

# Algorithmes STAP récursifs sur la distance pour radar aéroporté en configuration non latérale

## Range recursive STAP algorithms for airborne radar in non side looking configuration

**Sophie Beau, Sylvie Marcos**

Laboratoire des Signaux et systèmes, CNRS, Supelec, 3, rue Joliot-Curie 91192 Gif sur Yvette Cedex  
sophie.beau@lss.supelec.fr, sylvie.marcos@lss.supelec.fr

Tel. : +33 (0)1 69 85 17 29

Fax : +33 (0)1 69 85 17 65

Manuscrit reçu le 7 février 2008

Résumé et mots clés

Cet article traite de la détection de cibles mobiles sur fond de fouillis dans le cadre de radar aéroporté monostatique utilisant trois formes d'antennes : linéaire uniforme (ALU) ; courbée uniforme (ACoU) et circulaire uniforme (ACU), en visée non latérale. Le traitement STAP classique, tel que la méthode SMI (*Sample Matrix Inversion*) ou tel que les méthodes de type *eigencanceler*, nécessitent une estimation de la matrice de covariance des interférences plus bruit en moyennant les données provenant des cases distances secondaires. Outre le fait que ces méthodes sont très coûteuses en calculs, leurs performances ne sont pas optimales (l'estimateur de la matrice de covariance des interférences plus bruit est biaisé) dans le cas où les données ne sont pas iid. C'est le cas, par exemple, pour les radars utilisant une ALU en visée non latérale et ceux utilisant une ACoU ou ACU. En effet, dans ces configurations, la densité spectrale du fouillis présente une dépendance en distance qui introduit une non stationnarité des données. Notre but ici est de mettre en valeur la capacité de poursuite de cette non-stationnarité en distance par des algorithmes récursifs à complexité calculatoire linéaire et ce, sans utilisation de méthodes de compensation qui s'avèrent coûteuses en temps de calcul et/ou nécessitent la connaissance des paramètres du radar.

Techniques STAP, algorithmes récursifs, radar aéroporté, configuration non latérale, fouillis non stationnaire en distance.

Abstract and key words

We address the problem of detecting slow moving target within ground clutter with a non side looking monostatic airborne radar using either a uniformly spaced and linear antenna (ULA) or a uniformly curved or circular antenna (UCuA and UCA respectively). To obtain the optimal STAP filter, the space time covariance matrix has to be inverted [1]. In practice, it is unknown and has to be estimated. With a monostatic radar using an ULA in side looking configuration, the clutter has specific properties : the space-time repartition of the clutter spectral power which is called "Clutter Ridge" is range independent. The space-time snapshots are thus iid and the classical SMI (*Sample Matrix Inversion*) estimator can be used [1]. But in case of a monostatic radar using an ULA in a non side looking configuration or using a UcuA or a UCA, the clutter properties are changed. The clutter ridges are range dependent and specific methods must be applied to compensate the range dependency [4], [5], [6]. But these methods are computational expensive and/or require the knowledge of the radar parameters. In this paper, we are rather interested in range-recursive subspace-based algorithms of linear computational cost. First, we establish the range dependency of the model of the clutter ridges

when using ULA, UCuA and UCA in a non side looking configuration. We then present the various algorithms which are investigated. Some are based on a quadratic cost function : PAST[10], OPAST[11], API[12] and FAPI[12] algorithms, whereas another is based on a non quadratic cost function : NIC algorithm [13]. They were initially used in array processing. We here show the capability of these recursive algorithms to track the range non stationarity of the data without the use of compensation methods. We compare them to the optimal STAP filter and to the SMI and DBU algorithms in term of SINR loss. Simulations exhibit the high performance of the proposed approach to track the range non stationarity implied as well by the geometry of the antenna array as by the configuration of the radar : we here investigate both side looking and non side looking configurations. Contrary to the DBU algorithm, the range recursive algorithms converge with a short set of training data ( $2NM$  range cells are sufficient whereas DBU requires  $4NM$  range cells).

STAP, recursive algorithms, airborne radar, non side looking configuration, range dependent clutter.

# 1. Introduction

La détection de cibles mobiles sur fond de fouillis est le problème majeur dans le domaine du radar aéroporté air/sol. Lorsque la vitesse de la cible est faible, cette détection est fortement dégradée du fait du déplacement du radar, le fouillis possédant alors un Doppler non nul. Ainsi, la fréquence Doppler de la cible à faible vitesse peut être masquée par le spectre Doppler du fouillis (cf. Figure 1)). Notre but est donc de créer un filtre spatio-temporel (STAP) réjecteur du fouillis [1]. Idéalement, le filtre STAP optimal au sens de la variance minimale est obtenu en fonction de la matrice de covariance du fouillis. Dans le cadre d'un radar monostatique (l'émetteur et le récepteur sont sur le même porteur) utilisant une antenne linéaire uniforme (ALU) et pour une configuration latérale ou *sidelooking* (i.e. le vecteur vitesse du porteur est colinéaire à l'axe de l'antenne radar), cette matrice et donc le filtre STAP qui en découle peuvent être estimés à partir des signaux mesurés sur les cellules distances secondaires donnant lieu à la méthode SMI : *Sample Matrix Inversion* [1]. Par contre, pour les radars utilisant une ALU dans la configuration non latérale ou *non sidelooking* (i.e. le vecteur vitesse du porteur n'est pas colinéaire à l'axe de l'an-

tenne radar) ou ceux utilisant une antenne de forme quelconque comme par exemple une antenne courbée ou circulaire uniforme (ACoU, ACU), les propriétés du fouillis sont modifiées, notamment une dépendance en distance de la matrice de covariance est observée introduisant une non stationnarité sur les données. Il existe des méthodes de compensation de cette non stationnarité en distance [6] mais leur mise en oeuvre est très contraignante de par leur complexité calculatoire qui s'avère rédhibitoire pour la réalisation du traitement en temps réel, et/ou du fait qu'elles nécessitent la connaissance des paramètres du radar. Nous proposons ici, comme alternative, d'exploiter la capacité de poursuite des algorithmes adaptatifs récursifs initialement introduits dans le domaine temporel et plus récemment dans le domaine spatio-temporel [9]. La modélisation du problème incluant la géométrie des différentes architectures radar (ALU, ACoU et ACU) sera donnée en section 2, montrant ainsi la dépendance en distance engendrée sur les données. Puis les algorithmes récursifs sur la distance seront présentés et détaillés en section 4. Nous présenterons en section 5 des simulations illustrant leur capacité à poursuivre cette non stationnarité (dépendance en distance) des données. Enfin, nos remarques, conclusions ainsi que nos perspectives seront données en section 6.

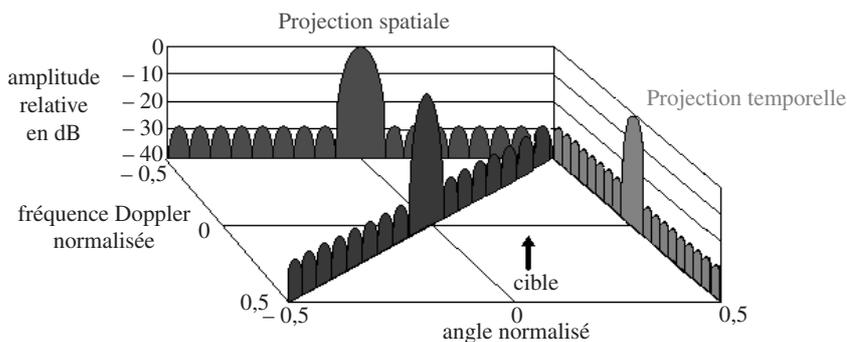


Figure 1. Densité spectrale de puissance du fouillis.

## 2. Modélisation des signaux

### 2.1. Modélisation des signaux reçus par une antenne linéaire uniforme

Nous considérons une antenne radar linéaire (cf. Figure 2) composée de  $N$  capteurs espacés uniformément d'une distance  $d$  équivalente à une demi longueur d'onde. Le radar émet un train de  $M$  impulsions à fréquence de répétition des impulsions (PRF) constante et est porté par une plate-forme se trouvant à l'altitude  $h$  et de vitesse  $\mathbf{v}_a$ . Les signaux reçus sur chacun des capteurs et provenant de chaque impulsion sont collectés à chaque distance  $k$ . Nous supposons l'absence de brouilleur. Le vecteur spatio-temporel reçu  $\mathbf{s}$  s'exprime par :

$$\mathbf{x}(k) = \alpha_t \cdot \mathbf{v}(\vartheta_t, \varpi_t) + \mathbf{x}_c + \mathbf{x}_b$$

où  $\alpha_t$  est l'amplitude de la cible,  $\mathbf{v}(\vartheta_t, \varpi_t)$  est le vecteur directionnel spatio-temporel de la cible et s'écrit comme

$$\mathbf{v}(\vartheta_t, \varpi_t) = \mathbf{b}(\varpi_t) \otimes \mathbf{a}(\vartheta_t) \quad (1)$$

où  $\otimes$  est le produit de Kronecker

$$\mathbf{b}(\varpi_t) = [1; e^{j2\pi\varpi_t}; \dots; e^{j(M-1)2\pi\varpi_t}] \quad (2)$$

est le vecteur temporel directionnel où  $\varpi_t$  est la fréquence Doppler normalisée de la cible et

$$\mathbf{a}(\vartheta_t) = [1; e^{j2\pi\vartheta_t}; \dots; e^{j(N-1)2\pi\vartheta_t}] \quad (3)$$

est le vecteur spatial directionnel où  $\vartheta_t$  est la fréquence spatiale de la cible.  $\mathbf{x}_b$  est la composante de bruit supposé Gaussien et blanc spatialement et temporellement. Enfin le vecteur spatio-temporel du fouillis est modélisé de la manière suivante : le lieu de provenance des signaux de fouillis est divisé en anneaux situés à la même distance  $R_c$  du radar, chaque anneau est lui-même subdivisé en  $N_c$  éléments et chaque élément de fouillis est localisé par son azimut noté  $\phi_c$  et son élévation notée  $\theta_c$ . Ainsi la composante de fouillis est de la forme :

$$\mathbf{x}_c = \sum_{i=1}^{N_c} \alpha_i \mathbf{v}(\vartheta_i, \varpi_i)$$

où  $\alpha_i$  est l'amplitude du signal provenant du  $i^{\text{ème}}$  élément de fouillis et  $\mathbf{v}$  est le vecteur spatio-temporel du fouillis défini de la même manière qu'en (1), (2) et (3) avec  $\vartheta_i$  et  $\varpi_i$  qui sont respectivement les fréquence spatiale et fréquence Doppler normalisée correspondant au  $i^{\text{ème}}$  élément de fouillis.

