



HAL
open science

Filtrage adapté stochastique : poursuite de sous-espace et filtrage multicritère. Application à l'acoustique sous-marine

Maissa Chagmani

► **To cite this version:**

Maissa Chagmani. Filtrage adapté stochastique : poursuite de sous-espace et filtrage multicritère. Application à l'acoustique sous-marine. Acoustique [physics.class-ph]. Université de Toulon, 2018. Français. NNT : 2018TOUL0018 . tel-04904708

HAL Id: tel-04904708

<https://theses.hal.science/tel-04904708v1>

Submitted on 21 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Ecole doctorale : Mer et Sciences (ED 548)

IM2NP, équipe Signal et Tracking

THÈSE présentée par :

Maissa CHAGMANI

soutenue le : **05 Décembre 2018**

pour obtenir le grade de Docteur en Automatique, Signal, Productique, Robotique

Spécialité : Traitement du signal

**Filtre adapté stochastique : poursuite de
sous-espace et filtrage multicritère.
Application à l'acoustique sous-marine**

THÈSE dirigée par :

M. Claude JAUFFRET

Professeur des Universités, Université de Toulon,
Directeur de Thèse.

JURY :

Mme. Meriem JAIDANE,

Professeur à l'ENIT, Tunis, Président du jury.

Mme. Sylvie MARCOS,

Directeur de recherche au CNRS, Université Paris
Saclay, Rapporteur.

M. Jean-Yves TOURNERET,

Professeur des Universités, INP-ENSEEIH,
Toulouse, Rapporteur.

M. Bruno BORLOZ,

Maître de Conférences, Université de Toulon, Co-
encadrant de Thèse.

M. Bernard XERRI,

Maître de Conférences, Université de Toulon, Co-
encadrant de Thèse

Remerciements

Avant de présenter ce travail, il convient d'exprimer ma vive reconnaissance envers les personnes qui m'ont apporté leur aide inestimable lors de la mise en œuvre de ce projet de thèse réalisé au sein de l'équipe « signal et tracking » aux laboratoires IM2NP à l'université de Toulon.

Mes vifs remerciements s'adressent à Monsieur Claude JAUFFRET, professeur à l'Université de Toulon et directeur de thèse, ainsi qu'à Messieurs Bernard XERRI et Bruno BORLOZ, co-encadrants de thèse, avec qui j'ai pris beaucoup de plaisir à travailler dans une ambiance très amicale.

Je remercie vivement Monsieur Jean-Yves TOURNERET, professeur à l'INP-ENSEEIH Toulouse, et Madame Sylvie MARCOS, directrice de recherche au CNRS, qui m'ont fait l'honneur d'être les rapporteurs de cette thèse.

Je remercie également Madame Meriem JAIDANE, professeure à l'ENIT de Tunis, d'avoir accepté de faire partie du jury et d'en être la présidente.

Ainsi, je congratulate toute l'équipe « signal et tracking » de l'IM2NP, de façon générale, et j'exprime mon profond respect à Annie-Claude PEREZ, Claude JAUFFRET, Bernard XERRI, Bruno BORLOZ, Jean BARRERE et Gilles CHABRIEL pour avoir facilité mon intégration dans un environnement dynamique, chaleureux et sympathique. Merci pour votre soutien et pour tous les moments agréables que j'ai passé à vos côtés.

Enfin, j'adresse mes remerciements à tous mes enseignants pour le temps et l'effort qu'ils m'ont consacré durant mes études.

Table des matières

Remerciements	2
Table des matières	3
Table de figures	6
Introduction générale	11
Chapitre I :	17
Le filtre adapté stochastique	17
I.1 Introduction	18
I.2 Le filtre adapté stochastique appliqué à la détection	18
I.2.1 Le filtre adapté (MF).....	18
I.2.2 Le filtre adapté stochastique (SMF).....	19
I.2.3 Le filtre adapté stochastique sous contrainte (CSMF)	20
I.3 Le CSMF appliqué à la classification.....	27
I.4 Conclusion.....	29
Chapitre II :	31
Le filtre multicritère	31
II.1 Introduction.....	32
II.2 Filtrage multicritère appliqué à des signaux simulés.....	33
II.2.1 Signaux utilisés et estimation des matrices	33
II.2.2 Résultats de classification	34
II.2.3 Etude des courbes COR.....	35
II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine	39
II.3.1 Nature des signaux	39

II.3.2 Estimation des matrices de covariance	43
II.3.3 Détection des signaux.....	43
II.3.4 Classification de signaux d'acoustique sous-marine	46
II.4 Conclusion	54
Chapitre III :	56
Poursuite du sous-espace optimal.....	56
III .1 Introduction.....	57
III .2 Le Tracking : poursuite de sous-espace.....	59
III.2.1 Principe.....	59
III.2.2 Nécessité de mettre en place des algorithmes	61
III.2.3 Algorithmes de poursuite du sous-espace principal	61
a) Algorithmes de tracking de la famille PAST	62
b) Algorithmes de tracking basés sur le principe de puissance itérée	64
c) Algorithmes de tracking basés sur la méthode de calcul itératif SVD.....	65
d) Algorithmes de tracking de type SP	65
e) Comparaison entre certains algorithmes de tracking	66
III .3 Les algorithmes de type SP.....	70
III .4 L'algorithme CSMFST.....	74
III.4.1 Présentation de l'algorithme	75
III.4.2 Convergence de l'algorithme	78
a) Convergence du rapport signal à bruit $\rho(t, t)$	78
b) Convergence du sous espace engendré par $W(t)$	80
III .5 Conclusion.....	84
Chapitre IV :	86
Applications : simulations et signaux réels.....	86
IV.1 Introduction	87
IV.2 Conditions de simulations.....	87
IV.3 Performances du CSMFST dans le cas stationnaire	88
IV.3.1 Premier scénario : cas stationnaire, bruit blanc	88
IV.3.2 Deuxième scénario : cas stationnaire, bruit non blanc	90

IV.4 Performances du CSMFST dans le cas non stationnaire	92
IV.4.1 Troisième scénario : cas non stationnaire, bruit blanc	92
IV.4.2 Quatrième scénario : cas non stationnaire, bruit non blanc	94
IV.4.3 Cinquième scénario : cas non stationnaire, bruit non blanc	95
IV.5 Version modifiée du CSMFST : modélisation additive (signal plus bruit)	96
IV.6 Application aux signaux bioacoustiques réels :	100
IV.7 Application au bruit de mer	103
IV.8 Conclusion	105
Conclusion générale	107
Annexe 1 :	110
Calculs des dérivées des critères	110
Annexe 2 :	113
Réalisation du CSMF	113
Annexe 3 :	116
Calcul des vecteurs propres de $M(t)$	116
Annexe 4 :	118
Matrice de covariance et sous-espace principal	118
Annexe 5 :	121
Applications numériques	121
Bibliographie	124

Table de figures

Figure II.1. Signal test à filtrer	33
Figure II.2. Filtrage associé au critère arithmétique	34
Figure II.3. Filtrage multicritère	35
Figure II.4. Courbes COR associées aux différents critères (signal s4).....	36
Figure II.5. Courbes COR associées aux différents critères (signal s3).....	37
Figure II.6. Courbes COR associées aux différents critères (signal s2).....	37
Figure II.7. Courbes COR associées aux différents critères (signal s1).....	38
Figure II.8. détection du signal s1 : résultats des courbes COR pour différents RSB.....	39
Figure II.9. clic de dauphin.....	41
Figure II.10. clic de cachalot (face)	41
Figure II.11. clics de cachalot (dos)	42
Figure II.12. Bruit de mer	42
Figure II.13. Détection d'un dauphin dans un clic de dauphin à l'aide du CSMF	44
Figure II.14. Détection d'un dauphin dans un clic de cachalot à l'aide du CSMF	44
Figure II.15. Détection d'un cachalot vu de face dans un clic de cachalot vu de face (CSMF)	45
Figure II.16. Détection d'un cachalot vu de face dans un clic de cachalot vu de dos (CSMF)	46
Figure II.17. signal test construit à partir de différents échantillons.....	47
Figure II.18. résultat du filtrage du signal test en considérant le critère harmonique ($p = 5$).....	48
Figure II.19. résultat du filtrage du signal test en considérant le filtre multicritère ($p = 5$).....	48
Figure II.20. courbe COR : filtrage du signal test en considérant le filtre multicritère	49
Figure II.21. courbe COR : résultat du filtrage par CSMF en mode détection.....	50
Figure II.22. résultat du filtrage du signal test en considérant le filtre géométrique ($p = 2$)	51
Figure II.23. courbe COR : résultat du filtrage en considérant le filtre multicritère ($p = 2$)	51
Figure II.24. courbe COR : résultat du filtre multicritère pour différentes valeur de p	52
Figure II.25. Détection de cachalots en considérant le filtre multicritère ($p = 2$)	53
Figure II.26. courbe COR : résultat du filtrage en considérant le filtre multicritère ($p = 2$)	53
Figure II.27. courbe COR : résultat du filtre multicritère pour différentes valeur de p	54
Figure III.1. Comparaison des algorithmes de tracking en termes de convergence (a) et d'orthogonalité (b) dans le cas stationnaire [67]	67
Figure III.2. trajectoire des sources [67]	68
Figure III.3. Comparaison des algorithmes de tracking en termes de convergence (a) et d'orthogonalité (b) dans le cas non stationnaire [67]	69

