

# Frank-Wolfe pour la reconstruction parcimonieuse

Farah CHERFAOUI<sup>1</sup>, Valentin EMIYA<sup>1</sup>, Liva RALAIVOLA<sup>1,3</sup>, Sandrine ANTHOINE<sup>2</sup>

<sup>1</sup>Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France.

<sup>2</sup>Aix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, France.

<sup>3</sup>IUF and Criteo

pre nom@univ-amu.fr

**Résumé** – Nous nous intéressons ici aux propriétés de l’algorithme de Frank-Wolfe pour le problème de reconstruction de signaux parcimonieux, appelé le problème m-EXACT-SPARSE. Nous démontrons que lorsque le dictionnaire est quasi-incohérent, à chaque itération, l’algorithme de Frank-Wolfe sélectionne uniquement un atome du support. Nous démontrons aussi que sous cette même condition, il existe une itération à partir de laquelle l’algorithme converge exponentiellement.

**Abstract** – We study the properties of the Frank-Wolfe algorithm to solve the m-EXACT-SPARSE problem. We prove that when the dictionary is *quasi-incoherent*, then the iterative process implemented by the Frank-Wolfe algorithm only recruits atoms from the support of the signal. We also prove that when the dictionary is quasi-incoherent, there exists an iteration beyond which the algorithm converges exponentially.

## 1 Introduction

### 1.1 Le problème m-EXACT-SPARSE

Un signal  $y \in \mathbb{R}^d$  est dit  $m$ -parcimonieux dans un dictionnaire  $\Phi \in \mathbb{R}^{d \times n}$ , s’il s’écrit comme une combinaison linéaire d’au plus  $m$  atomes (colonnes) de  $\Phi$ . Dans tout ce qui suit, nous noterons par  $\Phi = [\varphi_1 \cdots \varphi_n] \in \mathbb{R}^{d \times n}$  la matrice dont les colonnes sont les  $n$  atomes  $\varphi_1, \dots, \varphi_n \in \mathbb{R}^d$ , tel que  $\|\varphi_i\|_2 = 1, \forall i$ . Le support  $\Lambda$  d’un signal  $y$   $m$ -parcimonieux, est le plus petit ensemble  $\{1, \dots, n\}$  de taille  $m$  tel que  $y$  est une combinaison linéaire des atomes indexés par  $\Lambda$  (nous donnerons en section 3 les conditions nécessaires pour que le support soit unique). Pour un signal  $y$   $m$ -parcimonieux et un dictionnaire  $\Phi$ , le problème m-EXACT-SPARSE est le suivant :

$$\text{trouver } x \text{ s.c. } y = \Phi x \text{ et } \|x\|_0 \leq m,$$

où  $\|\cdot\|_0$  compte le nombre d’entrées non nulles de son argument. Comme  $y$  est une combinaison d’au plus  $m$  atomes de  $\Phi$ , il existe une solution au problème :

$$\arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|y - \Phi x\|_2^2 \text{ s.c. } \|x\|_0 \leq m, \quad (1)$$

qui est une solution de m-EXACT-SPARSE.

### 1.2 État de l’art

Le problème m-EXACT-SPARSE est NP-dur [5]. La question que nous pouvons donc nous poser est : sous quelles conditions ce problème devient faisable ? Au cours des trente dernières années, plusieurs algorithmes ont été proposés pour répondre à cette question, parmi eux les algorithmes gloutons Matching

Pursuit (MP) et Orthogonal Matching Pursuit (OMP) [2]. Une autre manière d’attaquer ce problème est de le relaxer, par exemple en remplaçant la quasi-norme  $l_0$  par la norme  $l_1$  (LASSO [6], Basis Pursuit [7]). Ceci donne lieu à un problème convexe, pour lequel un certain nombre de méthodes peuvent être utilisées. Ici, nous considérons la relaxation suivante :

$$\arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|y - \Phi x\|_2^2 \text{ s.c. } \|x\|_1 \leq \beta \quad (2)$$

et utilisons l’Algorithme de Frank-Wolfe pour la résoudre.