Le radar est en configuration non latérale lorsque le vecteur vitesse de la plate-forme n'est pas parfaitement aligné avec l'axe du radar. Ceci peut se produire en pratique dans deux cas : soit l'antenne radar n'est pas montée de côté (elle peut par

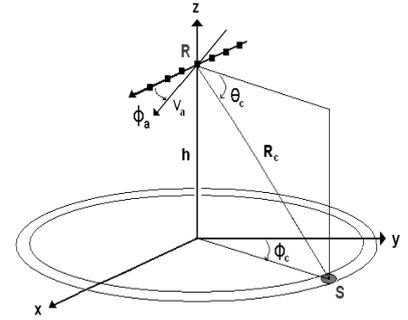


Figure 2. Géométrie d'un radar monostatique en configuration non sidelooking.

exemple être montée de face ou bien être montée de côté et être rotative), soit l'antenne est bien à visée latérale mais pour compenser les effets du vent, l'axe de la plate-forme doit être légèrement incliné afin de voler à l'horizontale malgré tout. L'angle formé alors entre le vecteur vitesse de la plate-forme et l'axe de l'antenne, appelé angle de crabe, est noté  $\phi_a$ .

La fréquence spatiale du fouillis s'écrit :

$$\vartheta_c = \frac{d \cdot \cos(\theta_c)}{\lambda} \sin(\phi_c) \quad (4)$$

et la fréquence Doppler normalisée

$$\varpi_c = \frac{2v_a T_r}{\lambda} \cos(\theta_c) \sin(\phi_c + \phi_a) \quad (5)$$

où  $\lambda$  est la longueur d'onde du radar et  $T_r$  est l'intervalle de répétition des impulsions (PRI). Soit  $\delta$  tel que  $\cos(\delta)$  soit le cosinus directeur entre l'axe  $\mathbf{x}$  et le vecteur unitaire de pointage de  $\mathbf{R}$  vers l'élément de fouillis  $\mathbf{S}$  (cf. Figure 2) :  $\cos(\delta) = \cos(\theta_c) \sin(\phi_c + \phi_a)$ . Et donc

$$\sin(\phi_c) = \frac{\cos(\phi_a) \cdot \cos(\delta)}{\cos(\theta_c)} \pm \sqrt{\left(\frac{\cos(\delta)}{\cos(\theta_c)}\right)^2 \cos^2(\phi_a) - \left(\frac{\cos(\delta)}{\cos(\theta_c)}\right)^2 + \sin^2(\phi_a)} \quad (6)$$

En supposant le rayon de courbure de la Terre infini (modèle plat), l'élévation peut être approximée de la manière suivante  $\sin(\theta_c) \approx \frac{h}{R_c}$  alors, (4) et (6) donnent

$$\vartheta_c = \frac{d}{\lambda} \left( \cos(\delta) \cos(\phi_a) \pm \sin(\phi_a) \sqrt{1 - \left(\frac{h}{R_c}\right)^2 - \cos^2(\delta)} \right) \quad (7)$$

De plus la fréquence Doppler normalisée peut s'exprimer comme

$$\varpi_c = \frac{d}{\lambda} \cos(\delta) \quad (8)$$

Ainsi nous obtenons la relation suivante entre les fréquences spatiale et Doppler normalisée du fouillis

$$v_c = \varpi_c \cos(\phi_a) \pm \frac{1}{2} \sin(\phi_a) \sqrt{1 - \left(\frac{h}{R_c}\right)^2 - 4\varpi_c^2} \quad (9)$$

La dépendance en distance impliquée par la configuration non latérale est entièrement due à la composante de fouillis. Nous remarquons qu'en configuration latérale, lorsque  $\phi_a = 0^\circ$ , la relation liant les fréquences spatiales et Doppler est indépendante de la distance  $R_c$ . Par contre, dans le cas d'une configuration non latérale, une dépendance en distance est observée. Par conséquent, la matrice de covariance du fouillis est aussi dépendante en distance. La répartition spatio-temporelle de la puissance de fouillis (*Clutter ridges*) en fonction de la fréquence Doppler et de la fréquence spatiale est représentée sur la Figure 3 pour différentes valeurs de la distance. Nous remarquons d'autre part que la non stationnarité diminue quand la distance augmente.

### 2.2. Modélisation des signaux reçus par une antenne circulaire ou courbée uniforme

Dans cette section, nous nous intéressons à un radar utilisant une antenne plane courbée ou circulaire uniforme. La Figure 4 montre la géométrie de l'antenne. Les caractéristiques de l'antenne radar (nombre de capteurs, nombre d'impulsions par CPI) restent les mêmes que précédemment. La différence réside en la courbure de l'antenne. En effet, pour l'analyse, nous pouvons choisir la courbure de l'antenne au travers d'un angle d'ouverture  $\gamma$ , la rayon du cercle qui contient les capteurs varie avec cet angle au travers de la relation :  $R = (N - 1)\lambda / (2\gamma)$ . Donc plus l'angle  $\gamma$  est petit, plus la courbure de l'antenne est faible (et plus le rayon du cercle est grand) et au contraire, plus l'angle d'ouverture est grand, plus l'antenne est courbée. La Figure 4 illustre le cas où l'angle d'ouverture vaut  $\pi$ . Le vecteur spatio-

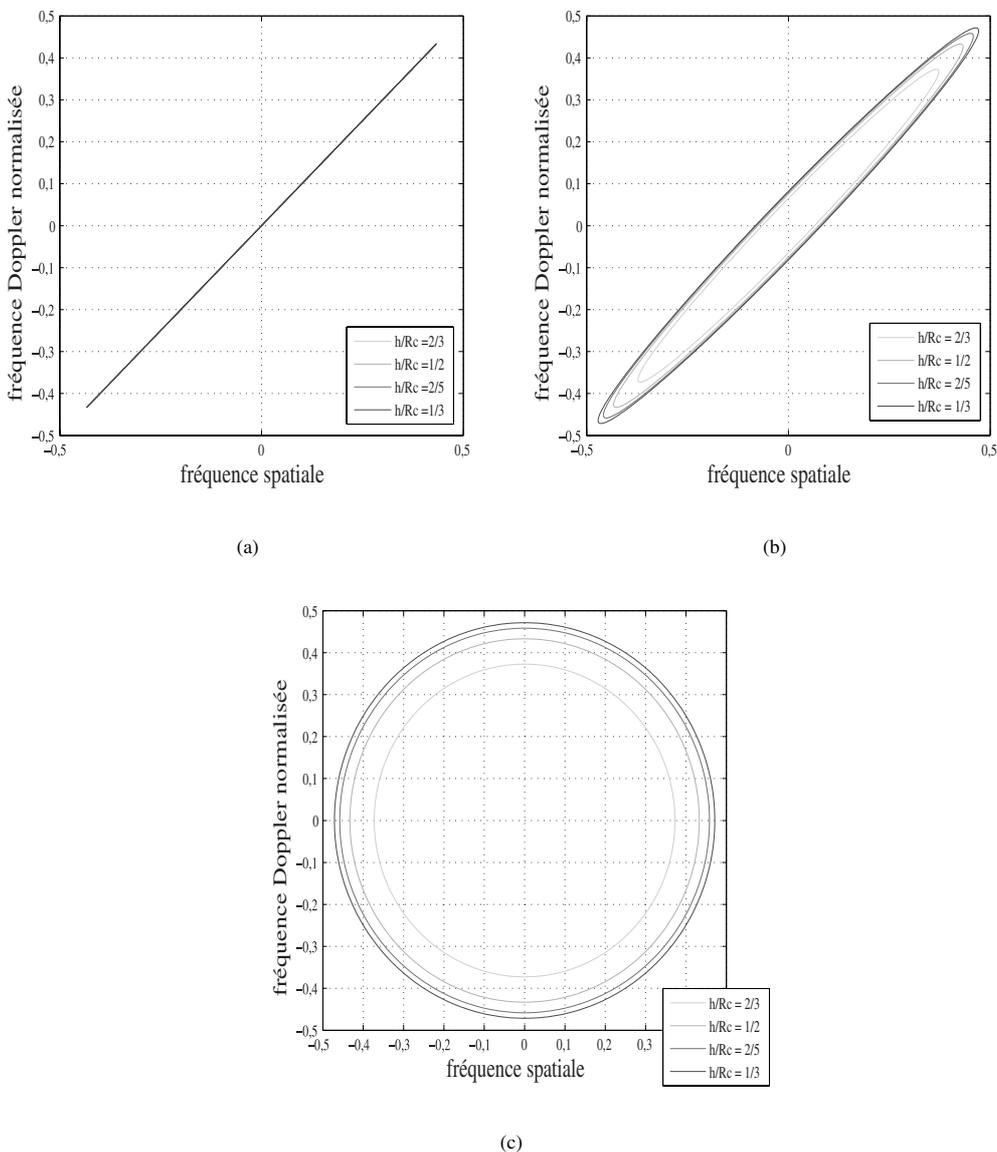


Figure 3. Exemples de Clutter ridges dans le cas où  $\phi_a = 0^\circ$  (a) ;  $\phi_a = 10^\circ$  (b),  $\phi_a = 90^\circ$  (c).