Figure IV.1. Performance de l'algorithme CSMFST dans le cas stationnaire : $d(Ept, Ep^*(t))$ (a), $d(Ept, \xi_p)$ (b) et RSB (c) [Annexe 5 scénario 1].	89
Figure IV.2. Comparaison dans le cas stationnaire entre un algorithme de type SP (NFYAST) et le CSMFST en présence d'un bruit blanc [Annexe 5 scénario 1].	90
Figure IV.3. Performance de l'algorithme CSMFST dans le cas stationnaire : $d(Ept, Ep^*(t))$ (a), $d(Ept, \xi_p)$ (b) et RSB (c) [Annexe 5 scénario 2].	91
Figure IV.4. Performance du CSMFST dans le cas non stationnaire en termes de $d(Ept, Ep^*(t))$ (a) et du RSB (b) [Annexe 5 scénario 3].	93
Figure IV.5 Distance $d(Ept, Ep^*(t))$ pour différentes valeurs de β .	94
Figure IV.6. Performance de l'algorithme CSMFST dans le cas non stationnaire en termes de distance $d(Ept, Ep^*(t))$ (a) et du RSB (b) [Annexe 5 scénario 4].	95
Figure IV.7. Performance de l'algorithme CSMFST dans le cas non stationnaire en termes de distance $d(Ept, Ep^*(t))$ (a) et RSB (b) [Annexe 5 scénario 5].	96
Figure IV.8. Performance du CSMFST en termes de distance $d(Ept, Ep^*(t))$ (a) et de RSB (b) dans le cas d'une matrice de bruit fixée [Annexe 5 scénario 6].	98
Figure IV.9. Performance du CSMFST en termes de distance $d(Ept, Ep^*(t))$ (a) et de RSB (b) dans le cas d'une matrice de bruit fixée [Annexe 5 scénario 7].	99
Figure IV.10. Signaux bioacoustiques	100
Figure IV.11. Performances du CSMFST en non stationnaire : $d(Ept, Ep^*(t))$ (a) et RSB (b).	101
Figure IV.12. Signal test.	102
Figure IV.13. Résultat du filtrage multicritère du signal test : détection d'orques.	102
Figure IV.14. Distance $d(Ept, Ep^*(t))$.	103
Figure IV.15. ρ_t et $\rho^*(t)$.	104
Figure IV.16. La densité spectrale de puissance des signaux avant filtrage.	104
Figure IV.17. La densité spectrale de puissance des signaux après filtrage.	105
Figure 1 : Signaux de test.	113

Liste des acronymes

C

COR	Caractéristiques O pérationnelles de R éception
CSMFST	Constrained S tochastic M atched F ilter S ubspace T racking

E

ESMF	Extended S tochastic M atched F ilter
EVD	Eigen V alue D ecomposition

F

FAP	Fast A pproximated P ower I teration
FASC	Filtre A dapté S tochastique sous C ontrainte
FASE	Filtre A dapté S tochastique E tendu
FCPAST	Fast C onstrained P rojection A pproximation S ubspace T racking
FDPM	Fast D ata P rojection M ethod for Stable Subspace Tracking
FIR	Finite I mpulse R esponse filter

G

GYAST	Generalized Y AST
-------	--------------------------

H

HPAST	Householder-based P AST
-------	--------------------------------

I

IHM	Interface H omme M achine
-----	---

L

LORAF	L OW- R ANK Adaptive F ilters
-------	--

N

NDPMB	A Novel Fast and Stable Data Projection Method algorithm
NFYAST	Not Fast Yet Another Subspace Tracker
NIC	the Novel Information Criterion
NPAST	Normalized PAST algorithm

O

OPAST	Orthonormal PAST algorithm
OSD	Orthogonal-Square Decomposition

P

PAST	Projection Approximation Subspace Tracking,
------	--

R

RSB	Rapport Signal à Bruit
-----	-------------------------------

S

SGYAST	Stable GYAST
SNR	Signal To Noise Ratio
SP	Subspace Projection algorithms
SVD	Single Values Decomposition

Y

YAST	Yet Another Subspace Tracker
------	-------------------------------------

Notations

Ensembles

\mathbb{N}, \mathbb{Z}	ensembles des entiers, respectivement relatifs et naturels
\mathbb{R}, \mathbb{C}	ensembles des nombres réels et des nombres complexes

Variables

a	scalaire (caractère maigre minuscule)
\mathbf{v}	vecteur (caractère gras minuscule)
$\mathbf{0}$	vecteur nul
v_i	composante du vecteur \mathbf{v} d'indice i
\mathbf{A}	matrice (caractère gras majuscule)
a_{ij}	coefficient de la matrice \mathbf{A} d'indices (i, j)

Matrices

$M_{n \times m}$	ensemble des matrices de dimension $n \times m$
\mathbb{I}_n	matrice identité de dimensions $n \times n$
$\mathbb{I}_{n \times p}$	matrice identité de dimensions $n \times p$
\mathbb{O}_n	matrice nulle de dimensions $n \times n$
$\mathbb{O}_{n \times p}$	matrice nulle de dimensions $n \times p$
$(.)^T$	transposé
$(.)^H$	conjugué hermitien
$\text{tr}(\cdot)$	trace d'une matrice

Probabilités

$\mathbb{E}(\cdot)$	espérance mathématique
---------------------	------------------------

Divers

\equiv	égal par définition
δ_{ij}	symbole de Kronecker
$O(\cdot)$	grandeur dominée par la fonction en argument
$\text{Span}\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$	espace engendré par les vecteurs $\mathbf{v}_1, \dots, \mathbf{v}_n$
\mathcal{E}_p^*	sous-espace optimal de dimension p

On note $\mathbf{A} = [a_{ij}]_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}}$ ou $[a_{ij}]$ la matrice de $M_{n \times m}$ dont les éléments sont les a_{ij} .

De même, on note $\mathbf{\Lambda} = [\lambda_i]_{1 \leq i \leq n} = [\lambda_i]$ la matrice diagonale de $M_{n \times n}$ dont les éléments sont les λ_i .

Introduction générale

Contexte général :

La détection d'un signal d'intérêt noyé dans un bruit de fond est un vieux problème et de nombreuses méthodes de détection ont été proposées, qui dépendent de la nature du signal à détecter : déterministe, stochastique, ... [1][2][3][4]. Une revue historique des méthodes de détection est présentée dans [3].

Pour ce problème, les considérations théoriques ont montré que la quantité essentielle à calculer est le rapport de vraisemblance, quantité comparée à un seuil dont la valeur est déterminée par le choix d'un critère. Ce test ne peut être utilisé que dans le cas où les densités de probabilité des signaux sont connues. Dans le cas contraire, la maximisation du rapport signal à bruit (RSB) après une opération de filtrage ou de projection dans un sous-espace adapté est un outil commode de détection [5].

Dans le cadre de notre étude, le signal utile, tout comme le bruit, sera un vecteur aléatoire de dimension n de propriétés statistiques connues à l'ordre deux : ceci justifie la méthode proposée et le choix d'étude du filtrage adapté.

Le Filtre Adapté (Matched Filter, FA ou MF), qui détecte un signal d'intérêt déterministe supposé parfaitement connu, noyé dans un bruit aléatoire dont les statistiques sont connues au second ordre, maximise le RSB en sortie du filtre *i.e.* après projection dans un sous-espace optimal de dimension un [2].

Son concept a été proposé dès la période de la seconde guerre mondiale [6] et a été introduit en 1943 dans un radar pulsé [7].

Cette technique puissante de détection n'a été appliquée dans le traitement de signal qu'à partir des années 1950 pour rendre maximum le RSB. En effet, c'est en cette année que les chercheurs du « Massachusetts Institute of Technology » MIT ont utilisé un filtre basé sur la fonction de corrélation : le signal reçu est comparé à un signal de référence constitué par l'impulsion émise. Ainsi la distance entre les deux signaux est minimale quand la fonction de corrélation entre ces mêmes signaux est maximale. La première implémentation de ce dispositif a été réalisée en 1953. [7]

Dans les systèmes RADAR et SONAR, le MF est toujours la méthode de détection de signal la plus utilisée (tenant compte de l'effet Doppler). Le MF est utilisé aussi dans une large gamme d'applications réelles telles que la télémétrie de cible et l'estimation de canal [8].

Le problème qui a été rencontré par la suite, est qu'il existe souvent une incertitude sur les caractéristiques du signal et/ou du bruit : en effet, la non-linéarité du canal, l'instabilité de

synchronisation, et d'autres effets ont tendance à distordre le signal et à le rendre imprévisible ou difficile à modéliser.

De plus, en raison des non-stationnarités, d'environnements changeants ou d'autres incertitudes liées à la modélisation, il arrive souvent que la caractérisation du bruit ne soit pas possible. Ceci a conduit certains auteurs à chercher d'autres techniques dites robustes qui imposent certaines contraintes sur la variabilité des réalisations autour d'un signal déterministe [9] [10].

En se basant sur ces constatations, une extension du filtre adapté, qui suppose que le signal utile est non plus déterministe mais aléatoire a vu le jour : c'est le Filtre Adapté Stochastique (Stochastic Matched Filter, FAS ou SMF) développé en 1991 par Cavassilas [11] et présenté par Cavassilas et Xerri en 1993 [12].

Le critère que l'on maximise est le RSB ou de manière équivalente le gain sur le RSB (ratio entre le RSB en sortie et en entrée du filtre).

Initialement proposé pour trouver un sous-espace optimal de dimension un dans lequel le RSB est maximal, il conduit naturellement à un ensemble de solutions qui peuvent être combinées pour engendrer un sous-espace de dimension $p > 1$ dans lequel le compromis entre la probabilité de détection et de fausse alarme est meilleur.

L'ensemble de cette base s'appelle Extended SMF ou FAS Etendu (FASE ou ESMF) [5] mais a continué de s'appeler SMF dans la littérature : elle n'engendre pas le sous-espace qui maximise le RSB (sous-espace optimal) et la filtrée du signal ainsi obtenu peut s'écarter fortement du signal émis. Cet inconvénient peut devenir majeur particulièrement dans les problèmes de filtrage, de compression, de reconstruction etc.... On est alors amené à chercher le sous-espace dans lequel le RSB est maximal.

Le SMF a été employé pour la détection de signaux et de séquences audios [13], dans le domaine du traitement d'images pour la détection des contours et des textures dans une image bruitée (SAR, SAS) [14] [15] [16], dans les systèmes RADAR et SONAR [17] [18] [19], dans la détection et la localisation des cétacés par étude des signaux bioacoustiques [20] [21] [22] [23] [24] et pour la détection d'engins sous-marins ou de plongeurs [25] [26] [27].

Dans ce contexte précis, une étude a été menée au sein de notre Laboratoire « IM2NP » par Borloz [5] sous la direction de Xerri dans le cadre d'un projet de thèse en 2005 et a consisté à résoudre ce problème de maximisation du RSB après filtrage (ou projection) en le décrivant dans un espace de dimension inférieure fixée *a priori*. On est alors conduit à résoudre une équation aux valeurs propres dont la matrice inconnue est paramétrée par le RSB. Ces travaux, ont conduit à une extension du SMF, le FASC (FAS sous contrainte) ou CSMF (Constrained SMF), permettant de donner à la fois le sous-espace optimal et le RSB dans ce sous-espace [28] [29] [30].

