Dans le contexte de notre étude, Φ est une transformée de Fourier dont on masque certains coefficients¹, et Ψ une transformée en ondelettes. Ces problèmes sont difficiles mathématiquement pour au moins deux raisons :

1. la fonction objectif ($\|\cdot\|_{l_1}$ ou $\|\cdot\|_{\text{TV}}$) n’est pas différentiable ;
2. la dimension de l’espace des solutions \mathbb{R}^N est très grande dans le cas des problèmes d’imagerie ($N \geq 10^6$ typiquement).

Certaines approches précédemment employées pour résoudre ces problèmes consistent à les reformuler sous la forme de problèmes linéaires ou coniques de second ordre (cf. [5, 3]), puis à les résoudre en utilisant des méthodes

1. i.e. Φ est obtenue en supprimant certaines lignes de la matrice de transformée de Fourier.

d’optimisation itératives telles que des méthodes de points intérieurs (cf. [2]). Cependant, l’obtention d’une solution par ces méthodes nécessite généralement d’inverser un système linéaire de taille N , ce qui est très coûteux en raison de la dimension élevée du problème. En fait, ce type d’approche non-spécialisée n’exploite pas les propriétés algébriques particulières de Φ et Ψ , ou le fait qu’il puisse exister un algorithme rapide pour appliquer les transformées sous-jacentes. Aussi, de nouveaux algorithmes mieux adaptés aux problèmes de reconstruction CS sont apparus ces dernières années, accroissant significativement l’intérêt pratique du *compressed sensing* en imagerie. Nous présentons une revue de trois de ces algorithmes², à savoir :

- NESTA (présenté par [1]), qui traite du cas où la matrice Φ vérifie la relation $\Phi\Phi^* = \text{Id}$ ³ ;
- RecPF (présenté par [10]), qui permet de résoudre les problèmes de reconstruction CS dans le cas particulier où la matrice d’échantillonnage Φ est constituée d’un sous-ensemble de vecteurs de Fourier ;
- SPGL1 (présenté par [8]), qui ne requiert pas de propriétés particulières sur Φ , mais qui n’est pas applicable pour la résolution des problèmes CS utilisant la norme TV comme fonction objectif.

2. Des implémentations Matlab® de ces algorithmes sont fournies par leurs auteurs respectifs aux adresses suivantes :

- NESTA : <http://www.acm.caltech.edu/~nESTA/>
- RecPF : <http://www.caam.rice.edu/~optimization/L1/RecPF/>
- SPGL1 : <http://www.cs.ubc.ca/labs/scl/spgl1/>

3. i.e. les lignes de la matrice Φ forment une famille orthonormale

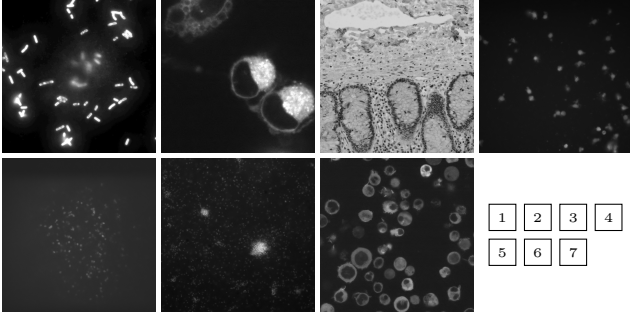


FIGURE 1 – Images (carrées) utilisées pour les tests, de largeurs respectives 425, 453, 716, 960, 992, 1024 et 1024 pixels. Nous présentons ici les résultats détaillés obtenus pour les images 1 (cf. fig. 2) et 7 (cf. fig. 3).

1.2 NESTA

NESTA est un algorithme spécialisé pour la résolution des problèmes de CS proposé par Becker *et al.* (cf. [1]), basé sur les travaux de Nesterov sur la minimisation de fonctions non-lisses (cf. [7]). Cet algorithme, qui peut être utilisé pour résoudre (1) ou (2), est un algorithme itératif de descente de gradient : à chaque étape de l’algorithme, le nouvel itéré est obtenu à partir d’une combinaison linéaire de deux termes :

1. un terme tendant à faire évoluer l’itéré dans la direction opposée au gradient de la fonction objectif,
2. un terme qui mémorise les directions des gradients calculés au cours des itérations précédentes. Dans [7], l’auteur montre que ce terme additionnel améliore les propriétés de convergence de l’algorithme.