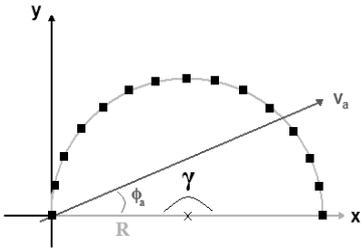


Figure 4. Antenne courbée ou circulaire vue du dessus.

temporel reçu contient toujours une composante de cible éventuellement (si la cible est présente à cette distance), une composante de fouillis et une composante de bruit. Nous rappelons que nous supposons l'absence de brouilleurs. La composante de fouillis est définie par

$$\mathbf{x}_c = \sum_{i=1}^{N_c} \alpha_i \mathbf{v}_c(\varpi_i, \phi_i, \theta_i)$$

où comme précédemment,  $\alpha_i$  est l'amplitude du signal émanant du  $i^{\text{ème}}$  élément de fouillis et  $\mathbf{v}_c$  est le vecteur directionnel spatio-temporel s'écrivant toujours comme le produit de Kronecker entre le vecteur directionnel temporel et le vecteur directionnel spatial. Comme dans le cas d'une ALU, le vecteur directionnel temporel dépend de la fréquence Doppler (cf. (2)). En effet sa  $m^{\text{ème}}$  composante est donnée par

$$\mathbf{b}_m(\varpi) = e^{j2\pi(m-1)\varpi}$$

où  $\varpi$ , la fréquence Doppler normalisée est définie par

$$\varpi = \frac{2\mathbf{k}(\phi, \theta) \mathbf{v}_a}{\lambda \cdot PRF} = \frac{2v_a}{\lambda PRF} \cos(\theta) \sin(\phi + \phi_a)$$

où  $v_a$  est la norme de  $\mathbf{v}_a$ . Dans le cas d'une ACU ou d'une AcoU, la  $m^{\text{ème}}$  composante du vecteur directionnel spatial s'exprime par

$$\mathbf{a}_m(\phi, \theta) = e^{j\frac{2\pi}{\lambda} \cdot \mathbf{k}(\phi, \theta) \cdot \mathbf{d}_m} \quad (10)$$

où  $\mathbf{k}$  est le vecteur unitaire pointant en direction  $(\phi, \theta)$  et s'exprimant comme

$$\mathbf{k}(\phi, \theta) = \cos(\theta) \sin(\phi) \mathbf{x} + \cos(\theta) \cos(\phi) \mathbf{y} + \sin(\theta) \mathbf{z}$$

$\mathbf{d}_m = d_{mx} \mathbf{x} + d_{my} \mathbf{y}$  est le vecteur de position du  $m^{\text{ème}}$  élément d'antenne dans le système de coordonnées cartésiennes (on considère une antenne plane). On obtient donc

$$\mathbf{a}_m(\phi, \theta) = e^{j\frac{2\pi}{\lambda} (d_{mx} \cdot \cos(\theta) \sin(\phi) + d_{my} \cdot \cos(\theta) \cos(\phi))}$$

Les coordonnées de l'antenne radar sont données par

$$\forall i \in [1; N] \quad d_{ix} = R \cdot \sin\left(\frac{\gamma}{2}\right) + R \cdot \sin\left(\frac{\gamma}{N-1} \left(-\frac{N-1}{2} + (i-1)\right)\right)$$

$$\forall i \in [1; N] \quad d_{iy} = \sqrt{\left(2R \cdot \sin\left(\frac{(i-1)\gamma}{2(N-1)}\right)\right)^2 - d_{ix}^2}$$

Contrairement au cas d'une ALU, dans le cas d'une antenne circulaire ou courbée, on ne peut pas parler de fréquence spatiale, nous devons, pour nous intéresser aux *clutter ridges* introduire une fréquence spatiale virtuelle [2]. Elle est définie à travers un vecteur directionnel spatial similaire à (3)

$$\mathbf{a}(\vartheta_{virtuelle}(\phi, \theta)) = [1; e^{j2\pi\vartheta_{virtuelle}(\phi, \theta)}; \dots; e^{j(N-1)2\pi\vartheta_{virtuelle}(\phi, \theta)}]$$

où  $\vartheta_{virtuelle}$  est la fréquence spatiale virtuelle définie comme

$$\vartheta_{virtuelle}(\phi, \theta) = \left(\sum_{n=0}^{N-1} n^2\right)^{\frac{1}{2}} \cdot \int_0^\phi \left\| \frac{\partial \mathbf{a}(\theta, u)}{\partial u} \right\| d(u)$$

où  $\mathbf{a}$  est le vecteur directionnel spatial défini en (3). La répartition spatio-temporelle théorique de la puissance de fouillis en fonction de la fréquence Doppler et de la fréquence spatiale virtuelle est représentée sur les Figures 5 et 6 et pour différentes valeurs de la distance. Pour la Figure 5, l'angle d'ouverture de l'antenne est de  $10^\circ$  et il est de  $180^\circ$  (antenne circulaire) pour la Figure 6. Nous remarquons que dans le cas où l'antenne est peu courbée, nous nous rapprochons des *clutter ridges* observées dans le cas d'une ALU. Dans tous les cas, nous pouvons conclure qu'en configuration latérale, les *clutter ridges* sont peu (ACoU ou ACU) ou pas dépendantes (ALU) de la distance et qu'en configuration latérale, elles sont dépendantes de la distance. Et cette dépendance en distance implique une non-stationnarité des données.

## 3. Traitement STAP et méthodes de compensation classiques

Le vecteur de pondération STAP optimal du filtre réjecteur des interférences et du bruit est donné par [1]

$$\mathbf{w}_{opt} = \kappa \mathbf{R}_{i+b}^{-1} \cdot \mathbf{v}_t$$

où  $\kappa$  est un scalaire de normalisation et  $\mathbf{R}_{i+b} = \mathbf{R}_c + \mathbf{R}_b$  est la matrice de covariance des interférences et du bruit. On suppose que les interférences ne sont constituées que de fouillis et que le fouillis et le bruit sont décorrélés.  $\mathbf{R}_c = E\{\mathbf{x}_c \mathbf{x}_c^H\}$  et  $\mathbf{R}_b = \sigma^2 \mathbf{I}$  sont respectivement les matrices de covariance de fouillis et de bruit. Théoriquement, le filtre optimal peut donc être calculé mais en pratique la matrice de covariance des interférences et du bruit n'est pas connue et doit donc être estimée à partir des signaux reçus. Une méthode d'estimation très classique est la

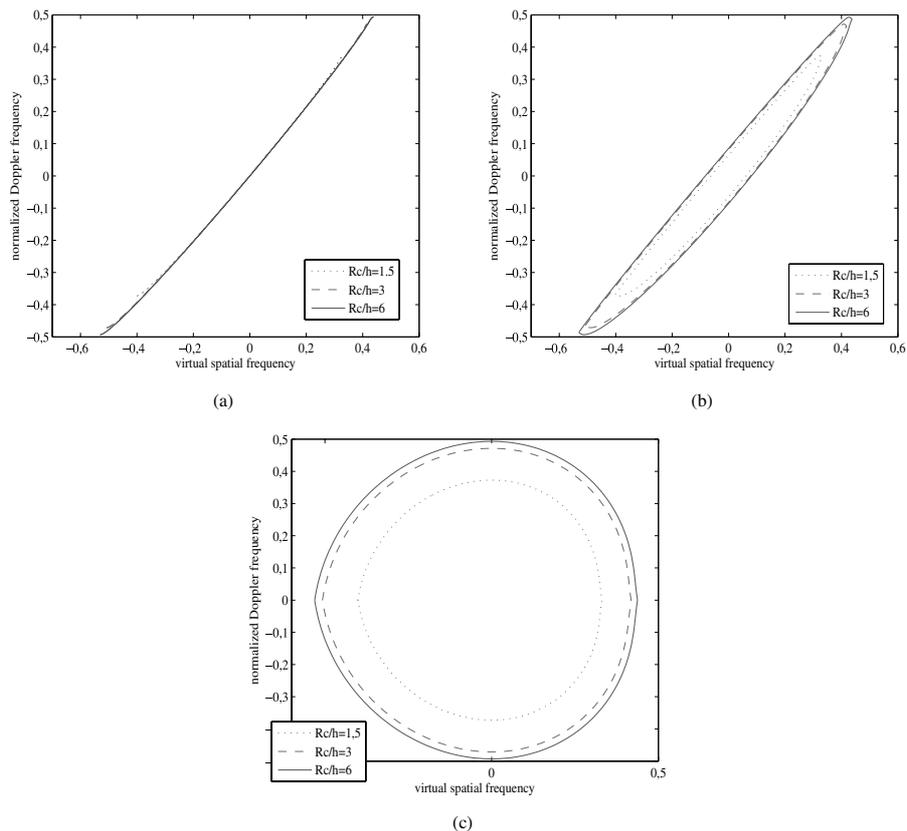


Figure 5. Exemples de Clutter ridges dans le cas d'une ACoU avec un angle d'ouverture de  $60^\circ$ , dans le cas où  $\phi_a = 0^\circ$  (a) ;  $\phi_a = 10^\circ$  (b) ;  $\phi_a = 90^\circ$  (c).