La méthode de filtrage sous contrainte proposée limite les hypothèses à la connaissance des seules propriétés statistiques d'ordre deux des processus aléatoires en jeu, par le biais de leurs matrices de covariance.

La même étude a permis d'étendre l'utilisation du CSMF au cas où le signal aléatoire utile est noyé dans un ou éventuellement plusieurs bruits perturbateurs superposés de manière additive : on parle alors de détection multi bruits ou abusivement de classification [5] [31].

Dans cet environnement multi-bruits, on peut définir un gain sur le RSB pour chacun des bruits.

Il y a toutefois nécessité de maximiser un critère unique qui est une combinaison de ces gains élémentaires. L'idée retenue était de prendre des moyennes des gains (éventuellement pondérés par la fréquence d'occurrence des signaux en jeu) ; on dispose alors de trois méthodes différentes liées au type de moyenne : arithmétique, géométrique et harmonique.

Le CSMF est cité dans la littérature : par exemple pour la détection active en SONAR [8], pour la détection et la localisation de particules de très hautes énergies (neutrinos) en acoustique sous-marine (projet ANTARES) [32] [33] et pour la détection de défauts des tuiles par mesures infrarouges sur un banc d'essai SATIR à Cadarache (projet de réacteur à fusion thermonucléaire) [34] [35].

Contribution et problématiques :

L'étude décrite dans cette thèse se situe dans ce contexte : nous nous sommes intéressés à résoudre deux problématiques liées au CSMF.

1/ Une extension du filtre CSMF plus performante dans le cas d'un environnement multi bruits.

La première problématique se pose dans le cadre de la détection multi bruits des signaux stationnaires par le CSMF : en effet, les applications ont montré que les trois méthodes proposées sont performantes mais on se trouve souvent dans l'incapacité de déterminer quelle est la meilleure d'entre elles en termes de détection (on constate le croisement de courbes COR comme on le verra dans le chapitre 2).

Nous proposons donc une extension du CSMF qui combine les trois méthodes (pour chacun des trois critères) pour tirer profit de chacune d'entre elles. Ceci nous a permis de mettre en place le filtre CSMF multicritère qui a prouvé son efficacité par rapport au filtrage CSMF à critère unique.

2/ Mettre en œuvre une version adaptative du CSMF pour la poursuite du sous-espace optimal.

La deuxième problématique se pose dans le cadre d'un seul bruit et de signaux non stationnaires *i.e.* dont les propriétés statistiques au second ordre évoluent dans le temps.

Nous nous sommes inspirés des méthodes de poursuite du sous-espace principal (pour lesquelles le bruit est blanc) tels que les algorithmes de type Subspace Tracking (SP).

Etant dans une situation d'un bruit non nécessairement blanc, il a fallu en adapter les principes au CSMF afin d'en déduire une version adaptative, le CSMF Subspace Tracking (CSMFST) qui recherche le sous-espace optimal.

Cette contribution permet le passage d'une poursuite de sous-espace principal engendré par les vecteurs principaux de la matrice de corrélation du signal, vers une poursuite du sous-espace optimal qui maximise le RSB.

À la suite de cette contribution, des articles ont été publiés dans le cadre de conférences internationales [36] [37].

Parmi les problèmes rencontrés à ce stade, l'inexistence d'algorithmes de poursuite (tracking) considérant un bruit autre que le bruit blanc dans les hypothèses de filtrage : de ce fait, le CSMFST n'est comparable à aucun autre algorithme, si ce n'est le CSMF qui ne fait pas de poursuite, dans le cas d'un bruit coloré.

Tous ces points seront abordés en détails dans ce manuscrit et seront illustrés par des exemples et des applications numériques.

Domaines applicatifs :

Comme on l'a déjà vu, les applications potentielles du CSMF et de ses extensions sont innombrables dans le traitement de signal : les signaux bioacoustiques (détection de cétacés ou de plongeurs pour la protection de fermes aquacoles), les systèmes RADAR et SONAR, le traitement d'images.

Description du document

Le document est organisé en quatre grandes parties.

Le filtrage adapté stochastique

Dans une première partie, on fait un rappel des travaux effectués précédemment concernant le filtre adapté et ses différentes extensions : le SMF qui maximise le RSB dans un sous-espace de dimension 1 et le CSMF qui maximise le RSB dans un sous-espace de dimension supérieure à un. Nous nous plaçons ici dans le cadre de signaux stationnaires pour lequel les matrices de covariances sont connues et fixes.

Pour le CSMF, nous présentons l'algorithme itératif qui permet d'obtenir la solution et qui s'applique dans les 2 cas :

- Détection : quand il y a un signal d'intérêt et un bruit.
- Classification (ou détection en mode multi-bruits) : quand on est en présence d'un signal d'intérêt et de plusieurs perturbations potentiellement présentes additivement.

Dans le cas de la détection, le critère à optimiser est unique tandis que dans celui de la classification, on peut envisager plusieurs critères à optimiser, on en présente trois : le critère arithmétique, géométrique et harmonique. Bien sûr, on cherche à maximiser l'un de ces critères.

Le filtrage multicritère

Dans ce chapitre, on introduit la première contribution de ce travail, la technique du « filtrage multicritère » utilisée en classification, qui prend en compte simultanément les trois critères énoncés précédemment pour le filtrage des signaux.

Cette méthode est développée et appliquée à des signaux simulés puis réels issus du monde de l'acoustique sous-marine. Ainsi nous avons pu détecter des signaux de dauphins et de cachalots. Les résultats de filtrage obtenus montrent son efficacité, supérieure à celle du CSMF.

L'étude des courbes COR permet de comparer les différents résultats de filtrages obtenus et de déterminer la méthode la plus performante.

Poursuite du sous-espace optimal :

Dans cette partie, on aborde le problème de poursuite du sous-espace optimal en temps réel dans le cas de processus aléatoires stationnaires ou non stationnaires.

Dans un premier temps, nous rappelons le problème classique de la poursuite du sous-espace signal ou principal (dont on connaît la dimension, situation classique et même systématique dans la littérature) dans le cas d'un signal d'intérêt noyé dans un bruit blanc.

Nous présentons ensuite quelques algorithmes de type Subspace-Projection (SP) utilisés pour résoudre ce problème dans le cas de signaux stationnaires ou non-stationnaires.

Nous présentons enfin la seconde contribution de notre travail, à savoir la poursuite du sous-espace optimal dans le cas de signaux stationnaires ou non-stationnaires dans un bruit qui n'est plus forcément blanc.

A cette fin, nous proposons un algorithme appelé CSMFST qui s'inspire des algorithmes de type SP et bien évidemment de l'algorithme CSMF présenté au premier chapitre.

La convergence de cet algorithme sera démontrée.

Applications et tests

Dans ce dernier chapitre, on met en œuvre les résultats du chapitre précédent, tout d'abord sur des signaux simulés afin de valider l'algorithme, puis sur des signaux réels d'acoustique sous-marine, domaine d'application qui nous intéresse particulièrement. En effet, afin de préserver le patrimoine marin, les scientifiques cherchent des solutions pour limiter l'impact des activités humaines sur les populations de cétacés et lancent des études [38] afin de déterminer les zones où les risques de collision sont les plus importants. Dans le cadre de détection d'animaux marins, on s'intéresse aux sons de type impulsif émis par les cétacés.

Dans cette partie, on teste aussi des variantes du CSMFST qui dépendent de la configuration de l'expérimentation.

Conclusion et perspectives

Dans cette dernière partie, on résume le travail réalisé puis on présente les perspectives envisageables concernant outre l'amélioration du nouvel algorithme CSMFST, de nouvelles applications.

Chapitre I :

Le filtre adapté stochastique

I.1 Introduction

Dans ce premier chapitre, on commence par présenter le Filtre Adapté Stochastique (FAS) ou *Stochastic Matched Filter* (SMF) et son concept général.

On présente ensuite le Filtre Adapté Stochastique sous Contrainte (FASC) ou *Constrained Stochastic Matched Filter* (CSMF) qui maximise le rapport signal à bruit (RSB) dans un sous-espace de dimension optimale dans le cas de la détection. On expose les caractéristiques du sous-espace optimal et on donne l'algorithme permettant de trouver ce filtre optimal.

Dans une seconde partie, on s'intéresse à la généralisation du FASC pour la détection d'un processus d'intérêt noyé potentiellement dans un ou plusieurs bruits additifs : on parle alors, abusivement, de classification des signaux.

I.2 Le filtre adapté stochastique appliqué à la détection

Tout au long de ce chapitre, les signaux considérés sont discrets et représentés par des vecteurs d'un espace vectoriel réel de dimension n que l'on note E^n .

On suppose qu'on est en présence d'un processus d'intérêt \mathbf{s} , que l'on cherche à détecter, noyé dans un bruit aléatoire additif \mathbf{b} .

Tous les signaux sont centrés (c'est l'hypothèse la plus défavorable) et de matrices de covariance connues notées respectivement \mathbf{A} et \mathbf{B} pour le signal à détecter et le bruit. $\mathbf{A}, \mathbf{B} \in M_{n \times n}$ sont symétriques et strictement définies positives (donc inversibles).

On considère dans ce chapitre que les processus sont invariants dans le temps, et donc les matrices \mathbf{A} et \mathbf{B} qui, seules sont nécessaires au calcul du filtre optimal, sont constantes ; elles peuvent être estimées à partir de réalisations des signaux.

Le but de cette partie est de présenter le SMF puis le CSMF qui en est une extension.

I.2.1 Le filtre adapté (MF)

Le Filtre Adapté (FA) ou *Matched Filter* (MF) est un opérateur linéaire \mathbf{h}^* qui rend maximal le rapport de la puissance instantanée d'un signal \mathbf{s} déterministe (de forme connue) à la puissance moyenne du bruit centré dont les propriétés statistiques à l'ordre deux sont connues (par sa matrice de covariance \mathbf{B}) :

$$RSB_f \equiv \frac{(\mathbf{h}^T \mathbf{s})^2}{\mathbf{h}^T \mathbf{B} \mathbf{h}} = \frac{\mathbf{h}^T \mathbf{s} \mathbf{s}^T \mathbf{h}}{\mathbf{h}^T \mathbf{B} \mathbf{h}}.$$

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

En effet, si la mesure $\mathbf{m} = \mathbf{s} + \mathbf{b}$ est l'entrée du filtre, la sortie du filtre s'écrit : $\mathbf{h}^T \mathbf{s} + \mathbf{h}^T \mathbf{b}$, le premier terme étant déterministe et le second aléatoire.