Afin de motiver un peu plus nos travaux, revenons au problème m-EXACT-SPARSE. Les algorithmes MP et OMP [2] ont été largement utilisés et étudiés pour la résolution de ce problème. Tropp [2], Gribonval et Vandergheynst [3] ont démontré que lorsque le dictionnaire est quasi-incohérent (par exemple proche d’une base orthonormée), alors à chaque itération MP et OMP sélectionnent un atome du support. Ils ont aussi démontré que ces deux algorithmes convergent exponentiellement vite.

D’un autre côté, l’algorithme de Frank-Wolfe [1] est un algorithme d’optimisation convexe, qui converge exponentiellement vite lorsque la fonction objectif est fortement convexe [4], et linéairement dans les autres cas. Dans ce travail, nous étudions les propriétés de reconstruction et de convergence de l’algorithme de Frank-Wolfe pour la résolution du Problème (2). Notons que comme les colonnes de  $\Phi$  peuvent être linéairement dépendantes, la fonction objectif considérée dans (2) n’est pas fortement convexe.

### 1.3 Principaux résultats

Nos résultats dépendent d'une quantité fondamentale associée à un dictionnaire  $\Phi = [\varphi_1 \cdots \varphi_n]$ , sa fonction de Babel :

$$\mu_1(m) = \max_{|\Lambda|=m} \max_{i \notin \Lambda} \sum_{j \in \Lambda} |\langle \varphi_i, \varphi_j \rangle|.$$

Pour  $m$  donné,  $\mu_1(m)$  mesure approximativement à quel point un atome de  $\Phi$  peut être exprimé comme une combinaison de  $m$  autres atomes. Lorsque  $m = 1$ , la fonction de Babel devient la cohérence de  $\Phi$  :

$$\mu = \mu_1(1) = \max_{j \neq k} |\langle \varphi_j, \varphi_k \rangle|.$$

Enfin, le dictionnaire est dit *quasi-incohérent* lorsque

$$m < \frac{1}{2}(\mu^{-1} + 1).$$

Dans ce travail, nous démontrons que lorsque le dictionnaire est quasi-incohérent, alors l'algorithme de Frank-Wolfe reconstruit exactement tout signal  $m$ -parcimonieux. En d'autres termes, à chaque étape, l'algorithme sélectionne un atome du support du signal. Nous démontrons aussi, que sous cette même condition, il existe une itération à partir de laquelle l'algorithme de Frank-Wolfe converge exponentiellement même si la fonction considérée n'est pas fortement convexe.

Le reste du papier est organisé comme suit. Nous présentons dans la section 2, l'instanciation de l'algorithme de Frank-Wolfe au Problème (2), ainsi que ses points communs avec les algorithmes MP et OMP. Nous énonçons nos résultats théoriques en section 3 (les preuves sont disponibles dans [8]). Enfin, nous vérifierons l'optimalité de nos résultats à travers des expériences numériques présentées en section 4.

## 2 MP, OMP et Frank-Wolfe

Dans cette section, nous rappelons rapidement les algorithmes MP, OMP, et Frank-Wolfe. Nous présentons ensuite l'instanciation de l'algorithme de Frank-Wolfe au Problème (2). Enfin, nous pointerons du doigt les similarités entre cette instanciation et les algorithmes MP et OMP.

### 2.1 Matching Pursuit et Orthogonal Matching Pursuit

Soit  $\Phi$  un dictionnaire et  $y$  un signal  $m$ -parcimonieux. Les algorithmes MP et OMP [2] sont des algorithmes gloutons utilisés pour approximer un signal. En notant  $y_k$  l'approximation courante et  $r_k$  le résidu associé, MP et OMP sélectionnent itérativement l'atome qui maximise son produit scalaire avec le résidu. Les deux algorithmes peuvent être résumés comme suit : après avoir initialisé  $y_0 = 0$  et  $r_0 = y$ , les trois étapes suivantes sont répétées :

1. Sélection d'atome :  $\lambda_k = \arg \max_i |\langle \varphi_i, r_k \rangle|$ .
2. Mise à jour de l'approximation :
  - (a) MP :  $y_{k+1} = y_k + \langle \varphi_{\lambda_k}, r_k \rangle \varphi_{\lambda_k}$
  - (b) OMP :  $y_{k+1} = \arg \min_{a \in \text{span}(\{\varphi_{\lambda_0}, \dots, \varphi_{\lambda_k}\})} \|y - a\|_2$
3. Mise à jour du résidu :  $r_{k+1} = y - y_{k+1}$ .