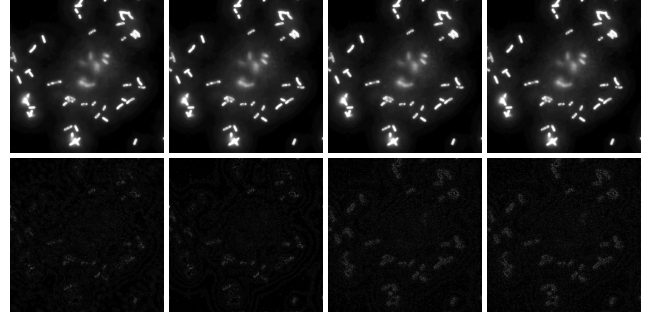
Ces termes sont tous deux définis à chaque étape comme étant la solution de problèmes auxiliaires d’optimisation quadratique, dont la solution peut être exprimée sous forme analytique. En outre, si la transformée d’échantillonnage Φ vérifie la relation $\Phi\Phi^* = \text{Id}$, tous les inverses matriciels apparaissant dans la solution analytique des problèmes auxiliaires peuvent être simplifiés, ouvrant ainsi la voie à un calcul très rapide de la solution : alors, la complexité algorithmique de chaque itération de NESTA est en $\mathcal{O}(N + \mathcal{C}_\Phi)$, où \mathcal{C}_Φ représente le nombre d’opérations à réaliser pour appliquer Φ ou Φ^* à un vecteur⁴.

1.3 RecPF

RecPF est un algorithme présenté par Yang *et al.* dans [10], spécialisé pour le cas précis où Φ est une transformée de Fourier sous-échantillonnée. Plus précisément, RecPF résout le problème suivant :

$$\min_{x \in \mathbb{R}^N} a_{\text{TV}} \|x\|_{\text{TV}} + a_{l_1} \|\Psi x\|_{l_1} + \frac{1}{2} \|\Phi x - y\|_{l_2}^2 \quad (3)$$

4. $\mathcal{C}_\Phi = \mathcal{O}(N \log(N))$ dans le cadre de notre étude, où Φ est une transformée de Fourier sous-échantillonnée



	Minimisation TV		Minimisation l_1	
	NESTA	RecPF	NESTA	SPGL1
Fct. obj.	2230	2254	3977	3985
RMS	7.5×10^{-3}	7.0×10^{-3}	9.3×10^{-3}	9.8×10^{-3}
SSIM	0.97	0.98	0.96	0.96
#iter.	136	26	73	283
Temps	14 sec.	1 sec.	10 sec.	47 sec.

FIGURE 2 – Résultats obtenus pour l’image *Shigella*. Les images reconstruites avec les quatre méthodes sont présentées sur la première ligne, les différences entre ces reconstructions et l’image originale débruitée sur la seconde ligne. L’ordre des images correspond à l’ordre des colonnes du tableau.

où a_{TV} et a_{l_1} sont deux constantes positives⁵. Les problèmes (1) et (3) sont équivalents si l’on impose $a_{l_1} = 0$.

L’idée clé de RecPF consiste à réécrire (3) en introduisant un jeu de variables auxiliaires de sorte que l’inconnue x n’apparaisse plus que dans des termes en $\|\cdot\|_{l_2}^2$. Ainsi, pour $a_{l_1} = 0$, (3) est réécrit sous la forme suivante :

$$\min_{x, w_h, w_v \in \mathbb{R}^N} \left\{ a_{\text{TV}} \|w_n\|_{l_1} + \frac{1}{2} \|\Phi x - y\|_{l_2}^2 + \frac{\beta}{2} \left(\|w_h - D_h x\|_{l_2}^2 + \|w_v - D_v x\|_{l_2}^2 \right) \right\} \quad (4)$$

où D_h et D_v sont les opérateurs de convolution calculant les dérivées horizontales et verticales de l’image x , et $w_n = \sqrt{w_h^2 + w_v^2}$ point-à-point. On remarque que (3) et (4) deviennent équivalents quand $\beta \rightarrow \infty$.