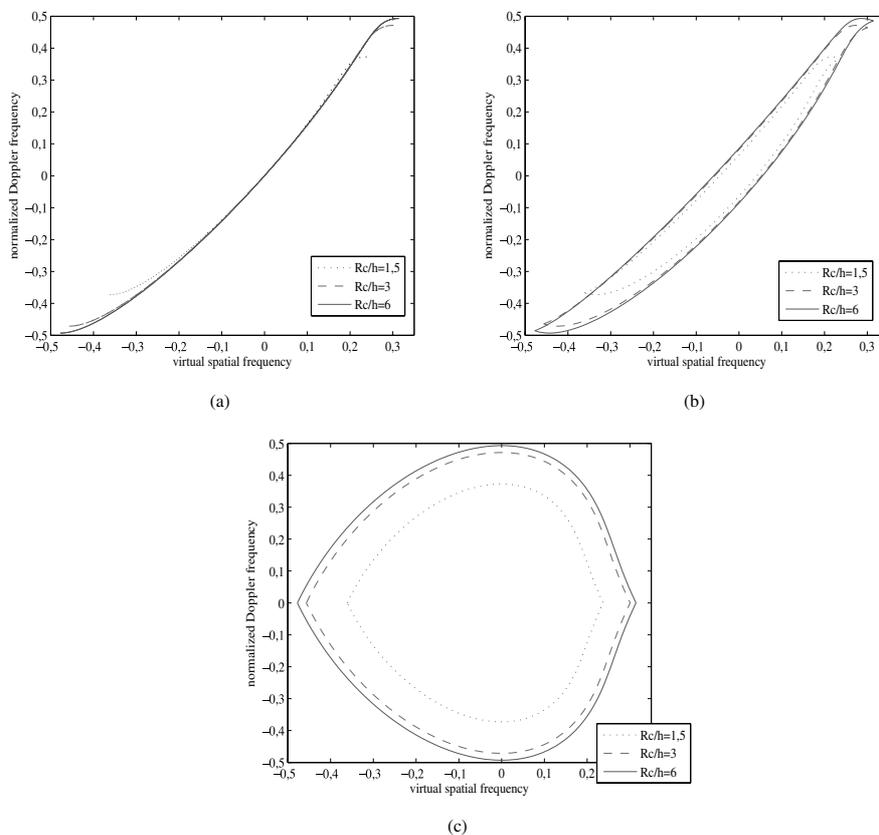


Figure 6. Exemples de Clutter ridges dans le cas d'une ACU avec un angle d'ouverture de  $180^\circ$ , dans le cas où  $\phi_a = 0^\circ$  (a) ;  $\phi_a = 10^\circ$  (b) ;  $\phi_a = 90^\circ$  (c).

méthode SMI (*sample matrix inversion*) [1], la matrice de covariance est estimée par moyennage des signaux reçus provenant des cellules distances secondaires,

$$\widehat{\mathbf{R}}_{i+b}(k) = \frac{1}{K} \sum_{l=1, l \neq k}^K \mathbf{x}_l \mathbf{x}_l^H \quad (11)$$

où  $k$  est la case distance de test et  $K$  est le nombre de cases distance utilisées. Le vecteur de pondération correspondant à cette méthode s'exprime par

$$\mathbf{w}_{smi} = \kappa \widehat{\mathbf{R}}_{i+b}^{-1} \mathbf{v}_t$$

Une autre approche basée sur les sous-espaces [3] exploitant la structure particulière de la matrice de covariance spatio-temporelle peut être utilisée pour construire le vecteur STAP de pondération. En effet, la matrice de covariance des interférences peut s'écrire

$$\mathbf{R}_{i+b} = \mathbf{E}_c \mathbf{\Lambda}_c \mathbf{E}_c^H + \sigma^2 \mathbf{I} \quad (12)$$

où la matrice  $\mathbf{\Lambda}_c$  est la matrice diagonale composée des  $r$  valeurs propres les plus grandes de la matrice  $\mathbf{R}_{i+b}$ ,  $\mathbf{\Lambda}_c = \text{diag}\{\lambda_1, \dots, \lambda_r\}$  où  $\lambda_1 \leq \dots \leq \lambda_r$ . Les vecteurs propres correspondants forment les colonnes de la matrice  $\mathbf{E}_c$  et engendrent le sous-espace interférence (ici fouillis)  $\text{esp}\{\mathbf{E}_c\}$ . Enfin  $\sigma^2$  est la variance du bruit et  $\mathbf{I}$  est la matrice identité. Le sous-espace complémentaire à  $\text{esp}\{\mathbf{E}_c\}$  dans l'espace des observations et qui est engendré par les  $MN - r$  valeurs propres les plus petites et égales à  $\sigma^2$  est le sous-espace bruit [3]. À partir du sous-espace fouillis, le vecteur STAP de pondération est formé en calculant  $\mathbf{w} = \mathbf{I} - \widehat{\mathbf{E}}_c \widehat{\mathbf{E}}_c^H$  où  $\widehat{\mathbf{E}}_c$  engendre le sous-espace fouillis estimé à partir de la matrice de covariance spatio-temporelle estimée  $\widehat{\mathbf{R}}_{i+b} = \widehat{\mathbf{E}}_c \widehat{\mathbf{\Lambda}}_c \widehat{\mathbf{E}}_c^H + \hat{\sigma}^2 \mathbf{I}$ . Cette méthode directe est très coûteuse en temps de calcul car il faut réaliser une décomposition en éléments propres. C'est pourquoi nous nous intéressons par la suite à une approche récursive sur la distance de cette méthode (cf. section 4).

Dans le cas d'une ALU, en configuration latérale, la relation entre la fréquence Doppler et la fréquence spatiale du fouillis est linéaire et indépendante de la distance. Ainsi, le lieu de la densité spectrale de puissance du fouillis est linéaire, les données reçues sont par conséquent iid. L'estimateur SMI, dans ce cas, n'est pas biaisé. Nous avons montré dans la section précédente que cette propriété d'indépendance en distance n'est plus vraie en configuration non latérale ou lorsque l'architecture de l'antenne est différente. C'est pourquoi des méthodes de compensation existent (cf. [6]), la compensation intervient sur différents éléments selon les méthodes: par exemple dans la méthode *Doppler Warping* (DW) [4], un point précis (à une fréquence Doppler et spatiale précise) des *Clutter Ridges* sont compensés en distance, dans la méthode *Angle-Doppler compensation* (ADC) [5], cette fois chaque point des *Clutter Ridges* est compensé. Mais sont très coûteuses en temps de calcul et/ou nécessitent des connaissances *a priori* sur le radar afin de pouvoir déterminer les points de référence pour les compensations.

Une méthode de compensation de référence est la méthode *Derivative Based Updating* (DBU) [7]. Elle est basée sur un développement en série de Taylor au premier ordre du vecteur de pondération STAP. Le vecteur de pondération est donc approximé à la distance  $k$  par

$$\mathbf{w}_k \approx \mathbf{w}_o + \alpha_k \Delta \mathbf{w}_o \quad (13)$$

où  $\alpha_k$  est un terme de normalisation du niveau de bruit (cf. [7]),  $\Delta \mathbf{w}$  représente la dérivée de  $\mathbf{w}$  par rapport à la distance. Les inconnues  $\mathbf{w}_o$  et  $\Delta \mathbf{w}_o$  sont calculées grâce à:

$$\tilde{\mathbf{w}} = \begin{bmatrix} \mathbf{w}_o \\ \Delta \mathbf{w}_o \end{bmatrix} = \widehat{\mathbf{R}}_E^{-1} \begin{bmatrix} \mathbf{v}_t \\ \mathbf{0} \end{bmatrix} \quad (14)$$

où l'estimée de la matrice de covariance étendue est donnée par

$$\widehat{\mathbf{R}}_E = \frac{1}{K} \sum_{l=1, l \neq k}^K \tilde{\mathbf{x}}_l \tilde{\mathbf{x}}_l^H \quad (15)$$

avec

$$\tilde{\mathbf{x}}_l = \begin{bmatrix} \mathbf{x}_l \\ \alpha_l \mathbf{x}_l \end{bmatrix} \quad (16)$$

À partir de (13), on obtient le filtre STAP de la méthode DBU. Cette méthode nécessite l'inversion de la matrice de covariance étendue  $\mathbf{R}_E$  qui est de taille  $2MN \times 2MN$ , ce qui nécessite au minimum  $2MN$  données (cellules distances). Ainsi, cette méthode est assez lourde à utiliser. Dans la section suivante, nous montrons que les algorithmes récursifs sur la distance pour l'estimation d'une base du sous-espace interférence sont une bonne alternative à ces problèmes.

## 4. Algorithmes récursifs sur la distance

Les algorithmes adaptatifs sont utilisés depuis longtemps dans de nombreux domaines du traitement du signal tels que le filtrage, l'analyse spectrale, le traitement d'antenne, la prédiction, etc. et ont de nombreuses applications telles que l'annulation du bruit, l'égalisation de canal, la poursuite de paramètres, le codage de la parole etc. Ils consistent à mettre à jour de façon récursive le vecteur de pondération à l'instant  $k$  à partir de celui à l'instant  $k - 1$  tout en tenant compte de l'observation courante. Ces algorithmes ont une complexité calculatoire faible (complexité linéaire) et sont capables de poursuivre certaines formes de non-stationnarité des données. Nous nous intéressons plus particulièrement aux algorithmes récursifs basés sur les méthodes des sous-espaces. Après avoir étudié leur capacité de poursuite de non stationnarité en traitement d'antenne (source et capteurs mobiles) [8], nous avons adapté ces algorithmes récursifs dans le temps au problème du STAP, c'est-à-dire à une

récurrentité sur la distance. Ils ont déjà montré de bonnes performances dans un cas stationnaire (configuration *sidelooking*) [9]. Notre but est ici de montrer si cette récurrentité sur la distance peut compenser la dépendance en distance due à la configuration du radar.