### 2.2 L'algorithme de Frank-Wolfe

L'algorithme de Frank-Wolfe [1] est un algorithme itératif développé pour résoudre des problèmes du type :

$$\min_{x \in \mathcal{C}} f(x) \text{ s.c. } x \in \mathcal{C} \quad (3)$$

où  $f$  est une fonction convexe et continûment différentiable, et  $\mathcal{C}$  est un ensemble compact convexe. La première étape de l'algorithme consiste à choisir un élément  $x_0$  quelconque de  $\mathcal{C}$ . Ensuite, pour toute itération  $k$ , l'algorithme applique les étapes suivantes :

1. Direction de descente :  $s_k = \arg \min_{s \in \mathcal{C}} \langle s, \nabla f(x_k) \rangle$ .
2. Calcul du pas :  $\gamma_k = \arg \min_{\gamma \in [0,1]} f((1-\gamma)x_k + \gamma s_k)$ .
3. Mise à jour :  $x_{k+1} = (1-\gamma_k)x_k + \gamma_k s_k$ .

Ici, nous instancions l'algorithme de Frank-Wolfe pour le Problème (2). La fonction objectif à minimiser est définie par :  $f(x) = \frac{1}{2} \|y - \Phi x\|_2^2$ , et l'ensemble convexe est quand à lui défini par :  $\mathcal{C} = \{x : \|x\|_1 \leq \beta\}$ .  $\mathcal{C}$  est la boule  $l_1$  de rayon  $\beta$ . Dans tout ce qui suit, nous la noterons  $\mathcal{B}_1(\beta)$ . Remarquons que  $\mathcal{B}_1(\beta)$  peut être écrite comme l'enveloppe convexe  $\mathcal{B}_1(\beta) = \text{conv}\{\pm \beta e_i | i \in \{1, \dots, n\}\}$ , où  $e_i$  est le  $i^{\text{ème}}$  vecteur de la base canonique. Comme  $\nabla f(x) = \Phi^t(\Phi x - y)$ , l'étape de sélection d'atome de Frank-Wolfe s'écrit :

$$s_k = \arg \min_{s \in \text{conv}\{\pm \beta e_i | i \in \{1, \dots, n\}\}} \langle s, \Phi^t(\Phi x_k - y) \rangle.$$

Comme le problème d'optimisation est linéaire et que  $\mathcal{B}_1(\beta)$  est un fermé borné, l'ensemble des solutions contient un point extremal de  $\mathcal{B}_1(\beta)$  :

$$s_k = \arg \min_{s \in \{\pm \beta e_i | i \in \{1, \dots, n\}\}} \langle s, \Phi^t(\Phi x_k - y) \rangle.$$

Vu que  $s = \pm \beta e_i$  alors  $\Phi s = \pm \beta \varphi_i$ . La sélection du point de direction de descente de l'algorithme de Frank-Wolfe obtenue est décrite en lignes 3 et 4 de l'Algorithme 1. Rappelons que le résidu  $r_k$  est défini comme :

$$r_k = y - \Phi x_k.$$

Nous remarquons que l'étape de sélection d'atome de l'algorithme de Frank-Wolfe est la même que celle de MP et OMP :

$$i_k = \lambda_k = \arg \max_i |\langle \varphi_i, r_k \rangle|.$$

Pour finir, en initialisant  $x_0$  à 0 (qui appartient bien à  $\mathcal{B}_1(\beta)$ ), nous obtenons l'instanciation de l'algorithme de Frank-Wolfe au Problème (2) décrite dans l'Algorithme 1.