Le MF $\mathbf{h} \in E^n$ est donné par :

$$\mathbf{h}^* = \mathbf{B}^{-1} \mathbf{s} = \operatorname{argmax}_{\mathbf{h} \in E^n} \frac{(\mathbf{h}^T \mathbf{s})^2}{\mathbf{h}^T \mathbf{B} \mathbf{h}}.$$

C'est le résultat, optimal, que l'on trouve au sens du maximum de vraisemblance si le bruit est gaussien. Mais cette hypothèse de gaussianité n'est pas nécessaire quand on maximise le RSB : toutefois ça « valide » le résultat.

I.2.2 Le filtre adapté stochastique (SMF)

Dans le cas d'un signal \mathbf{s} non plus déterministe mais aléatoire et stationnaire (de propriétés statistiques invariantes dans le temps) de matrice de covariance \mathbf{A} , on parle de Filtre Adapté Stochastique (FAS) ou *Stochastic Matched Filter* (SMF).

Ce dernier consiste à projeter le signal mesuré dans un sous-espace de dimension $p = 1$ engendré par un vecteur normé \mathbf{h}^* dit optimal, tel que le rapport signal-à-bruit y est maximisé.

Ainsi, le SMF est une généralisation du MF.

On montre que le rapport signal à bruit en sortie du filtre linéaire \mathbf{h} , peut se mettre sous la forme d'un quotient de Rayleigh :

$$RSB_f = \frac{\mathbf{h}^T \mathbf{A} \mathbf{h}}{\mathbf{h}^T \mathbf{B} \mathbf{h}}. \quad (\text{I.1})$$

où le filtre est $\mathbf{h} \in E^n$.

Note : pour un signal $\mathbf{s} \in E^n$ de matrice de covariance \mathbf{A} et de puissance P_s , on a :

$\mathbf{A} \equiv \mathbb{E}(\mathbf{s}\mathbf{s}^T)$ et $P_s \equiv \mathbb{E}(\mathbf{s}^T \mathbf{s}) = \mathbb{E}(\operatorname{tr}(\mathbf{s}\mathbf{s}^T)) = \operatorname{tr}(\mathbf{A})$ et $P_{s'} \equiv \mathbb{E}(\mathbf{s}'^T \mathbf{s}') = \mathbf{h}^T \mathbf{A} \mathbf{h}$, où $\mathbf{s}' = \mathbf{h}^T \mathbf{s}$ est la sortie du filtre.

Si l'on normalise la puissance du signal, *i.e.* $P_s = 1$, alors $\operatorname{tr}(\mathbf{A}) = 1$.

Si l'on choisit de normaliser les processus d'intérêt \mathbf{s} et \mathbf{b} , on obtient alors

$$RSB_f = \frac{\operatorname{tr} \mathbf{A}}{\operatorname{tr} \mathbf{B}} \cdot \frac{\mathbf{h}^T \tilde{\mathbf{A}} \mathbf{h}}{\mathbf{h}^T \tilde{\mathbf{B}} \mathbf{h}}$$

avec $\tilde{\mathbf{A}} = \frac{\mathbf{A}}{\operatorname{tr}(\mathbf{A})}$ et $\tilde{\mathbf{B}} = \frac{\mathbf{B}}{\operatorname{tr}(\mathbf{B})}$ les matrices de covariance des processus normalisés.

Dans l'expression de RSB_f , il intervient le terme suivant

$$RSB_i \equiv \frac{\operatorname{tr} \mathbf{A}}{\operatorname{tr} \mathbf{B}} = \frac{P_s}{P_b}$$

qui est le RSB avant filtrage, ou RSB initial.

Maximiser RSB_f revient donc à maximiser le gain sur le RSB suivant :

$$\rho \equiv \frac{RSB_f}{RSB_i} = \frac{\mathbf{h}^T \tilde{\mathbf{A}} \mathbf{h}}{\mathbf{h}^T \tilde{\mathbf{B}} \mathbf{h}}. \quad (\text{I.2})$$

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

Il a été démontré que le filtre optimal \mathbf{h}^* défini par

$$\mathbf{h}^* = \operatorname{argmax}_{\mathbf{h} \in E^n} \frac{\mathbf{h}^T \tilde{\mathbf{A}} \mathbf{h}}{\mathbf{h}^T \tilde{\mathbf{B}} \mathbf{h}} = \operatorname{argmax}_{\substack{\mathbf{h} \in E^n \\ \mathbf{h}^T \mathbf{h} = 1}} \frac{\mathbf{h}^T \tilde{\mathbf{A}} \mathbf{h}}{\mathbf{h}^T \tilde{\mathbf{B}} \mathbf{h}}$$

vérifie :

$$\tilde{\mathbf{A}} \mathbf{h}^* = \rho^* \tilde{\mathbf{B}} \mathbf{h}^* \quad (\text{I.3})$$

et que $\rho^* > 1$ [30]. \mathbf{h}^* est alors le vecteur propre associé à la plus grande valeur propre ρ^* de $\tilde{\mathbf{B}}^{-1} \tilde{\mathbf{A}}$.

$\rho^* = \frac{\mathbf{h}^{*T} \tilde{\mathbf{A}} \mathbf{h}^*}{\mathbf{h}^{*T} \tilde{\mathbf{B}} \mathbf{h}^*}$ s'interprète comme le gain sur le RSB que l'on obtient si l'on projette les mesures sur un sous-espace de dimension 1 porté par le vecteur \mathbf{h}^* : ce sous-espace sera noté \mathcal{E}_1^* .

Néanmoins, $\tilde{\mathbf{B}}^{-1} \tilde{\mathbf{A}}$ possède n valeurs propres, dont certaines peuvent être plus grandes que 1, voire proches de ρ^* , et peuvent également s'interpréter comme le gain sur le RSB que l'on obtient si l'on projette les mesures sur un sous-espace de dimension 1 porté par le vecteur propre associé. Il a été montré [5] que dans bien de situations, il est avantageux (en termes de détection, le juge étant la courbe COR) de prendre en compte un certain nombre $p > 1$ de directions propres et de projeter les mesures dans le sous-espace de dimension p engendré par celles-ci. Ce filtre a été appelé FAS Etendu (FASE) ou *Extended SMF* (ESMF).

Toutefois, à l'exception du cas particulier où le bruit est blanc ($\tilde{\mathbf{B}} = \mathbb{I}_n$; on obtient la base de Kahrnenen-Loève), ce sous-espace de dimension p n'est pas celui dans lequel est maximisé le RSB. De cette remarque découle la recherche du sous-espace optimal de dimension p et le CSMF.

I.2.3 Le filtre adapté stochastique sous contrainte (CSMF)

Le Filtre Adapté Stochastique sous Contrainte (FASC) ou *Constrained SMF* (CSMF) a pour objectif de maximiser le RSB dans un sous-espace de dimension quelconque $p \geq 1$ fixée *a priori*. La contrainte dont il est question est ainsi la prise en compte de la dimension du sous-espace dans lequel on fait la projection des signaux.

Le CSMF est une extension du SMF, dans le sens où pour $p = 1$ on obtient le même résultat.

Contrairement au ESMF, il peut être généralisé à la détection d'un processus d'intérêt \mathbf{s} noyé potentiellement dans un ou plusieurs bruits additifs. Cela fera l'objet d'un autre paragraphe.

Evidemment, la puissance d'un processus aléatoire ne dépend que du sous-espace dans lequel il est défini et non de la base qui génère ce sous-espace. Cette propriété est donc aussi vérifiée par le RSB.

Ainsi, afin de simplifier les calculs, on caractérise le sous-espace optimal recherché par une base orthonormée (il en existe une infinité).

Considérons des processus normalisés. Le RSB avant filtrage s'écrit dans E^n :

Le filtre adapté stochastique

1.2 Le filtre adapté stochastique appliqué à la détection

$$RSB_n = \frac{\text{tr}(\mathbf{A})}{\text{tr}(\mathbf{B})} = 1.$$

En effet, soit $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_n] \in M_{n \times n}$ une base orthonormée de E^n (on a donc $\mathbf{V}^T \mathbf{V} = \mathbf{V} \mathbf{V}^T = \mathbb{I}_n$).

On montre que

$$RSB_n = \frac{\sum_{i=1}^n \mathbf{v}_i^T \mathbf{A} \mathbf{v}_i}{\sum_{i=1}^n \mathbf{v}_i^T \mathbf{B} \mathbf{v}_i} = \frac{\text{tr}(\mathbf{V}^T \mathbf{A} \mathbf{V})}{\text{tr}(\mathbf{V}^T \mathbf{B} \mathbf{V})} = \frac{\text{tr}(\mathbf{A} \mathbf{V} \mathbf{V}^T)}{\text{tr}(\mathbf{B} \mathbf{V} \mathbf{V}^T)} = \frac{\text{tr}(\mathbf{A})}{\text{tr}(\mathbf{B})}.$$

Soit E^p un sous-espace de dimension $p < n$ et $\mathbf{X}_p = [\mathbf{x}_1 \dots \mathbf{x}_p] \in M_{n \times p}$ une base orthonormée de E^p ($\mathbf{x}_i^T \mathbf{x}_j = \delta_{ij}$ ou $\mathbf{X}_p^T \mathbf{X}_p = \mathbb{I}_p$: attention, $\mathbf{X}_p \mathbf{X}_p^T \neq \mathbb{I}_n$). Alors, on montre de même que le RSB dans E^p est

$$RSB_p = \frac{\sum_{i=1}^p \mathbf{x}_i^T \mathbf{A} \mathbf{x}_i}{\sum_{i=1}^p \mathbf{x}_i^T \mathbf{B} \mathbf{x}_i} = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B} \mathbf{X}_p)}. \quad (\text{I.4})$$

Note : on peut introduire les projecteurs $\mathbf{P}_i = \mathbf{x}_i \mathbf{x}_i^T \in M_{n \times n}$ qui vérifient $\mathbf{P}_i^T = \mathbf{P}_i$ et $\mathbf{P}_i \mathbf{P}_j = \delta_{ij} \mathbf{P}_i$.