#### 4.1. Algorithmes à fonction de coût quadratique

Nous présentons tout d'abord quatre algorithmes qui sont basés sur la minimisation de l'erreur quadratique moyenne entre les observations et leur projection sur le sous-espace interférence dont une base  $\mathbf{W}$  est recherchée. En effet, le but est de minimiser la fonction de coût suivante :

$$J(\mathbf{W}(k)) = E \|\mathbf{x}(k) - \mathbf{W}(k)\mathbf{W}(k)^H \mathbf{x}(k)\|^2 \quad (17)$$

où  $\mathbf{x}(k)$  représente le vecteur de données et  $\mathbf{W}$  le sous-espace interférence estimé et  $k$  est la distance d'intérêt. On cherche à minimiser  $J(\mathbf{W})$  par des méthodes itératives.  $J(\mathbf{W})$  ne possède qu'un minimum global atteint en  $\mathbf{W}_{\min} = \mathbf{U}_c \mathbf{Q}$  où  $\mathbf{Q}$  est une matrice unitaire et  $\mathbf{U}_c$  est une matrice dont les colonnes génèrent une base du sous espace interférence. Les algorithmes suivants minimisent cette fonction de coût en utilisant en plus des approximations que nous allons présenter.

##### 4.1.1. Algorithmes PAST et OPAST

L'algorithme PAST (Projection Approximation Subspace Tracking) [10] propose de remplacer l'espérance mathématique par une somme exponentiellement pondérée pour le calcul de la matrice covariance des interférences et du bruit,

$$\begin{aligned} \mathbf{R}_{i+b}(k) &= \sum_{i=1}^k \beta^{k-i} \mathbf{x}_{i+b}(i) \mathbf{x}_{i+b}(i)^H \\ &= \beta \mathbf{R}_{i+b}(k-1) + \mathbf{x}_{i+b}(k) \mathbf{x}_{i+b}(k)^H \end{aligned} \quad (18)$$

où  $0 < \beta \leq 1$  est un facteur d'oubli. La fonction de coût (17) devient en remplaçant l'espérance mathématique par une somme exponentiellement pondérée

$$J(\mathbf{W}(k)) = \sum_{i=1}^k \beta^{k-i} \|\mathbf{x}(i) - \mathbf{W}(k)\mathbf{W}(k)^H \mathbf{x}(i)\|^2 \quad (19)$$

La clé de l'algorithme PAST réside en l'approximation de la projection de  $\mathbf{x}(i)$  dans (19) sur les colonnes de  $\mathbf{W}(k)$  qui n'est pas connue par  $\mathbf{W}^H(i-1)\mathbf{x}(i) \stackrel{\text{def}}{=} \mathbf{y}(i)$  qui, elle, peut être calculée  $\forall 1 < i < k$ , (19) devient

$$J_2(\mathbf{W}(k)) = \sum_{i=1}^k \beta^{k-i} \|\mathbf{x}(i) - \mathbf{W}(k)\mathbf{y}(i)\|^2 \quad (20)$$

$J_2(\mathbf{W}(k))$  est minimisée si

$$\mathbf{W}(k) = \mathbf{R}_{xy}(k) \mathbf{R}_{yy}(k)^{-1} \quad (21)$$

où

$$\begin{aligned} \mathbf{R}_{xy}(k) &= \beta \mathbf{R}_{xy}(k-1) + \mathbf{x}(k) \mathbf{x}(k)^H \\ \text{et } \mathbf{R}_{yy}(k) &= \beta \mathbf{R}_{yy}(k-1) + \mathbf{y}(k) \mathbf{y}(k)^H \end{aligned} \quad (22)$$

Les quantités disponibles à la distance  $k$  sont  $\mathbf{W}(k-1) = \mathbf{R}_{xy}(k-1) \mathbf{R}_{yy}(k-1)^{-1}$ .

En utilisant le lemme d'inversion matricielle  $(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{B} (\mathbf{A}^{-1} + \mathbf{D} \mathbf{A}^{-1} \mathbf{B})^{-1} \mathbf{D} \mathbf{A}^{-1}$  et en posant  $\mathbf{R}_{yy}^{-1}(k) \stackrel{\text{def}}{=} \mathbf{Z}(k)$ , on a

$$\mathbf{Z}(k) = \frac{1}{\beta} \mathbf{Z}(k-1) - \frac{\left(\frac{1}{\beta}\right)^2 \mathbf{Z}(k-1) \mathbf{y}(k) \mathbf{y}(k)^H \mathbf{Z}(k-1)}{1 + \frac{1}{\beta} \mathbf{y}(k)^H \mathbf{Z}(k-1) \mathbf{y}(k)} \quad (23)$$

En posant

$$\mathbf{t}(k) \stackrel{\text{def}}{=} \frac{\left(\frac{1}{\beta}\right) \mathbf{y}(k)^H \mathbf{Z}(k-1)}{1 + \frac{1}{\beta} \mathbf{y}(k)^H \mathbf{Z}(k-1) \mathbf{y}(k)}$$

on obtient la mise à jour suivante

$$\mathbf{Z}(k) = \frac{1}{\beta} \mathbf{Z}(k-1) - \frac{1}{\beta} \mathbf{Z}(k-1) \mathbf{y}(k) \mathbf{t}(k) \quad (24)$$

Pour la mise à jour de  $\mathbf{W}$ , (22) donne, en utilisant (21) et (24)

$$\mathbf{W}(k) = \mathbf{W}(k-1) + (\mathbf{x} - \mathbf{W}(k-1) \mathbf{y}(k)) \mathbf{t}(k) \quad (25)$$

Pour obtenir une base orthonormée, on peut par exemple utiliser l'orthonormalisation de Gram-Schmidt.

L'algorithme OPAST (Orthogonal Projection Approximation Subspace Tracking) [11] est une version orthonormalisée de l'algorithme précédent. En effet, à chaque itération,  $\mathbf{W}$  est orthonormalisée de la façon suivante  $\mathbf{W}(k) = \mathbf{W}(k) (\mathbf{W}(k)^H \mathbf{W}(k))^{-\frac{1}{2}}$ . Une implémentation rapide de l'inverse de cette racine carrée est proposée dans [11], en posant  $\tilde{\mathbf{x}} \stackrel{\text{def}}{=} \|\mathbf{x} - \mathbf{W}(k-1) \mathbf{y}(k)\|^2 \mathbf{t}(k)^H$ ,  $\mathbf{W}(k)^H \mathbf{W}(k) = \mathbf{I} + \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H$ . Alors comme

$$(\mathbf{I} - \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H)^{-\frac{1}{2}} = \mathbf{I} + \frac{1}{\|\tilde{\mathbf{x}}\|^2} \left( \frac{1}{\sqrt{1 + \|\tilde{\mathbf{x}}\|^2}} - 1 \right) \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H$$

l'étape d'orthonormalisation évite l'utilisation de l'orthonormalisation de Gram-Schmidt qui est coûteuse en temps de calcul. Le détail du code des algorithmes est donné dans le tableau 1.

##### 4.1.2. Algorithmes API et FAPI

L'algorithme API (Approximated Power Iteration) [12] réalise une approximation moins restrictive que l'algorithme PAST. En effet c'est une approximation des projections du sous-espace estimé et non plus du sous-espace lui-même qui est réalisée

$$\mathbf{W}(k) \mathbf{W}(k)^H \approx \mathbf{W}(k-1) \mathbf{W}(k-1)^H \quad (26)$$

en utilisant (26) et en posant  $\theta(k) \stackrel{\text{def}}{=} \mathbf{W}(k-1)^H \mathbf{W}(k)$ , on a

$$\mathbf{W}(k) = \mathbf{W}(k-1)\theta(k) \quad (27)$$

d'où  $\mathbf{R}_{xy}(k) = \beta \mathbf{R}_{xy}(k-1)\theta(k-1) + \mathbf{x}(k)\mathbf{y}(k)^H$ . Par le même raisonnement que pour l'algorithme précédent, les mise à jour de  $\mathbf{Z}(k)$ ,  $\mathbf{W}(k)$  et  $\theta(k)$  sont calculées.

L'algorithme FAPI (Fast Approximated Power Iteration) [12] est une version de cet algorithme avec une mise en oeuvre rapide, il est basé sur un choix particulier pour la matrice  $\theta(k)$  qui réduit la complexité. Le détail du code des algorithmes est donné dans le tableau 2.

#### 4.2. Algorithme à fonction de coût non quadratique : algorithme NIC

L'analyse en sous-espaces principaux peut se formuler par la minimisation de l'erreur quadratique moyenne, c'est-à-dire minimiser (17) mais aussi par la maximisation de la variance de  $\mathbf{y}(k)$  avec une contrainte d'orthonormalité sur  $\mathbf{W}$ ,

$$\begin{aligned} J'(\mathbf{W}(k)) &= \frac{1}{2} \text{E}(\mathbf{y}(k)\mathbf{y}(k)^H) \\ &= \frac{1}{2} \text{Tr} \mathbf{W}^H(k) \mathbf{R}_{i+b}(k) \mathbf{W}(k) \end{aligned} \quad (28)$$

sous la contrainte  $\mathbf{W}^H(k)\mathbf{W}(k) = \mathbf{I}$ . L'algorithme NIC (Novel Information Criterion) [13] utilise une variante non quadratique généralisée, la fonction de coût à maximiser est :

$$J_{NIC}(\mathbf{W}) = \frac{1}{2} (\text{tr} \{ \log(\mathbf{W}^H \mathbf{R}_{i+b} \mathbf{W}) \} - \text{tr} \{ \mathbf{W}^H \mathbf{W} \})$$

[13] montre que  $J_{NIC}$  ne présente qu'un maximum global correspondant au sous-espace interférence estimé. La maximisation récursive de la contrainte se réalise de manière similaire à l'algorithme PAST excepté une mise à jour à chaque itération de la matrice  $\mathbf{W}$  avec l'utilisation d'un scalaire  $0 < \eta < 1$  représentant la taille du pas d'apprentissage (cf. [13]). Dans le cas où  $\eta = 1$ , nous retrouvons l'algorithme PAST. Le détail du code de l'algorithme est donné dans le tableau 3.