## 3 Reconstruction exacte et convergence exponentielle

Dans ce qui suit, nous nous intéresserons aux propriétés de reconstruction et de convergence de l'Algorithme 2 dans le cas de dictionnaire *quasi-incohérent* ( $m < \frac{1}{2}(\mu^{-1} + 1)$ ). Pour un tel dictionnaire, tout sous-ensemble d'au plus  $m$  atomes ( $\{\varphi_i | i \in \Lambda\}$  tel que  $|\Lambda| \leq m$ ) est linéairement indépendant.

---

**Algorithme 1** : algorithme de Frank-Wolfe pour Pb. (2)

---

**Données :**  $y, \Phi = [\varphi_1, \dots, \varphi_n], \beta > 0$ .  
1  $x_0 = 0, k = 0$   
2 **tant que** critère d'arrêt non vérifié **faire**  
3      $i_k = \arg \max_{i \in \{1, \dots, n\}} |\langle \varphi_i, y - \Phi x_k \rangle|$   
4      $s_k = \text{signe}(\langle \varphi_{i_k}, y - \Phi x_k \rangle) \beta e_{i_k}$   
5      $\gamma_k = \arg \min_{\gamma \in [0, 1]} \|y - \Phi(x_k + \gamma(s_k - x_k))\|_2^2$   
6      $x_{k+1} = x_k + \gamma_k(s_k - x_k)$   
7      $k = k + 1$   
8 **fin**

---

Pour tout signal  $m$ -parcimonieux  $y$ , le vecteur de coefficients  $x^*$  ( $\|x^*\|_0 \leq m$ ) tel que  $y = \Phi x^*$  ainsi que son support sont alors uniques. Dans ce cas,  $x^*$  est l'unique solution du problème m-EXACT-SPARSE et du Problème (1). De plus,  $x^*$  est solution du Problème (2) si et seulement si, elle est faisable (i.e.  $\|x^*\|_1 \leq \beta$ ). Avant d'aller plus loin, définissons quelques notations. Pour tout signal  $m$ -parcimonieux  $y = \Phi x^*$ , notons par  $\Lambda_{opt}$  son support i.e.  $y = \sum_{i \in \Lambda_{opt}} x^*[i] \varphi_i$  tel que  $|\Lambda_{opt}| \leq m$ . Pour tout  $\Lambda \subset \{1, \dots, n\}$ ,  $\Phi_\Lambda$  est la matrice dont les colonnes sont les atomes indexés par  $\Lambda$ . Lorsque  $\Lambda$  est le support  $\Lambda_{opt}$ , nous notons  $\lambda_{min}^*$  (resp.  $\lambda_{max}^*$ ) sa plus petite (resp. la plus grande) valeur singulière. Pour toute matrice  $\Phi$ ,  $\text{span}(\Phi)$  est l'espace vectoriel engendré par ses colonnes. Pour finir, lors de l'étude de convergence de l'Algorithme 1, nous considérerons la norme au carré du résidu qui est liée à la fonction objectif de la manière suivante :

$$f(x_k) = \frac{1}{2} \|y - \Phi x_k\|_2^2 = \frac{1}{2} \|r_k\|_2^2.$$

Le Théorème 1 présente la propriété de reconstruction de l'algorithme de Frank-Wolfe, tandis que le Théorème 2 aborde la question de vitesse de convergence de l'algorithme.

Tropp [2], Gribonval et Vandergheynst [3] démontrent que lorsque le dictionnaire est quasi-incohérent, les algorithmes MP et OMP reconstruisent exactement tout signal  $m$ -parcimonieux. La preuve de leurs résultats réside sur le fait qu'à chaque itération, les algorithmes sélectionnent uniquement un atome du support. Nous étendons ce résultat à l'algorithme de Frank-Wolfe dans le Théorème 1.