Après réécriture, l’algorithme va minimiser l’expression (4) alternativement en w à x fixé, puis en x à w fixé, puis à nouveau en w , etc :

- la minimisation en w est réalisée en $\mathcal{O}(N)$ opérations dans la mesure où (4) est séparable en w ;
- la minimisation en x est un problème quadratique dont la solution peut être calculée en $\mathcal{O}(N \log(N))$ opérations, notamment grâce au fait que la transformée de Fourier diagonalise les convolutions D_h et D_v .

Au final, chaque itération de RecPF a une complexité algorithmique en $\mathcal{O}(N \log(N))$.

5. À noter que [10] présente RecPF dans un cadre imposant $a_{\text{TV}} > 0$ et $a_{l_1} > 0$, mais que l’implémentation Matlab® fournie par les auteurs de l’algorithme autorise et prévoit explicitement l’annulation du paramètre a_{l_1} .

1.4 SPGL1

SPGL1 est un algorithme introduit par Van den Berg *et al.* dans [8] pour résoudre le problème suivant :

$$\min_{x \in \mathbb{R}^N} \|x\|_{l_1} \quad \text{t.q.} \quad \|\Phi x - y\|_{l_2} \leq \epsilon \quad (5)$$

Cet algorithme ne requiert pas de propriété algébrique particulière sur Φ , pourvu que son produit avec un vecteur puisse être calculé efficacement. L'adaptation de SPGL1 pour la résolution de (2) est directe dans le cas où l'opérateur Ψ est inversible et bien conditionné; en revanche, SPGL1 ne peut pas résoudre (1) (reconstruction TV).

L'idée maîtresse présentée dans [8] consiste à étudier le problème alternatif suivant :

$$\min_{x \in \mathbb{R}^N} \|\Phi x - y\|_{l_2} \quad \text{t.q.} \quad \|x\|_{l_1} \leq \tau \quad (6)$$

En résolvant plusieurs instances de ce problème pour différentes valeurs du paramètre τ , on arrive à approcher une solution de (5). Chaque instance de (6) est résolue itérativement par descente de gradient avec projection sur l'ensemble des solutions admissibles. SPGL1 est présenté comme très efficace pour les raisons suivantes :

- il utilise une méthode très rapide pour calculer la projection sur l'ensemble convexe $\{x \text{ t.q. } \|x\|_{l_1} \leq \tau\}$;
- grâce à une étude des propriétés de la courbe $\epsilon = \phi(\tau)$ où $\phi(\tau)$ est la valeur de la solution de (6), les auteurs adoptent une stratégie efficace d'ajustement du paramètre τ permettant de limiter le nombre d'instances de (6) à résoudre pour approcher la solution de (5).

2 Simulations comparatives

2.1 Méthodologie

Nous avons testé ces algorithmes de reconstruction sur un jeu de sept images biologiques présentant des caractéristiques diverses en termes de contenu, de niveau de bruit et de taille. Les images ont été reconstruites en utilisant NESTA et RecPF avec une contrainte de régularisation TV, puis en utilisant NESTA et SPGL1 avec une contrainte de régularisation l_1 sur les coefficients d'ondelettes⁶. Toutes les simulations utilisent le même motif d'échantillonnage aléatoire gaussien⁷, avec un taux de sous-échantillonnage de 15%.

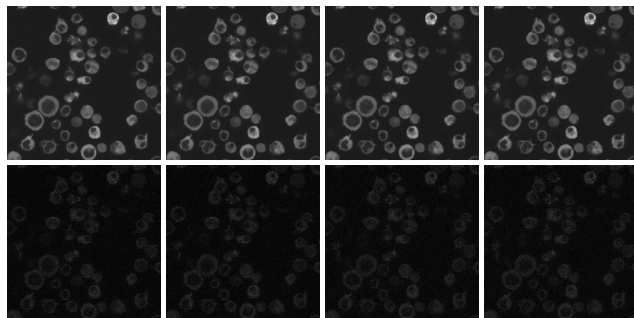
Pour chaque image et chaque méthode de reconstruction, nous avons évalué :

- la valeur finale de la fonction objectif⁸,
- la fidélité de la reconstruction par rapport à l'image source (évaluée à partir de l'écart quadratique moyen

6. ondelette de Daubechies d'ordre 4

7. i.e. la probabilité d'échantillonner le coefficient correspondant à la fréquence spatiale $k \in \mathbb{R}^2$ de l'espace de Fourier est définie par $\exp\left(-\left(\|k\|_{l_2} / \rho\right)^2\right)$, où le paramètre ρ est ajusté en fonction du taux d'échantillonnage visé.