On note $\mathbf{P}_{total} = \mathbf{X}_p \mathbf{X}_p^T = \sum_{i=1}^p \mathbf{P}_i \in M_{n \times n}$ la projection sur l'espace $E^p = \text{Span}\{\mathbf{x}_1, \dots, \mathbf{x}_p\}$ parallèlement à $E_\perp^p = \text{Span}\{\mathbf{x}_{p+1}, \dots, \mathbf{x}_n\}$: on a bien $\mathbf{P}_{total}^T = \mathbf{P}_{total}$ et $\mathbf{P}_{total}^2 = \mathbf{P}_{total}$.

On peut donc calculer le projeté $\hat{\mathbf{s}}$ de \mathbf{s} sur E^p :

$$\hat{\mathbf{s}} = \mathbf{P}_{total} \mathbf{s} = \mathbf{X}_p \mathbf{X}_p^T \mathbf{s},$$

dont la matrice de covariance s'écrit $\hat{\mathbf{A}} = \mathbb{E}(\hat{\mathbf{s}} \hat{\mathbf{s}}^T)$ et la puissance s'écrit :

$$\hat{P} = \text{tr}(\hat{\mathbf{A}}) = \text{tr}(\mathbf{P}_{total} \mathbf{A} \mathbf{P}_{total}^T) = \text{tr}(\mathbf{A} \mathbf{P}_{total}^T \mathbf{P}_{total}) = \text{tr}(\mathbf{A} \mathbf{P}_{total}) = \text{tr}(\mathbf{A} \mathbf{X}_p \mathbf{X}_p^T) = \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)$$

Le but est de maximiser le rapport RSB_p . Cela revient à chercher parmi tous les espaces de dimension p possibles (p étant fixé *a priori*), celui dans lequel le RSB atteint la valeur maximale.

Ce sous-espace dit optimal sera noté \mathcal{E}^* .

Eléments caractéristiques du sous-espace optimal

Le sous-espace optimal \mathcal{E}^* est généré par un ensemble de p vecteurs orthonormés $\{\mathbf{x}_i\}_{i=1, \dots, p}$.

Le gain ρ est maximal dans ce sous-espace et son expression est donné par :

$$\rho = \frac{\sum_{i=1}^p \mathbf{x}_i^T \mathbf{A} \mathbf{x}_i}{\sum_{i=1}^p \mathbf{x}_i^T \mathbf{B} \mathbf{x}_i} = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B} \mathbf{X}_p)},$$

avec $\text{tr}(\mathbf{A}) = \text{tr}(\mathbf{B}) = 1$, $\mathbf{X}_p = [\mathbf{x}_1 \dots \mathbf{x}_p]$ et $\mathbf{x}_i^T \mathbf{x}_j = \delta_{ij}$.

Les inconnues du problème sont :

- La valeur maximale de ρ notée ρ^* .
- Les p vecteurs orthonormés $\mathbf{X}_p \in M_{n \times p}$ qui maximisent ρ .

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

Afin de résoudre ce problème d'optimisation sous contraintes de type égalité, on utilise la méthode des multiplicateurs de Lagrange. La contrainte $\mathbf{X}_p^T \mathbf{X}_p = \mathbb{I}_p$ est indispensable puisque l'expression de ρ qui a été trouvée l'a été sous l'hypothèse d'être calculée dans une base orthonormée.

On définit le lagrangien avec contraintes égalité :

$$\mathcal{L}(\mathbf{X}_p, \boldsymbol{\Omega}) = \rho + \sum_{i=1}^p \sum_{j=i}^p \omega_{ij} (\mathbf{x}_i^T \mathbf{x}_j - \delta_{ij}),$$

où $\boldsymbol{\Omega} \equiv [\omega_{ij}] \in M_{p \times p}$ est symétrique : ses éléments ω_{ij} sont les multiplicateurs de Lagrange.

Cette expression s'écrit aussi sous la forme matricielle :

$$\mathcal{L}(\mathbf{X}_p, \boldsymbol{\Omega}) = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B} \mathbf{X}_p)} + \text{tr}(\boldsymbol{\Omega} (\mathbf{X}_p^T \mathbf{X}_p - \mathbb{I}_p)). \quad (\text{I.5})$$

Pour trouver la valeur maximale du rapport signal à bruit, il faut résoudre (condition nécessaire) l'équation $\frac{\partial \mathcal{L}}{\partial \mathbf{X}_p} = 0$, ce qui est vérifié pour tout extremum local ; il reste ensuite à vérifier que cet extremum est un maximum.

Ceci conduit à résoudre l'équation suivante (qui n'est pas une équation aux valeurs propres) :

$$(\mathbf{A} - \rho \mathbf{B}) \mathbf{X}_p = \mathbf{X}_p \boldsymbol{\Omega}_0 \quad (\text{I.6})$$

où $\boldsymbol{\Omega}_0 = -\text{tr}(\mathbf{X}_p^T \mathbf{B} \mathbf{X}_p) \boldsymbol{\Omega} \in M_{p \times p}$ est une matrice symétrique réelle donc diagonalisable.

On peut alors faire intervenir une matrice réelle unitaire $\boldsymbol{\Pi} \in M_{p \times p}$ et une matrice réelle diagonale $\Delta_u = [\mu_i] \in M_{p \times p}$ telles que :

$$\boldsymbol{\Omega}_0 \boldsymbol{\Pi} = \boldsymbol{\Pi} \Delta_u.$$

L'équation (I.6) conduit à :

$$(\mathbf{A} - \rho \mathbf{B}) \mathbf{X}_p \boldsymbol{\Pi} = \mathbf{X}_p \boldsymbol{\Pi} \Delta_u.$$

qui est une équation aux valeurs propres si l'on pose $\mathbf{W}_p = \mathbf{X}_p \boldsymbol{\Pi} = [\mathbf{w}_1 \dots \mathbf{w}_p] \in M_{n \times p}$.

\mathbf{W}_p engendre le même sous-espace que \mathbf{X}_p puisque $\boldsymbol{\Pi}$ est unitaire : $\text{Span}\{\mathbf{W}_p\} = \text{Span}\{\mathbf{X}_p\}$:

$$(\mathbf{A} - \rho \mathbf{B}) \mathbf{W}_p = \mathbf{W}_p \Delta_u. \quad (\text{I.7})$$

L'équation (I.7) constitue un ensemble de p équations aux valeurs propres de la matrice $\mathbf{A} - \rho \mathbf{B}$ qui est de dimension $n \times n$ et en compte donc n .

Il y a deux problèmes : tout d'abord, la valeur optimale de ρ , notée ρ^* , est inconnue, et en conséquence $\mathbf{A} - \rho^* \mathbf{B}$ également. Ensuite, $(\mathbf{A} - \rho^* \mathbf{B}) \in M_{n \times n}$ mais seuls p des vecteurs propres nous intéressent, les \mathbf{w}_i^* (ou $\mathbf{W}_p^* \in M_{n \times p}$) qui vérifient que

$$\rho^* = \frac{\sum_{i=1}^p \mathbf{w}_i^{*T} \mathbf{A} \mathbf{w}_i^*}{\sum_{i=1}^p \mathbf{w}_i^{*T} \mathbf{B} \mathbf{w}_i^*} = \frac{\text{tr}(\mathbf{W}_p^{*T} \mathbf{A} \mathbf{W}_p^*)}{\text{tr}(\mathbf{W}_p^{*T} \mathbf{B} \mathbf{W}_p^*)},$$

est maximum.

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

Il n'est donc pas possible de calculer simplement \mathbf{W}_p^* et ρ^* , et une procédure algorithmique doit être mise en œuvre pour atteindre ces valeurs.

La matrice $\mathbf{A} - \rho\mathbf{B}$, qui dépend de la valeur de ρ , est diagonalisable car symétrique réelle et \mathbf{W}_p constitue un ensemble de vecteur orthogonaux, que l'on peut normer.

Ce qui nous intéresse, c'est de trouver les p vecteurs inconnus \mathbf{W}_p ainsi que les valeurs propres de $\mathbf{A} - \rho\mathbf{B}$.

On note $\mathbf{M}(\rho) = \mathbf{A} - \rho\mathbf{B}$ cela nous permet d'écrire :

$$\mathbf{M}(\rho)\mathbf{W}_p(\rho) = \mathbf{W}_p(\rho)\Delta_u(\rho) \quad (\text{I.8})$$

et de montrer que \mathbf{W}_p et Δ_u dépendent évidemment de ρ .

Choix des vecteurs du sous-espace optimal

Dans cette partie, on se propose de donner la relation entre les vecteurs propres ou les valeurs propres qui permettent de caractériser le sous-espace optimal recherché \mathcal{E}_p^* .

On peut écrire l'équation (I.8) sous la forme :

$$\mathbf{M}(\rho)\mathbf{w}_i = \mu_i\mathbf{w}_i, \quad i = 1, \dots, n.$$

Il a été montré [5] que dans $\mathcal{E}_p^* = \text{Span}\{\mathbf{w}_1^*, \dots, \mathbf{w}_p^*\}$, si l'on classe les valeurs propres :

$$\mu_1^* \geq \mu_2^* \geq \dots \geq \mu_n^*$$

alors

$$\sum_{i=1}^p \mu_i^* = 0. \quad (\text{I.9})$$

Ce sont les p plus grandes valeurs propres (dont nécessairement certaines négatives) auxquelles sont associés les vecteurs propres optimaux.

Ainsi, l'expression du gain maximum ρ^* est donnée par :

$$\rho^* = \frac{\sum_{i=1}^p \mathbf{w}_i^{*\top} \mathbf{A} \mathbf{w}_i^*}{\sum_{i=1}^p \mathbf{w}_i^{*\top} \mathbf{B} \mathbf{w}_i^*} = \frac{\text{tr}(\mathbf{W}_p^{*\top} \mathbf{A} \mathbf{W}_p^*)}{\text{tr}(\mathbf{W}_p^{*\top} \mathbf{B} \mathbf{W}_p^*)} \quad (\text{I.10})$$

Cas du bruit blanc

Dans le cas où le bruit est blanc (et normalisé donc de puissance 1), on a $\mathbf{B} = \mathbb{I}_n$. On peut alors calculer simplement, sans mise en place d'un algorithme particulier, le sous-espace optimal. Il est engendré par les vecteurs propres principaux de \mathbf{A} . En effet, résoudre

$$(\mathbf{A} - \rho\mathbb{I}_n)\mathbf{W}_p = \mathbf{W}_p\Delta_u,$$

revient à résoudre

$$\mathbf{A}\mathbf{W}_p = \mathbf{W}_p(\Delta_u + \rho\mathbb{I}_n) = \mathbf{W}_p\Delta,$$

soit $\Delta_u = \Delta - \rho\mathbb{I}_n$ où Δ est la matrice des valeurs propres de \mathbf{A} .