**Théorème 1.** Soit  $\Phi$  un dictionnaire de cohérence  $\mu$ , et  $y$  un signal  $m$ -parcimonieux de support  $\Lambda_{opt}$ . Si  $m < \frac{1}{2}(\mu^{-1} + 1)$ , alors à chaque itération l'Algorithme 1 sélectionne un atome du support (i.e.  $\forall k, i_k \in \Lambda_{opt}$ ).

La preuve de ce théorème est très similaire à celle de [2] (voir [8] pour le détail). Lorsque la solution  $x^*$  est solution du problème (i.e.  $\|x^*\|_1 \leq \beta$ ), nous pouvons démontrer que les itérées  $x_k$  convergent vers celle-ci.

**Corollaire 1.** Soit  $\Phi$  un dictionnaire de cohérence  $\mu$ , et  $y$  un signal  $m$ -parcimonieux de support  $\Lambda_{opt}$ . Si  $m < \frac{1}{2}(\mu^{-1} + 1)$  et  $\|x^*\|_1 \leq \beta$  alors la séquence des  $x_k$  produite par l'Algorithme 1 converge vers  $x^*$ .

De manière générale, l'algorithme de Frank-Wolfe converge exponentiellement à partir d'une certaine itération lorsque la

fonction objectif est fortement convexe [4], et linéairement dans les autres cas. Le Théorème 2 montre que lorsque le dictionnaire est quasi-incohérent, l'algorithme de Frank-Wolfe converge exponentiellement à partir d'une certaine itération même si la fonction objectif n'est pas fortement convexe.

**Théorème 2.** Soit  $\Phi$  un dictionnaire,  $\mu$  sa cohérence,  $\mu_1$  sa fonction de Babel, et  $y$  un signal  $m$ -parcimonieux. Si  $m < \frac{1}{2}(\mu^{-1} + 1)$  et  $\|x^*\|_1 \leq \beta$  alors il existe une étape  $K$  telle que pour toute itération  $k \geq K$  de l'Algorithme 1 :

$$\|r_{k+1}\|_2^2 \leq \|r_k\|_2^2 (1 - \theta)$$
$$\text{où } \theta = \frac{1}{16} \left( \frac{1 - \mu_1(m-1)}{m} \right) \left( 1 - \frac{\|x^*\|_1}{\beta} \right)^2.$$

**Remarque 1.** Remarquons que  $0 < \theta \leq 1$  car  $m < \frac{1}{2}(\mu^{-1} + 1)$  implique  $0 < \mu_1(m-1) < 1$ .

**Remarque 2.** Le même résultat est obtenu lorsque les conditions ERC [2] sont vérifiées,  $\mu_1(m-1) < 1$  et  $\|x^*\|_1 < \beta$ .

Nous donnons ici les grandes lignes de la preuve (c.f. [8]).

**Idée de preuve** (Théorème 2). La première étape de la preuve est de montrer que lorsque le dictionnaire est quasi-incohérent, le pas  $\gamma_k$  est dans  $]0, 1[$ . L'expression de  $\|r_{k+1}\|_2^2$  devient alors :

$$\|r_{k+1}\|_2^2 = \|r_k\|_2^2 - \frac{\langle r_k, \Phi(s_k - x_k) \rangle^2}{\|\Phi(s_k - x_k)\|_2^2}.$$

D'après le Corollaire 1, la suite des  $x_k$  converge vers  $x^*$ . Il existe donc une itération  $K$  telle que  $\forall k \geq K : x_k \in \mathcal{B}_2(x^*, \epsilon)$  où  $\mathcal{B}_2(x^*, \epsilon)$  est la boule  $l_2$  centrée en  $x^*$  et de rayon  $\epsilon$ . Donc  $x_k - \epsilon \frac{\nabla f(x_k)}{\|\nabla f(x_k)\|} \in \mathcal{B}_2(x^*, 2\epsilon) \subseteq \mathcal{B}_1(\beta)$ . Par définition de  $s_k$  :

$$\langle s_k, \nabla f(x_k) \rangle \leq \langle x_k - \epsilon \frac{\nabla f(x_k)}{\|\nabla f(x_k)\|}, \nabla f(x_k) \rangle.$$