8. $\|\Psi x\|_{l_1}$ pour SPGL1 et NESTA, $\|x\|_{\text{TV}}$ pour RecPF et NESTA



	Minimisation TV		Minimisation l_1	
	NESTA	RecPF	NESTA	SPGL1
Fct. obj.	4962	5177	15.6×10^3	15.7×10^3
RMS	1.2×10^{-2}	1.2×10^{-2}	1.2×10^{-2}	1.2×10^{-2}
SSIM	0.92	0.92	0.92	0.92
#iter.	179	20	68	1693
Temps	82 sec.	2 sec.	47 sec.	3176 sec.

FIGURE 3 – Résultats obtenus pour l'image *Lymphocytes T* (même présentation que pour la figure 2).

(RMS) et d'un indice de similarité (SSIM⁹) entre la reconstruction et une vérité terrain construite par débruitage de l'image originale),

- la vitesse de convergence de l'algorithme (en nombre d'itérations et en temps de calcul¹⁰).

Les résultats obtenus avec ces quatre méthodes d'optimisation sont présentés sur les figures 2 et 3.

2.2 Analyse des résultats obtenus

2.2.1 Minimisation de la fonction objectif

Les algorithmes testés produisent des résultats très similaires en termes de valeur finale de la fonction objectif minimisée : les écarts relatifs du minimum $\|x^*\|_{\text{TV}}$ entre les solutions x^* exhibées par NESTA et par RecPF sont généralement compris entre 1% et 5%¹¹; les écarts relatifs de $\|\Psi x^*\|_{l_1}$ entre les solutions x^* exhibées par NESTA et par SPGL1 sont quand à eux systématiquement plus petits que 1%. Ce critère n'est donc pas discriminant dans une perspective de comparaison des algorithmes.

9. L'indice SSIM est valué entre -1 et 1, une valeur proche de 1 indiquant une grande similarité entre les deux images comparées (cf. [9] pour davantage de détails quant à la construction de cet indice).

10. Cette dernière mesure dépendant bien évidemment de l'implémentation des algorithmes et de la machine utilisée pour réaliser les simulations. Ces données ont été obtenues à l'aide des implémentations Matlab® des algorithmes fournies par leurs auteurs, sur une station de travail Mac OS équipée de 24Go de RAM et de deux CPUs Quad-Core Intel Xeon fonctionnant à 2.4 GHz.

11. À noter toutefois, dans le cas particulier de RecPF, que la fonction objectif incorpore le terme d'attache aux données $\|\Phi x - y\|_{l_2}$; pour nos simulations, afin de pouvoir comparer les résultats obtenus avec RecPF et avec NESTA, nous avons veillé à ajuster le multiplicateur de Lagrange entre le terme $\|x\|_{\text{TV}}$ et le terme d'attache aux données de sorte que les valeurs finales de ce dernier obtenues avec les deux algorithmes soient similaires. La détermination d'une valeur générique adéquate pour ce paramètre ne présente pas de difficultés particulières.

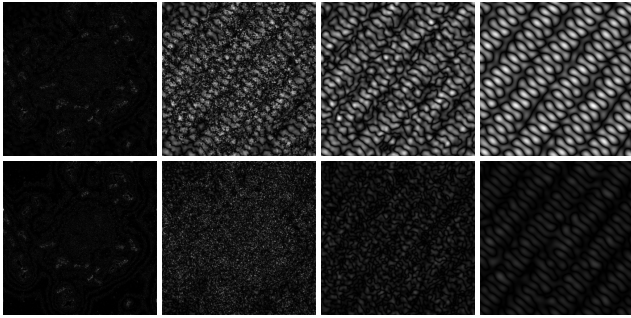


FIGURE 4 – Images de résidus obtenues pour *Shigella* par régularisation TV avec NESTA (rangée supérieure) et RecPF (rangée inférieure), et trois versions de ces images ayant subi des filtrages basse-fréquence avec des fréquences de coupure décroissantes. La même palette de niveaux de gris est utilisée pour les deux images de chaque colonne.