Si l'on trie les valeurs propres de \mathbf{A} ,

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

$$\lambda_1 \geq \dots \geq \lambda_p > \lambda_{p+1} \geq \dots \geq \lambda_n > 0,$$

alors

$$\mu_i = \lambda_i - \rho$$

et pour l'optimum,

$$\sum_{i=1}^p \mu_i^* = 0 \quad \Rightarrow \quad \sum_{i=1}^p \lambda_i - p\rho^* = 0$$

soit

$$\rho^* = \frac{1}{p} \sum_{i=1}^p \lambda_i.$$

Cette expression est valable pour une valeur de p fixée *a priori*.

Cas où $p = 1$:

Le gain maximal ρ^* obtenu dans un sous-espace de dimension 1 correspond à la plus grande valeur propre λ^* de la matrice $\mathbf{B}^{-1}\mathbf{A}$.

$$\rho^* = \frac{1}{p} \sum_{i=1}^p \lambda_i = \lambda_1 = \lambda^*.$$

On a

$$\mu_1 = \lambda_1 - \rho$$

Donc

$$\mu_1 = \mu^* = 0.$$

D'où la vérification de la condition $\sum_{i=1}^p \mu_i^* = 0$ pour $p = 1$.

Pour maximiser ρ et trouver \mathcal{E}_p^* , il faut prendre les vecteurs propres $\mathbf{W}_p^* = \mathbf{W}_p(1:p)$ associés aux valeurs propres principales de \mathbf{A} (et donc naturellement orthogonaux). C'est la décomposition de Kahrnen-Loève. \mathcal{E}_p^* s'appelle alors le sous-espace principal, lié au signal d'intérêt noyé dans un bruit blanc.

On a le même résultat si l'on cherche $\mathbf{W} \in M_{n \times p}$ qui maximise $\text{tr}(\mathbf{W}^T \mathbf{A} \mathbf{W})$ sous la contrainte $\mathbf{W}^T \mathbf{W} = \mathbb{I}_p$.

$$\mathbf{W}_p^* = \underset{\substack{\mathbf{W} \in M_{n \times p} \\ \mathbf{W}^T \mathbf{W} = \mathbb{I}_p}}{\text{argmax}} \frac{\text{tr}(\mathbf{W}^T \mathbf{A} \mathbf{W})}{\text{tr}(\mathbf{W}^T \mathbf{W})} = \underset{\substack{\mathbf{W} \in M_{n \times p} \\ \mathbf{W}^T \mathbf{W} = \mathbb{I}_p}}{\text{argmax}} \text{tr}(\mathbf{W}^T \mathbf{A} \mathbf{W}).$$

Algorithme

Dans cette partie, on rappelle l'algorithme [30,5] qui permet de trouver le sous-espace optimal \mathcal{E}_p^* dans lequel est maximisé le RSB.

Le filtre adapté stochastique

1.2 Le filtre adapté stochastique appliqué à la détection

\mathcal{E}_p^* est engendré par p vecteurs qui sont solutions de l'équation (1.8).

L'algorithme alterne assez naturellement, après initialisation, le calcul de $\mathbf{M}(\rho)$ et des éléments propres associés, et le choix des p vecteurs propres pour lesquels le RSB est le plus grand.

Il opère donc de la manière suivante :

- tout d'abord, on se fixe une valeur initiale de ρ appelée $\rho^{(0)}$; par exemple, on peut choisir de prendre la plus grande valeur propre de $\mathbf{B}^{-1}\mathbf{A}$ qui correspond au gain maximal obtenu dans un sous-espace de dimension 1.

- par la suite, on exprime la matrice symétrique $\mathbf{M}_0 = \mathbf{M}(\rho^{(0)})$ dont on calcule les valeurs propres et les vecteurs propres afin de choisir parmi ceux-ci les p pour lesquels le RSB est le plus grand.

Ces p vecteurs engendrent un sous-espace $\mathcal{E}_p^{(0)}$ dans lequel le RSB sera noté $\rho^{(1)}$; l'itération suivante consiste à calculer la nouvelle matrice $\mathbf{M}_1 = \mathbf{M}(\rho^{(1)})$ et le nouveau sous-espace $\mathcal{E}_p^{(1)}$.

On a, pour $k \geq 0$:

$$\mathbf{M}_k \equiv \mathbf{A} - \rho^{(k)}\mathbf{B}$$

$$\mathbf{M}_k \mathbf{w}_{i,k} = \mu_{i,k} \mathbf{w}_{i,k} \quad 1 \leq i \leq n$$

et

$$\rho^{(k+1)} = \frac{\sum_{i \in J_k} \mathbf{w}_{i,k}^T \mathbf{A} \mathbf{w}_{i,k}}{\sum_{i \in J_k} \mathbf{w}_{i,k}^T \mathbf{B} \mathbf{w}_{i,k}},$$

où J_k est la combinaison parmi les C_n^p de p éléments de $\{1, 2, \dots, n\}$ avec $\text{card}(J_k) = p$ pour laquelle on obtient la valeur maximale de $\rho^{(k+1)}$.

On note $\mathcal{E}_p^{(k)} \equiv \text{Span}\{\mathbf{w}_{1,k}, \dots, \mathbf{w}_{p,k}\}$ le sous-espace estimé à la $k^{\text{ème}}$ itération.

Cette procédure sera répétée jusqu'à ce qu'on atteigne la convergence, ou plutôt la condition d'arrêt, celle-ci pouvant être soit :

- la somme des valeurs propres est inférieure à une valeur seuil donnée :

$$\left| \sum_{i=1}^p \mu_{i,k} \right| < \varepsilon$$

- l'écart entre deux valeurs consécutives de $\rho^{(k)}$ aux étapes k et $k + 1$ est inférieure à une valeur seuil donnée :

$$|\rho^{(k+1)} - \rho^{(k)}| < \varepsilon$$

Ces deux conditions nous permettent d'avoir deux versions différentes de l'algorithme et il a été démontré dans [5] qu'elles donnent le même résultat :

La première condition considère les p vecteurs propres correspondant aux valeurs propres dont la somme est nulle.

Le filtre adapté stochastique

I.2 Le filtre adapté stochastique appliqué à la détection

La deuxième condition assure que le rapport signal-à-bruit ne varie pas de manière significative : c'est cette condition qui sera principalement considérée dans les applications numériques de l'algorithme.

La convergence de cet algorithme a été démontrée [30] et il a été prouvé que ρ^* est unique.

Pseudo-code de l'algorithme CSMF

Choix de la valeur initiale de $\rho^{(0)}$ = la plus grande valeur propre de $\mathbf{B}^{-1}\mathbf{A}$

Boucle sur l'indice k de 0 jusqu'à $+\infty$

Tant que $\{|\rho^{(k+1)} - \rho^{(k)}| > \varepsilon\}$, faire

1. Calcul de la matrice $\mathbf{M}_k = \mathbf{A} - \rho^{(k)}\mathbf{B}$.
2. Calcul des valeurs propres $\mu_{i,k}$ et des vecteurs propres $\mathbf{w}_{i,k}$ de \mathbf{M}_k , $1 \leq i \leq n$.
3. Calcul des p vecteurs propres pour lesquels le RSB $\rho^{(k+1)}$ est maximal parmi les n vecteurs calculés.

Fin de la boucle

$$\begin{aligned}\mu_i^* &= \mu_{i,k}, \quad 1 \leq i \leq p \\ \mathbf{w}_i^* &= \mathbf{w}_{i,k}, \quad 1 \leq i \leq p \implies \mathcal{E}_p^* = \text{Span}\{\mathbf{w}_1^*, \dots, \mathbf{w}_p^*\} \\ \rho^* &= \rho^{(k+1)}\end{aligned}$$

Fin

Cet algorithme permet de déterminer le sous-espace optimal \mathcal{E}_p^* dans lequel ρ est maximisé.

Note : cet algorithme est lourd en termes de calculs puisqu'il nécessite de trouver à chaque itération les éléments propres d'une matrice $n \times n$, mais également le calcul de C_n^p valeurs possibles de $\rho^{(k+1)}$ afin d'en trouver la plus grande.

D'après la remarque selon laquelle ce sont, à la convergence, les plus grandes valeurs propres qu'il faut retenir, une version allégée (toujours vérifiée mais pas démontrée) consiste à ne calculer à chaque itération, après classement des $\mu_{i,k}$, que

$$\rho^{(k+1)} = \frac{\sum_{i=1}^p \mathbf{w}_{i,k}^T \mathbf{A} \mathbf{w}_{i,k}}{\sum_{i=1}^p \mathbf{w}_{i,k}^T \mathbf{B} \mathbf{w}_{i,k}}.$$

Sur l'ensemble des exemples traités, on a constaté que cette version simplifiée donne les mêmes résultats que la version optimale de l'algorithme et ce avec un temps de calcul considérablement réduit.

Dans ce qui suit, nous étendons ce problème à celui d'un signal d'intérêt mélangé avec K processus perturbateurs et donc à un problème de classification de signaux.

I.3 Le CSMF appliqué à la classification

Dans cette partie, au lieu de considérer un processus d'intérêt noyé dans un bruit aléatoire, on suppose qu'on est en présence de plusieurs perturbations potentiellement présentes, seules ou simultanément. Dans ce cas-là, on considère que le signal d'intérêt s est décorrélé de chacun des K bruits aléatoires étudiés, eux-mêmes décorrélés deux à deux.

Ceci veut dire, qu'au lieu d'avoir seulement deux hypothèses :

$$H_0 : \mathbf{b}$$

$$H_1 : \mathbf{s} + \mathbf{b}$$

L'hypothèse H_1 suppose que nous avons une multitude de types de bruits présents simultanément telles que :

$$\mathbf{b} = \mathbf{b}_1; \mathbf{b} = \mathbf{b}_1 + \mathbf{b}_3 \text{ ou } \mathbf{b} = \mathbf{b}_2 + \mathbf{b}_4 \text{ etc...}$$

Notre objectif est le même : rechercher le sous-espace optimal \mathcal{E}_p^* dans lequel est maximisé le rapport signal à bruit.