#### 4.3. Récursivité sur la distance

Ces algorithmes sont récurrents sur la distance dans la mesure où le vecteur de pondération du filtre réjecteur à une cellule distance  $k$  est mis à jour à partir de celui obtenu à la cellule distance précédente ( $k-1$ ) au travers d'un facteur d'oubli  $\beta$  défini dans (22) et compris entre 0 et 1. Plus  $\beta$  est proche de 0, plus le poids accordé à la valeur du vecteur à la distance précédente est faible et au contraire plus  $\beta$  est proche de 1, plus on en tient compte dans la mise à jour du passé. Nous avons vu dans la section 2 que la dépendance en distance n'est pas constante. En effet, pour des distances proches, la dépendance est très forte et

plus les distances deviennent importantes, plus la dépendance s'atténue, et asymptotiquement nous retombons dans un cas stationnaire.

---

#### Algorithm 1 Algorithmes PAST et OPAST

---

**Initialisation:**  $\mathbf{W}(0) \leftarrow \mathbf{I}_{MN \times r}$ ,  $\mathbf{Z}(0) \leftarrow \mathbf{I}_{r \times r}$   
**for**  $k = 1$  to  $Nbr_{snapshot}$  **do**  
   **section principale**  
    $\mathbf{y}(k) = \mathbf{W}(k-1)^H \cdot \mathbf{x}(k)$   
    $\mathbf{h}(k) = \mathbf{Z}(k-1) \cdot \mathbf{y}(k)$   
    $\mathbf{g}(k) = \frac{\mathbf{h}(k)}{\beta + \mathbf{y}(k)^H \cdot \mathbf{h}(k)}$   
    $\mathbf{e}(k) = \mathbf{x}(k) - \mathbf{W}(k-1) \cdot \mathbf{y}(k)$   
   **section secondaire de PAST**  
    $\mathbf{Z}(k) = \frac{1}{\beta} \cdot (\mathbf{Z}(k-1) - \mathbf{g}(k) \cdot \mathbf{y}(k)^H \cdot \mathbf{Z}(k-1))$   
    $\mathbf{W}(k) = \mathbf{W}(k-1) + \mathbf{e}(k) \cdot \mathbf{g}(k)^H$   
   **section secondaire de OPAST**  
    $\gamma(k) = \frac{1}{\beta + \mathbf{y}(k)^H \cdot \mathbf{h}(k)}$   
    $\mathbf{Z}(k) = \frac{1}{\beta} \cdot (\mathbf{Z}(k-1) - \mathbf{g}(k) \cdot \mathbf{y}(k)^H \cdot \mathbf{Z}(k-1))$   
    $\tau(k) = \frac{\beta^2}{\|\mathbf{h}(k)\|^2} \cdot \left( \frac{1}{\sqrt{\beta^2 + \|\mathbf{y}(k) \cdot \mathbf{e}(k)\|^2 \|\mathbf{h}(k)\|^2}} - 1 \right)$   
    $\mathbf{p}'(k) = \frac{1}{\beta} \tau(k) \mathbf{W}(k-1) \mathbf{h}(k)$   
    $\phantom{\mathbf{p}'(k)} + \left( \beta^2 + \tau(k) \|\mathbf{h}(k)\|^2 \right) \gamma(k) \mathbf{e}(k)$   
    $\mathbf{W}(k) = \mathbf{W}(k-1) + \frac{1}{\beta} \mathbf{p}'(k) \cdot \mathbf{h}(k)^H$   
**end for**

---



---

#### Algorithm 2 Algorithmes API et FAPI

---

**Initialisation:**  $\mathbf{W}(0) \leftarrow \mathbf{I}_{MN \times r}$ ,  $\mathbf{Z}(0) \leftarrow \mathbf{I}_{r \times r}$   
**for**  $k = 1$  to  $Nbr_{snapshot}$  **do**  
   **section principale de PAST** (see Algorithm 1)  
   **section secondaire de API**  
    $\Theta(k) = \frac{1}{\sqrt{\mathbf{I}_{N \times N} + \|\mathbf{e}(k)\|^2 \mathbf{g}(k) \mathbf{g}(k)^H}}$   
    $\mathbf{Z}(k) = \frac{1}{\beta} \cdot (\mathbf{I} - \mathbf{g}(k) \cdot \mathbf{y}(k)^H) \mathbf{Z}(k-1) \Theta(k)^{-H}$   
    $\mathbf{W}(k) = (\mathbf{W}(k-1) + \mathbf{e}(k) \mathbf{g}(k)^H) \Theta(k)$   
   **section secondaire de FAPI**  
    $\varepsilon^2(k) = \|\mathbf{x}(k)\|^2 - \|\mathbf{y}(k)\|^2$   
    $\tau(k) = \frac{\varepsilon^2(k)}{1 + \varepsilon^2(k) \|\mathbf{g}(k)\|^2 + \sqrt{1 + \varepsilon^2(k) \|\mathbf{g}(k)\|^2}}$   
    $\eta(k) = 1 - \tau(k) \|\mathbf{g}(k)\|^2$   
    $\mathbf{y}'(k) = \eta(k) \mathbf{y}(k) + \tau(k) \mathbf{g}(k)$   
    $\mathbf{h}'(k) = \mathbf{Z}(k-1)^H \mathbf{y}'(k)$   
    $\varepsilon(k) = \frac{\tau(k)}{\eta(k)} (\mathbf{Z}(k-1) \mathbf{g}(k) - (\mathbf{h}'(k) \mathbf{g}(k)) \mathbf{g}(k))$   
    $\mathbf{Z}(k) = \frac{1}{\beta} (\mathbf{Z}(k-1) - \mathbf{g}(k) \mathbf{h}'(k)^H + \varepsilon(k) \mathbf{g}(k) \mathbf{g}(k)^H)$   
    $\mathbf{e}'(k) = \eta(k) \mathbf{x}(k) - \mathbf{W}(k-1) \mathbf{y}'(k)$   
    $\mathbf{W}(k) = \mathbf{W}(k-1) + \mathbf{e}(k) \cdot \mathbf{g}(k)^H$   
**end for**

---

---

**Algorithm 3** Algorithme NIC
 

---

**Initialisation :**  $\mathbf{W}(0) \leftarrow \mathbf{I}_{MN \times r}$ ,  $\delta$  petit entier positif,  
 $\mathbf{Z}(0) \leftarrow \delta \cdot \mathbf{I}_{r \times r}$ ,  $\tilde{\mathbf{W}}(0) \leftarrow$  nul matrix  
**for**  $k = 1$  to Nbr snapsho **do**  
 $\mathbf{y}(k) = \mathbf{W}(k-1)^H \cdot \mathbf{x}(k)$   
 $\mathbf{h}(k) = \mathbf{Z}(k-1) \cdot \mathbf{y}(k)$   
 $\mathbf{g}(k) = \frac{\mathbf{h}(k)}{\beta + \mathbf{y}^H(k) \cdot \mathbf{h}(k)}$   
 $\mathbf{Z}(k) = \frac{1}{\beta} \cdot (\mathbf{Z}(k-1) - \mathbf{g}(k) \cdot \mathbf{y}(k)^H \cdot \mathbf{Z}(k-1))$   
 $\mathbf{e}(k) = \mathbf{x}(k) - \tilde{\mathbf{W}}(k-1)^H \cdot \mathbf{y}(k)$   
 $\tilde{\mathbf{W}}(k) = \tilde{\mathbf{W}}(k-1) + \mathbf{e}(k) \cdot \mathbf{g}(k)^H$   
 $\mathbf{W}(k) = (1 - \eta) \cdot \mathbf{W}(k-1) + \eta \cdot \tilde{\mathbf{W}}(k)$   
**end for**

---

## 5. Résultats obtenus

Les performances des algorithmes sont représentées par la perte en SINR qui correspond au rapport entre le rapport signal à interférence plus bruit à la sortie du filtre réjecteur et le rapport signal à bruit dans un environnement sans interférence

$$SINR_{loss} = \frac{\sigma^2 \cdot |\mathbf{w}^H \mathbf{v}_t|^2}{NM \cdot \mathbf{w}^H \mathbf{R}_{i+b} \mathbf{w}}$$

où  $\mathbf{w}$  le vecteur de pondération calculé avec les algorithmes précédents de la manière suivante

$$\mathbf{w} = \mathbf{I} - \mathbf{W}\mathbf{W}^H$$

où  $\mathbf{W}$  est le sous espace interférence estimé avec les algorithmes récursifs sur la distance (cf. Tables 1 à 3). La perte en SINR minimale, c'est-à-dire en connaissant la matrice  $\mathbf{R}_{i+b}$ , s'exprime par

$$SINR_{loss_{opt}} = \frac{\sigma^2 \cdot \mathbf{v}_t^H \mathbf{R}_{i+b}^{-1} \mathbf{v}_t}{NM}$$

L'altitude du porteur est de 9 km. La vitesse de la plate forme est constante et vaut  $100 \text{ m.s}^{-1}$ . L'antenne radar est composée de 12 capteurs ( $N = 12$ ) et de 10 impulsions ( $M = 10$ ).