Or  $\langle r_k, \Phi(s_k - x_k) \rangle \geq \epsilon \|\Phi^t r_k\|$  car  $\nabla f(x_k) = -\Phi^t r_k$ . Par le Théorème 1, nous avons que  $r_k \in \text{span}(\Lambda_{opt})$ , et par [2] que  $\lambda_{min}^* \geq (1 - \mu_1(m-1))$ . D'où :

$$\langle r_k, \Phi(s_k - x_k) \rangle \geq \epsilon(1 - \mu_1(m-1)) \|r_k\|_2.$$

Pour finir, on utilise :  $\|\Phi(s_k - x_k)\|_2 \leq 2\beta\sqrt{d}$ .

**Remarque 3.** Nous prouvons aussi dans [8] que pour  $\beta$  assez grand, la convergence est exponentielle dès la première itération.

## 4 Expériences numériques

Nous réalisons des expériences sur données synthétiques nous permettant de tester la minimalité des conditions du Théorème 2.

Nous considérons des données de taille  $d = 1000$  et un dictionnaire de 2000 atomes. Le dictionnaire est l'union d'une DCT-II et de l'identité. Il a une faible cohérence ( $\mu = 4.5 \times 10^{-2}$ ), et  $m^* = \lceil \frac{1}{2}(\mu^{-1} - 1) \rceil = 11$  est la plus grande valeur de parcimonie vérifiant la condition du Théorème 2. Les supports sont tirés aléatoirement uniformément, et les coefficients

selon une loi normale. Pour chaque expérience, l'Algorithme 1 est appliqué pour 10000 réalisations de tels signaux. Dans le Théorème 2, la convergence exponentielle est donnée par :

$$\|r_k\|_2^2 \leq \|r_{k-1}\|_2^2(1 - \theta),$$

ce qui est équivalent à :

$$\log \|r_k\|_2^2 \leq \log \|y\|_2^2 + k \log(1 - \theta).$$

Pour illustrer les expériences, nous traçons  $\log \|r_k\|_2^2$  en fonction de  $k$ . La convergence est exponentielle lorsque  $\log \|r_k\|_2^2$  est dominée par une droite de pente négative.

Pour la première expérience, fixons  $\beta = 8\|x^*\|_1$ , et  $m = m^*$ . La Figure 1 montre la moyenne et le maximum de la fonction  $\log \|r_k\|_2^2$  pour 10000 réalisations. Nous remarquons que la moyenne et le maximum sont dominés par des droites de pente négative, et donc convergent exponentiellement. Notons que contrairement à ce que laisse croire la Figure 1, la borne théorique est elle aussi dominée par une droite de pente négative ( $2 \times 10^{-3}$  contre  $3.8 \times 10^{-1}$  pour la courbe de la moyenne). Nous remarquons aussi qu'en pratique, la moyenne et le maximum sont plus petits que la borne théorique du Théorème 2. Ceci suggère que la borne théorique pourrait être améliorée.

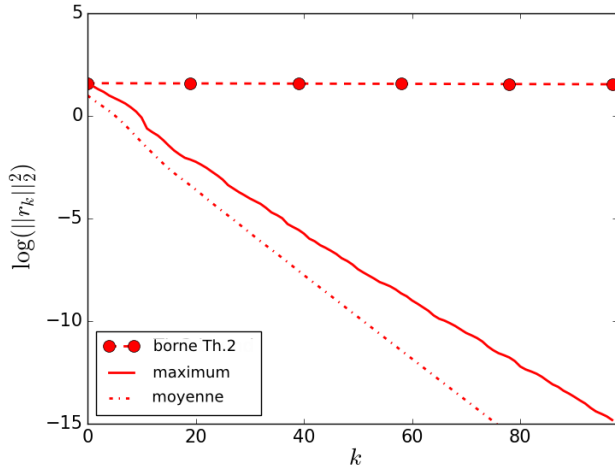


FIGURE 1 – Evolution de  $\log \|r_k\|_2^2$  dans l'Algorithme 1, pour  $\beta = 8\|x^*\|_1$ , et  $m = m^* = \lceil \frac{1}{2}(\mu^{-1} - 1) \rceil$ .