On note \mathbf{A} la matrice de covariance du signal à détecter s et \mathbf{B}_i les matrices de covariance des différentes perturbations \mathbf{b}_i .

La problématique est que le signal d'intérêt peut être noyé dans l'un des K bruits potentiels, voire dans un mélange additif de ces bruits. Ceci rend la détection plus difficile.

Par permutation circulaire, on peut faire jouer à chaque processus le rôle du processus d'intérêt. Le problème devient alors symétrique et on pourra faire une classification de signaux.

Comme précédemment, on peut définir autant de gains élémentaires que de bruits présents après projection dans le sous espace de dimension p . Ces gains sont tous strictement positifs et leur expression est :

$$\rho_i = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}, \quad \forall i = 1, \dots, K, \quad (\text{I.11})$$

avec $\mathbf{X}_p = [\mathbf{x}_1 \dots \mathbf{x}_p] \in M_{n \times p}$.

Il est donc nécessaire de définir un critère à maximiser qui conduira à un nouveau sous-espace optimal.

Critères possibles

On décide de construire des critères basés sur les trois moyennes suivantes des gains élémentaires :

- La moyenne géométrique,
- La moyenne arithmétique,
- La moyenne harmonique.

Le filtre adapté stochastique

I.3 Le CSMF appliqué à la classification

➤ Moyenne géométrique

La moyenne géométrique des gains est donnée par l'expression :

$$\rho = \sqrt[K]{\prod_{i=1}^K \rho_i} = \sqrt[K]{\prod_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}}.$$

Plutôt que maximiser ρ , on maximise ρ^K . On choisit donc de maximiser la puissance au lieu de la racine $K^{\text{ème}}$ pour simplifier les calculs.

Tout comme dans le chapitre précédent, il s'agit d'un problème d'optimisation sous contrainte. Nous utilisons donc la fonction de Lagrange :

$$\mathcal{L}(\mathbf{X}_p, \boldsymbol{\Omega}) = \rho^K + \text{tr}(\boldsymbol{\Omega}(\mathbf{X}_p^T \mathbf{X}_p - \mathbb{I}_p)). \quad (\text{I.12})$$

Cette fonction atteint son maximum quand $\frac{\partial \mathcal{L}}{\partial \mathbf{X}_p} = 0$, c'est-à-dire, après calcul (cf. annexe 1) :

$$\left(\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \rho_i \mathbf{B}_i \right) \mathbf{X} = \mathbf{X} \boldsymbol{\Omega}_0, \quad (\text{I.13})$$

expression où intervient la moyenne arithmétique des \mathbf{B}_i pondérées par des poids $\alpha_i = \rho_i$.

La matrice $\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \rho_i \mathbf{B}_i$ dépend des K paramètres ρ_i : nous la notons $\mathbf{M}(\rho_1, \dots, \rho_K)$.

Le terme lié à la matrice de covariance du bruit $\rho \mathbf{B}$ obtenu dans le cas de la détection est remplacé par une moyenne pondérée des matrices de covariance des bruits $\frac{1}{K} \sum_{i=1}^K \rho_i \mathbf{B}_i$.

Les coefficients de pondération sont les inconnues de notre problème.

➤ Moyenne arithmétique

La moyenne arithmétique des gains est donnée par l'expression :

$$\rho = \frac{1}{K} \sum_{i=1}^K \rho_i$$

Tous calculs faits, on montre que la solution \mathbf{X} doit vérifier (cf. annexe 1) :

$$\left(\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \frac{\rho_i^2}{\rho} \mathbf{B}_i \right) \mathbf{X} = \mathbf{X} \boldsymbol{\Omega}_0, \quad (\text{I.14})$$

expression qui fait intervenir la moyenne arithmétique des \mathbf{B}_i pondérées par des poids $\alpha_i = \frac{\rho_i^2}{\rho}$.

La forme de cette expression est donc identique à celle trouvée avec le critère géométrique sauf qu'on avait $\alpha_i = \rho_i$.

A chaque itération de l'algorithme, on doit calculer les nouveaux termes α_i pour pouvoir calculer la matrice $\mathbf{M}(\rho_1, \dots, \rho_K)$.

➤ Moyenne harmonique

En ce qui concerne la moyenne harmonique, l'expression du gain ρ est donnée par :

$$\frac{1}{\rho} = \frac{1}{K} \sum_{i=1}^K \frac{1}{\rho_i}$$

On montre que la solution \mathbf{X} de cette équation doit vérifier : [annexe 1]

$$\left(\mathbf{A} - \frac{\rho}{K} \sum_{i=1}^K \mathbf{B}_i \right) \mathbf{X} = \mathbf{X} \mathbf{\Omega}_0 \quad (\text{I.15})$$

Cette expression fait intervenir la moyenne arithmétique des \mathbf{B}_i sans pondération. Cela veut dire que dans ce cas-là on est ramené à étudier un seul processus perturbateur qui représente la moyenne de tous les processus perturbateurs. C'est habituellement, ce qui est fait dans la littérature [5].

Généralisation

On peut donner une forme générale de l'équation à résoudre dont le coefficient α_i est à adapter en fonction des critères.

$$\left(\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \alpha_i \mathbf{B}_i \right) \mathbf{X} = \mathbf{X} \mathbf{\Omega}_0$$

A chaque critère correspond des expressions précises de α_i .

- moyenne géométrique : $\alpha_i = \rho_i$.
- moyenne arithmétique : $\alpha_i = \frac{\rho_i^2}{\rho}$.
- moyenne harmonique : $\alpha_i = \rho$.

De même que pour la détection, il existe un algorithme qui permet d'atteindre la solution optimale [5]. Il est basé sur le même principe mais il est un peu compliqué puisque l'on cherche, non pas une valeur de ρ , mais K valeurs ρ_i pour en déduire ρ .

I.4 Conclusion

Dans ce premier chapitre, on s'est intéressé à la détection et à la classification de signaux aléatoires en utilisant le Filtrage Adapté Stochastique sous Contraintes (FASC) ou *Constrained Stochastic Matched Filter* (CSMF) qui repose sur deux principes :

- la projection des signaux dans un espace de dimension fixée *a priori* qu'on note p ,
- la maximisation du rapport signal à bruit RSB, quantité qui s'écrit naturellement en fonction des matrices de covariance des signaux traités.

L'idée du CSMF est de maximiser le RSB après filtrage ou projection dans un sous-espace qu'on appelle « sous-espace optimal ».

Le CSMF peut être appliqué dans deux situations que l'on appelle ici désormais :

- Détection : quand il y a un signal d'intérêt et un bruit.
- Classification : quand on est en présence d'un signal d'intérêt et de plusieurs perturbations potentiellement présentes additivement, chacune pouvant devenir le signal d'intérêt par permutation.

Le signal d'intérêt s est décorrélé de chacun des bruits aléatoires possibles, eux-mêmes décorrélés deux à deux. Dans le cas de la détection, le critère à optimiser est unique tandis que dans celui de la classification, on a envisagé plusieurs critères à optimiser : on en a présenté trois qui sont définis par les moyennes arithmétique, géométrique et harmonique.

Chaque critère conduit à un résultat différent et il est parfois difficile de comparer entre ces résultats (problème de courbes COR qui se croisent) : ceci sera illustré dans le chapitre suivant.

Chapitre II :

Le filtre multicritère

II.1 Introduction

Pour tirer profit de la performance de chacun des critères, on introduit la notion de fusion des critères qui consiste à prendre en considération plusieurs critères en même temps lors du filtrage des signaux.

Dans notre cas, on se propose de fusionner les trois critères déjà énoncés : arithmétique, géométrique et harmonique.

Cette technique de filtrage va nous permettre d'obtenir les extensions multicritères du CSMF.

Le but de cette partie, est de comparer les résultats de filtrage obtenus ainsi que les différentes courbes COR après application de ce filtre multicritère sur le signal test.

Afin d'obtenir un filtre issu de plusieurs critères, on suit la procédure suivante : tout d'abord, on calcule les filtres du signal test avec chacun des p vecteurs définissant l'espace optimal relatif à chacun des critères : on obtient donc $\mathbf{X}_{p1}, \mathbf{X}_{p2}$ et $\mathbf{X}_{p3} \in M_{n \times p}$.

le filtrage réalisé par \mathbf{X}_{p1} conduit à un vecteur dont la norme au carré représente une variable aléatoire d'espérance mathématique $\beta_1 = \text{tr}(\mathbf{X}_{p1}^T \mathbf{A} \mathbf{X}_{p1})$, de même pour les deux autres critères.

On propose donc de normaliser le résultat par racine de β_1 .

Ainsi, on décide donc de normaliser avant concaténation :

$$\mathbf{X}_{pm} = \begin{bmatrix} \frac{\mathbf{X}_{p1}}{\sqrt{\beta_1}} & \frac{\mathbf{X}_{p2}}{\sqrt{\beta_2}} & \frac{\mathbf{X}_{p3}}{\sqrt{\beta_3}} \end{bmatrix} \in M_{n \times (3p)}$$

avec $\beta_i = \text{tr}(\mathbf{X}_{pi}^T \mathbf{A} \mathbf{X}_{pi})$, $i = 1, 2, 3$, la puissance du signal à détecter projeté dans le sous espace optimal engendré par \mathbf{X}_{pi} .

\mathbf{X}_{pm} sera donc associée au filtrage multicritère.

Après filtrage par \mathbf{X}_{pm} , on obtient un signal \mathbf{s}' de puissance :

$$\mathbb{E}(\mathbf{s}'^T \mathbf{s}') = \text{tr}(\mathbf{A}') = \text{tr}(\mathbf{X}_{pm}^T \mathbf{A} \mathbf{X}_{pm})$$

\mathbf{A} étant la matrice de covariance du signal avant filtrage, \mathbf{A}' celle du signal après filtrage et \mathbf{X}_{pm} le sous-espace optimal utilisé pour le filtrage.

L'étude du résultat du filtrage des signaux par les courbes COR nous permettra de juger l'efficacité de ce type de filtrage.

II.2 Filtrage multicritère appliqué à des signaux simulés

II.2.1 Signaux utilisés et estimation des matrices

Création des signaux :

Nous avons créé les signaux $s_i, i = 1..4$ par filtrage en bande de fréquence de processus blancs indépendants [annexe 2].