### 5.1. Cas d'une antenne linéaire uniforme

Un radar bande X est utilisé. Dans les simulations correspondant aux figures 7(a) à 7(c), les différents algorithmes sont comparés à l'optimal et au SMI pour différentes valeurs d'angle de crabe et une distance de 9000 m. Les figures 7(a) à 7(d) montrent les résultats obtenus au cours des différentes simulations. La taille du support est de 480 cellules distance secondaires (ce qui correspond  $4NM$  où  $N$  est le nombre de capteurs (ici égal à 12) et  $M$  le nombre d'impulsions (ici égal à 10)). Cette taille de support correspond à une convergence acceptable des algorithmes. De

plus, la résolution utilisée étant ici de l'ordre du mètre, ce support correspond à la couverture d'une plage d'environ 500 m, ce qui est tout à fait raisonnable. Nous remarquons que les algorithmes récursifs sur la distance atteignent de meilleures performances que le SMI. Ils se comportent d'ailleurs globalement de la même façon. Cependant, les algorithmes API et FAPI convergent mieux dans toutes les situations (quelles que soient l'altitude ou la valeur de l'angle de crabe). Nous proposons aussi un facteur d'oubli variable capable de s'adapter à la non stationnarité du problème. Au début de la simulation, lorsque les distances traitées sont faibles, nous choisissons une faible valeur pour  $\beta$  qui augmente ensuite au cours de la simulation pour être très proche de 1 en fin de simulation. Nous avons testé deux types de variations : une variation linéaire et une variation exponentielle. La figure 7(d) présente l'amélioration de performance obtenue avec un facteur d'oubli variable au cours de la simulation. Nous n'avons représenté que les résultats obtenus avec l'algorithme FAPI par souci de lisibilité mais tous les algorithmes présentés dans la section 4 montrent le même comportement. Les deux types de variation compensent mieux la non stationnarité qu'un pas constant au cours de la simulation.

### 5.2. Cas d'une antenne courbée et circulaire uniforme

Dans les simulations correspondant aux figures 8(a) et 8(b), l'antenne radar utilisée est une antenne circulaire (l'angle d'ouverture est de  $180^\circ$ ) et les figures 9(a) et 10(b) correspondent à l'utilisation d'une antenne courbée (angle d'ouverture de  $10^\circ$ ), pour les figures 9(a) et 9(b),  $2NM$  cellules distances ont été utilisées et pour les figures 10(a) et 10(b),  $4NM$  cellules distances ont été utilisées. La fréquence porteuse du radar vaut 450 MHz. Nous n'avons tracé que les performances de l'algorithme FAPI en ce qui concerne les algorithmes récursifs sur la distance toujours par souci de lisibilité. Il est comparé à l'optimal et aux algorithmes SMI et DBU. Les figures 8(a), 8(b) représentent la perte en SINR à la fréquence spatiale  $0^\circ$  en configuration latérale et frontale. L'antenne utilisée est une antenne circulaire. La portée est de 10km. Nous sommes donc dans un cas où les données sont très non stationnaires. Nous remarquons que les algorithmes SMI et DBU présentent des performances nettement moins bonnes que l'algorithme FAPI. Les figures 9(a), 9(b) représentent la perte en SINR à la fréquence spatiale  $0^\circ$  en configuration latérale et frontale pour une antenne légèrement courbée (angle d'ouverture =  $10^\circ$ ). En configuration latérale, la portée est de 25 km, ce qui rend le contexte « peu non stationnaire » tandis qu'en configuration frontale, la portée est de 10 km, ce qui est un cas plus non stationnaire. Nous remarquons que dans tous les cas l'algorithme FAPI surpasse les algorithmes SMI et DBU. Dans le cas faiblement non stationnaire, l'algorithme SMI obtient cependant de bonnes performances. L'algorithme DBU présente jusqu'ici des performances médiocres car notre support de données secondaires était de  $2NM$ , si nous utilisons le double de données secondaires ( $4NM$ ) et les mêmes conditions de simulation par

ailleurs, nous obtenons les résultats présentés figures 10(a) et 10(b). Nous remarquons que dans le cas le plus non stationnaire, les performances de DBU sont légèrement meilleures que celles de FAPI tandis que dans le cas le moins non stationnaire, FAPI le surpasse.

## 6. Conclusion

Dans ce papier, nous montrons comment une non stationnarité des données peut être compensée par la récursivité en distance

des algorithmes adaptatifs utilisés sans utilisation complémentaire de méthodes de compensation dans le cadre de radar à antenne linéaire uniforme ou courbée/circulaire uniforme. Ainsi, sans connaître de paramètres *a priori* sur le radar et en restant dans la complexité calculatoire permettant un traitement en temps réel des données, nous pouvons, grâce à la récursivité sur la distance, pallier le problème de configuration non latérale. Nous avons vu que les performances de ces algorithmes peuvent encore être améliorées par l'utilisation d'un facteur d'oubli variable. Une étude plus théorique de la variation des sous-espaces en fonction de cette non-stationnarité est à l'étude.

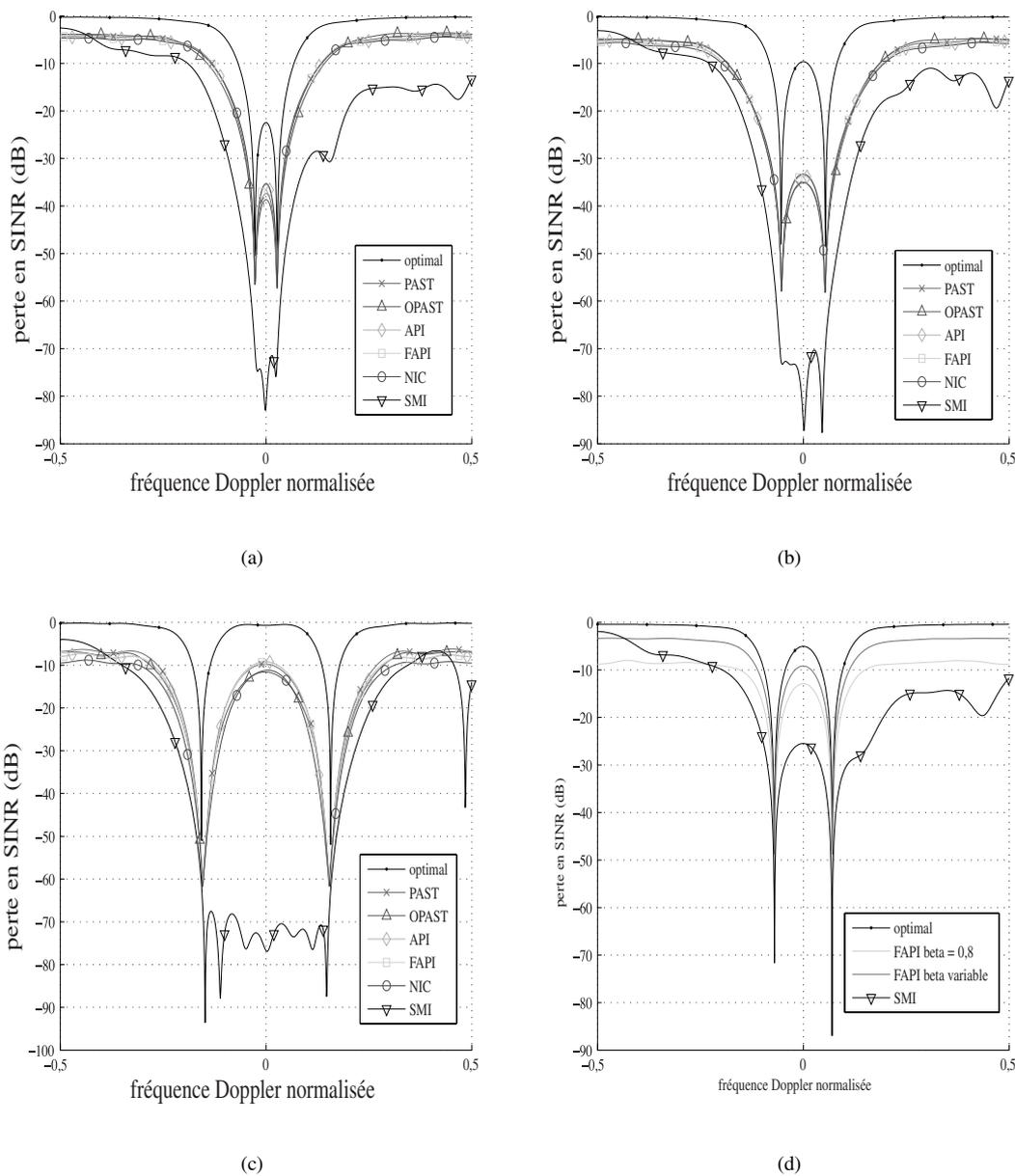


Figure 7. Perte en SINR à la fréquence spatiale  $0^\circ$  (sur 20 simulations Monte Carlo). Avec un angle de crabe de  $10^\circ$ ,  $R_c = 9000$  m (a), avec un angle de crabe de  $20^\circ$ ,  $R_c = 9000$  m (b), avec un angle de crabe de  $90^\circ$ ,  $R_c = 9000$  m (c), avec un angle de crabe de  $10^\circ$ ,  $R_c = 15000$  m (d).