Dans la seconde expérience, nous vérifions si la convergence est toujours exponentielle pour des signaux de parcimonie supérieure à  $m^*$  (i.e. lorsque la condition du Théorème 2 n'est plus vérifiée). Fixons  $\beta = 8\|x^*\|_1$  et traçons dans la Figure 2,  $\log \|r_k\|_2^2$  pour  $m = m^*, 2m^*, 5m^*$  et  $20m^*$ . Nous remarquons que pour des parcimonies de  $2m^*, 5m^*$ , et  $20m^*$  la convergence est toujours exponentielle. Néanmoins, la vitesse de convergence décroît lorsque la valeur de  $m^*$  augmente. Cela suggère que la condition  $m \leq m^*$  du Théorème 2 pourrait elle aussi être améliorée.

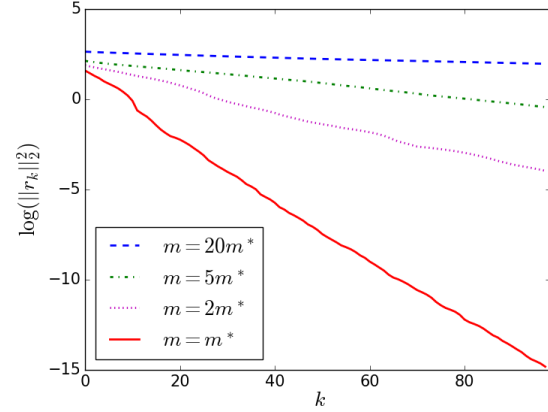


FIGURE 2 – Évaluation du maximum de  $\log \|r_k\|_2^2$  pour  $\beta = 8\|x^*\|_1$  et pour différentes valeurs de  $m$ .

## 5 Conclusion

Dans ce travail nous avons étudié les propriétés de l'algorithme de Frank-Wolfe pour la résolution du problème m-EXACT-SPARSE. Nous avons démontré que lorsque le dictionnaire est quasi-incohérent, tout comme les algorithmes MP et OMP, l'algorithme de Frank-Wolfe sélectionne uniquement des atomes du support. Sous les mêmes conditions, l'algorithme de Frank-Wolfe converge exponentiellement. Les expériences numériques illustrent ces résultats et laissent à penser que la condition sur la parcimonie pourrait être améliorée. On peut également se demander si ces résultats peuvent être étendus au cas de signaux non exactement parcimonieux (ou encore compressibles).

## Références

- [1] M. Frank, et P. Wolfe. *An algorithm for quadratic programming*. Naval Research Logistics, 1956.
- [2] J. Tropp. *Greed is good : Algorithmic results for sparse approximation*. IEEE Trans. on Information theory, 2004.
- [3] R. Gribonval, et P. Vandergheynst. *On the exponential convergence of matching pursuit in quasi-incoherent dictionaries*. IEEE Trans. on Information theory, 2006.
- [4] J. Guélat et P. Marcotte. *Some comments on wolfe's 'away step'*. Math. Programming, 35(1) :110–119, 1986.
- [5] G. Davis, S. Mallat, et M. Avellaneda. *Adaptive greedy approximations*. Constructive approximation 1997.
- [6] R. Tibshirani. *Regression shrinkage and selection via the lasso*. Journal of the Royal Statistical Society 1996.
- [7] S. Chen, D. Donoho, et M. Saunders. *Atomic decomposition by basis pursuit*. SIAM review 2001.
- [8] F. Cherfaoui, V. Emiya, L. Ralaivola, and S. Anthoine. *Recovery and convergence rate of the frank-wolfe algorithm for the m-exact-sparse problem*. To appear in IEEE Transactions on Information Theory, 2019.