2.2.2 Fidélité de la reconstruction

En termes de qualité de reconstruction, tous les algorithmes produisent également des résultats globalement similaires, que la comparaison soit réalisée sur la base des mesures de fidélité RMS ou SSIM, ou que l'on procède par évaluation visuelle des résultats. L'effet de débruitage induit par le CS peut être observé sur les images de résidus, lesquelles présentent des valeurs plus importantes dans les zones d'intensité élevée (distributions de type Poisson).

Toutefois, une étude plus poussée des résidus montre que l'algorithme NESTA utilisé avec une régularisation TV produit des artefacts texturés de basse fréquence indésirables (cf. figure 4). Ces artefacts correspondent aux quelques coefficients basses fréquences non-échantillonnés par le CS, qui ne sont pas reconstruits correctement. Les raisons expliquant la sensibilité importante de cet algorithme à ces coefficients manquants restent à expliquer.

2.2.3 Vitesse de reconstruction

La différence principale entre ces méthodes de reconstruction réside au niveau du temps d'exécution nécessaire pour le calcul des solutions des problèmes (1) et (2). Si les ratios exacts entre les temps de calcul dépendent fortement des instances effectives des problèmes, on peut retenir des données présentées sur les figures 2 et 3 que :

- RecPF est plus rapide que NESTA d'un facteur environ 10 pour des images 400×400 pixels, et que cet écart s'accroît au fur et à mesure que la taille des images augmente ;
- NESTA est plus rapide que SPGL1 d'un facteur au moins 3 ou 4, mais que ce facteur peut s'avérer beaucoup plus important lors de la résolution de certaines instances du problème (2).

Cette dernière observation est cohérente avec les résultats présentés dans [1], où les auteurs mettent en évidence

le fait que l'algorithme SPGL1 peut dans certains cas s'exécuter très rapidement (plus rapidement que NESTA), mais que ce temps d'exécution varie de façon importante en fonction du signal d'entrée ; à l'inverse, le temps d'exécution de NESTA est assez stable pour une taille de signal d'entrée donnée.

En outre, le fait que RecPF soit plus rapide que NESTA n'est pas suffisant pour disqualifier ce dernier algorithme, et ce au moins pour deux raisons :

- NESTA est plus polyvalent que RecPF, dans le sens où RecPF est limité à un échantillonnage dans l'espace de Fourier ;
- l'implémentation Matlab® de RecPF utilise des fonctions MEX écrites en C (alors que l'implémentation de NESTA n'en utilise pas), ce qui peut introduire un biais dans la mesure des temps de calcul.

Les comptages d'itérations semblent indiquer toutefois une très rapide convergence de RecPF par rapport à NESTA, même si ces données sont à considérer avec prudence¹².

3 Conclusion

Dans ce papier, nous comparons trois algorithmes résolvant les problèmes de reconstruction d'images rencontrés en *compressed sensing* (NESTA, RecPF, et SPGL1). Pour des résultats similaires en termes de qualité de reconstruction, les simulations montrent une grande disparité au niveau des temps de calcul, en faveur de RecPF.

Références

- [1] S. Becker, J. Bobin, and E. Candès. NESTA : A fast and accurate first-order method for sparse recovery. *SIAM J. on Imaging Sciences*, 4 :1–39, 2011.
- [2] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- [3] E. Candès and J. Romberg. L1-MAGIC : Recovery of sparse signals via convex programming. Technical report, Caltech, 2005.
- [4] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles : exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. on Information Theory*, 52 :489–509, 2006.
- [5] E. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59 :1207–1223, 2006.
- [6] Y. Le Montagner, E. Angelini, and J.-C. Olivo-Marin. Comparison of reconstruction algorithms in compressed sensing applied to biological imaging. In *ISBI*, 2011.
- [7] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Université Catholique de Louvain, 2007.
- [8] E. van den Berg and M. P. Friedlander. Probing the Pareto frontier for basis pursuit solutions. *SIAM J. on Scientific Computing*, 31 :890–912, 2008.
- [9] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment : from error visibility to structural similarity. *IEEE Trans. on Image Processing*, 13 :600–612, Avril 2004.
- [10] J. Yang, Y. Zhang, and W. Yin. A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data. *IEEE J. of Selected Topics in Signal Processing*, 4 :288–297, Avril 2010.

¹² Même si la complexité algorithmique d'une itération RecPF est la même que celle d'une itération NESTA, rien ne garantit que leurs coûts respectifs en temps d'exécution soient les mêmes.