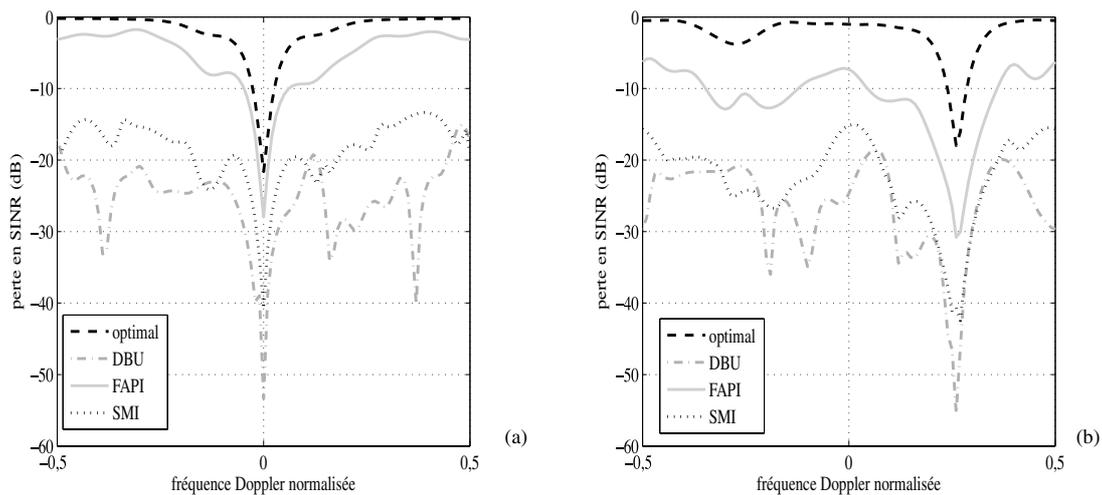


Figure 8. Perte en SINR à la fréquence spatiale  $0^\circ$  avec une antenne circulaire (angle d'ouverture =  $180^\circ$ ), en configuration latérale (a), en configuration non latérale (angle de crabe =  $90^\circ$ ) (b).

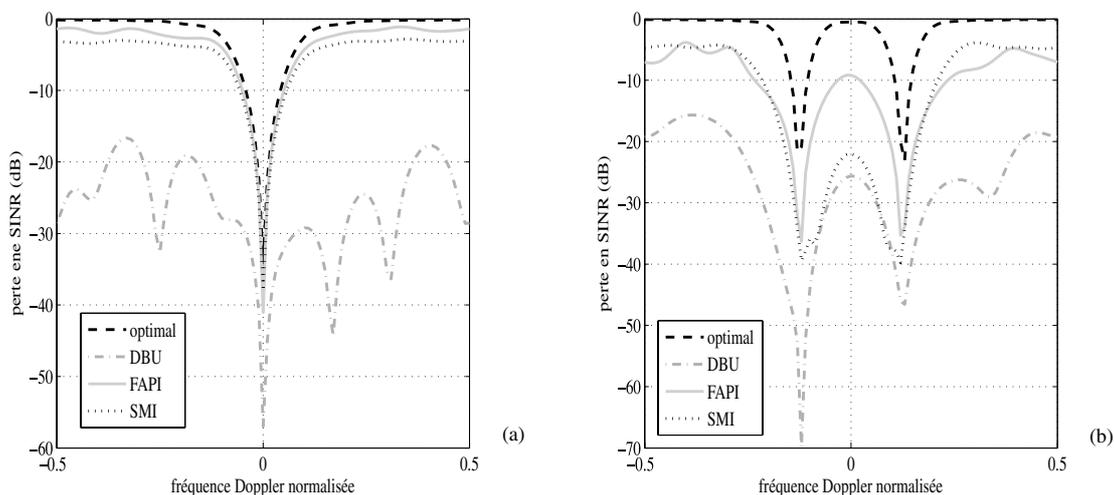


Figure 9. Perte en SINR à la fréquence spatiale  $0^\circ$  avec une antenne courbée (angle d'ouverture =  $10^\circ$ ), en configuration latérale (a), en configuration non latérale (angle de crabe =  $90^\circ$ ) (b); avec 2MN cellules distance.

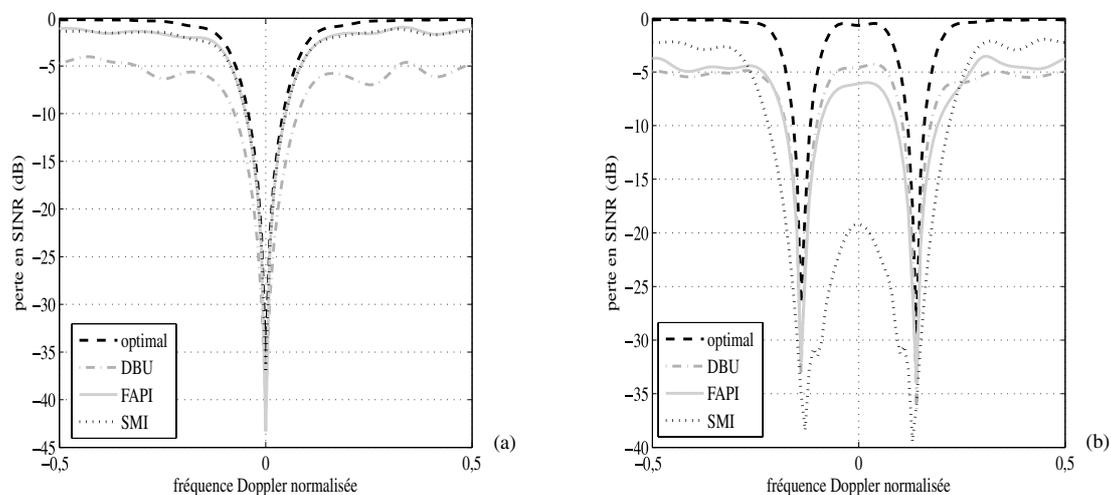


Figure 10. Perte en SINR à la fréquence spatiale  $0^\circ$  avec une antenne courbée (angle d'ouverture =  $10^\circ$ ), en configuration latérale (a), en configuration non latérale (angle de crabe =  $90^\circ$ ) (b); avec 4MN cellules distance.

# Références

- [1] J. WARD, "Space-Time Adaptive Processing for airborne radar," *Technical Report 1015*, Lincoln Laboratory MIT, Dec. 1994.
- [2] M. BÜHREN, M. PESAVENTO and J.F. BÖHME, "Virtual Array Design for Array Interpolation using Differential Geometry," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 229-232, May 2004.
- [3] A. HAIMOVICH, "The eigencanceler: adaptive radar by eigenanalysis methods," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, pp. 532-542, Apr 1996.
- [4] G. BORSARI, "Mitigating effects on STAP processing caused by an inclined array", in *Proceedings of 1998 IEEE Radar Conference*, Dallas, TX, May 1998, pp. 135-140.
- [5] B. HIMED, Y. ZHANG and A. HAJJARI, "STAP with angle Doppler compensation for bistatic airborne radars", in *Proceedings of IEEE Radar Conference*, 2002, pp. 311-317.
- [6] F. LAPIERRE, "Registration-based Range-dependence compensation in Airborne Bistatic Radar STAP", *PhD Thesis*, Liège, 2005.
- [7] M. ZATMAN, "Performance Analysis of the Derivative Based Updating Method," ASAP workshop 2001.
- [8] S. BEAU and S. MARCOS, "Performances et capacité de poursuite d'algorithmes adaptatifs à complexité linéaire basés sur des méthodes de sous-espaces," *Internal Report*, CNRS, Supelec, Nov. 2006.
- [9] H. BELKACEMI and S. MARCOS, "Fast iterative subspace algorithms for airborne STAP radar", in *Journal of Signal Processing Special Issue on Multisensor Processing for Signal extraction and applications*, 2006, article ID 37296.
- [10] B. YANG, "Projection Approximation Subspace Tracking," *IEEE Transactions on Signal Processing*, vol. 43, pp. 95-107, Jan. 1995.
- [11] K. ABED-MERAIM, A. CHKEIF and Y. HUA, "Fast Orthonormal PAST Algorithm," *IEEE Signal Processing Letters*, vol. 7, pp. 60-62, Mar. 2000.
- [12] R. BADEAU, B. DAVID, and G. RICHARD, "Fast approximated power iteration subspace tracking," *IEEE Transactions on Signal Processing*, vol. 53, pp. 2931-2941, Aug. 2005.
- [13] Y. MIAO and Y. HUA, "Fast Subspace Tracking and Neural Network Learning by a Novel Information Criterion," *IEEE Transactions on Signal Processing*, vol. 46, pp. 1967-1979, Jul. 1998.



Sophie **Beau**

Sophie Beau est ingénieur de l'école supérieure de mécanique et d'électricité (ESME Sudria) depuis 2005. Elle réalise son master de recherche en automatique et traitement du signal, en 2005, à l'université d'Orsay (paris XI). Elle termine actuellement son doctorat en traitement du signal au Laboratoire des signaux et systèmes (LSS) à Supelec (Gif sur Yvette) sous la direction du Dr. Sylvie Marcos. Ses principaux domaines d'intérêt sont le traitement d'antenne, le traitement spatio-temporel du signal appliqué notamment aux radars aéroportés.

Sylvie **Marcos**

Sylvie Marcos est ingénieur de l'école Centrale de Paris (1984). Elle obtient son doctorat (1987) ainsi que son habilitation à diriger la recherche (1995) à l'université d'Orsay (paris XI). Elle est directeur de recherche au CNRS et travaille au laboratoire des signaux et systèmes (LSS) à Supelec (Gif sur Yvette). Ses domaines de recherche sont le filtrage adaptatif linéaire et non-linéaire, le traitement spatial et spatio-temporel du signal avec des applications aux radio-communications, à l'acoustique sous-marine et au radar, l'égalisation de canaux de transmission et la détection multi-utilisateurs en CDMA avec recherche de séquences d'étalement.