Estimation des matrices :

Les matrices de covariance qui correspondent à chaque signal s_i ont été estimées sur la totalité des points en utilisant une fenêtre glissante sur le signal de taille $K = 4096$.

La puissance des signaux est normalisée à un en entrée du filtre.

Création du signal à filtrer : Signal test

On construit un signal test par juxtaposition et/ou par superposition des réalisations des signaux s_i . Ce nouveau signal est représenté par la barre inférieure sur la figure II.1, les numéros précisant les signaux présents dans chaque intervalle de temps.

Nous avons construit 11 zones différentes contenant chacune un certain nombre de points [annexe 2].

Par la suite, nous détectons les occurrences du signal « s_4 » dans le signal test construit en appliquant l’algorithme CSMF avec les matrices estimées.

On choisit $n = 15$ et $p = 5$ pour cette simulation.

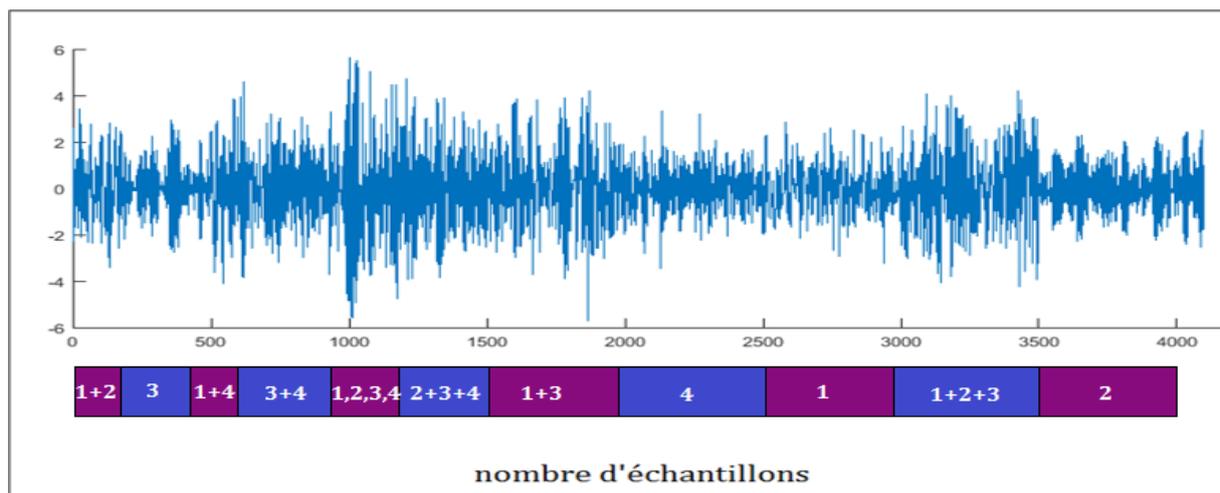


Figure II.1. Signal test à filtrer

Le filtre multicritère

II.2 Filtrage multicritère appliqué à des signaux simulés

Filtrage du signal

Le signal filtré est obtenu par projection du signal original (signal test) dans le sous-espace engendré par X_{pi} dans le cas d'un filtrage à critère unique ou par X_{pm} dans le cas d'un filtrage multicritère.

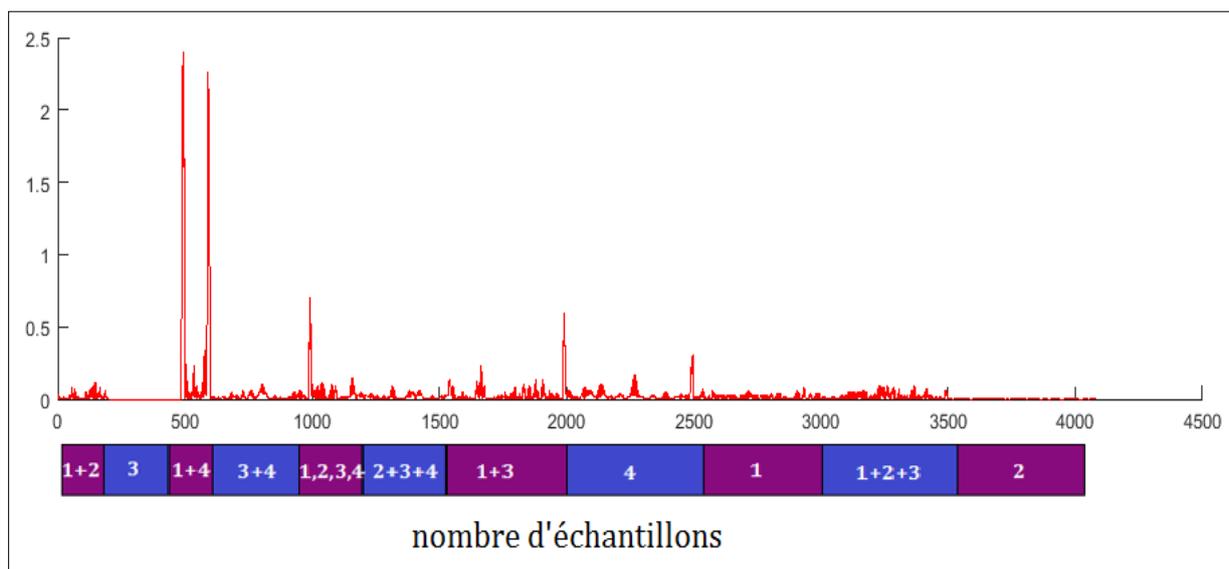
Ainsi on fait une fenêtre glissante en se décalant d'un seul point à chaque itération, on projette le signal dans le sous-espace optimal et enfin on trace le module au carré du résultat (signal filtré).

II.2.2 Résultats de classification

Dans un premier temps, on va filtrer ce signal en utilisant un seul critère : on choisit le critère arithmétique par exemple, puis on va utiliser le filtre multicritère.

Filtrage en utilisant le critère arithmétique

La puissance du signal projeté dans le sous-espace optimal correspondant au critère arithmétique donne le résultat présenté sur la figure II.2.



Filtrage en utilisant le filtre multicritère

La puissance après filtrage du même signal (présenté sur la figure II.1), en utilisant la matrice X_{pm} , donne le résultat présenté sur la figure II.3.

Le filtre multicritère

II.2 Filtrage multicritère appliqué à des signaux simulés

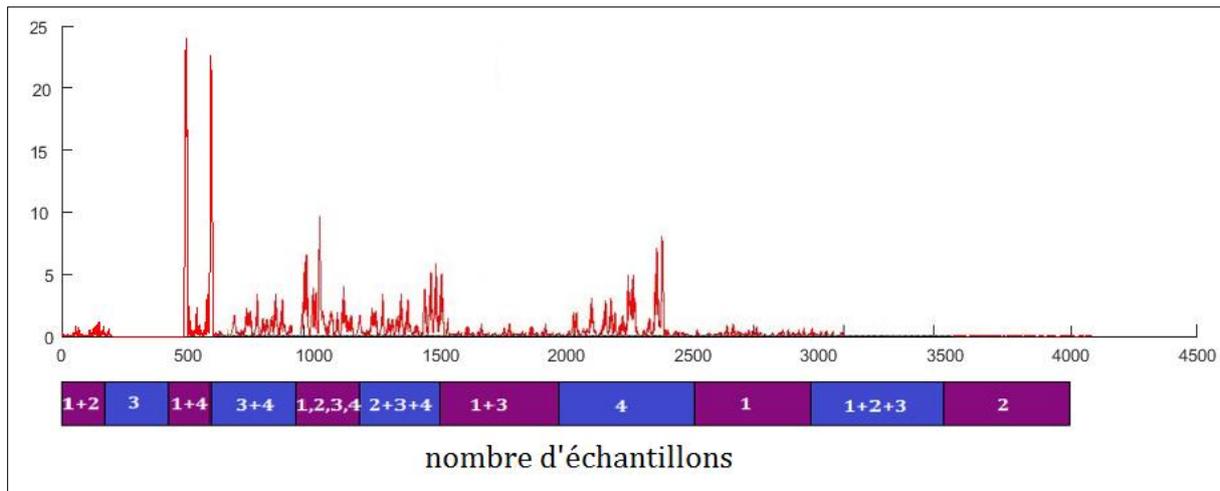


Figure II.3. Filtrage multicritère

D'après les résultats obtenus, la détection du signal « s_4 » semble meilleure dans le deuxième cas. Nous confirmerons ceci par l'étude des courbes COR.

II.2.3 Etude des courbes COR

La fonction d'efficacité du récepteur, plus fréquemment désignée sous le terme « courbe COR » pour « Caractéristiques opérationnelles du récepteur » a pour objectif de catégoriser des éléments en deux groupes distincts sur la base d'une ou de plusieurs caractéristiques de chacun de ces éléments [39].

Pour notre problème de classification, ces courbes vont nous servir d'outil d'évaluation et de comparaison des différents résultats de filtrages obtenus en changeant à chaque fois le signal à détecter ainsi que le critère (moyenne des gains élémentaires) d'exécution de l'algorithme.

Les bonnes courbes COR doivent se situer dans la partie supérieure gauche de l'espace des COR, c'est-à-dire présentant un meilleur compromis que la première bissectrice. En effet, celle-ci représente un état de détection aléatoire : $P_D = P_{FA}$.

Le point optimal (0,1) n'est jamais atteint, cela signifie qu'il n'existe pas de seuil pour lequel toutes les détections possibles sont réalisées sans aucune fausse alarme.

- **Détection du signal s_4 :**

Désormais, on trace les différentes courbes COR correspondant aux signaux obtenus après filtrage arithmétique, géométrique, harmonique puis multicritère.

Pour le signal s_4 , utilisé dans l'exemple précédent on obtient les courbes COR présentées par la figure II.4.

Il faut noter ici que les deux critères arithmétique et géométrique donnent presque le même résultat (les deux courbes sont quasiment confondues) et que les courbes se croisent avec celle

Le filtre multicritère

II.2 Filtrage multicritère appliqué à des signaux simulés

qui représente le critère harmonique : ceci ne nous permet pas de choisir la meilleure d'entre elles en termes de détection.

Par contre, la courbe associée au filtre multicritère est située au-dessus de toutes les autres courbes et présente donc le meilleur compromis.

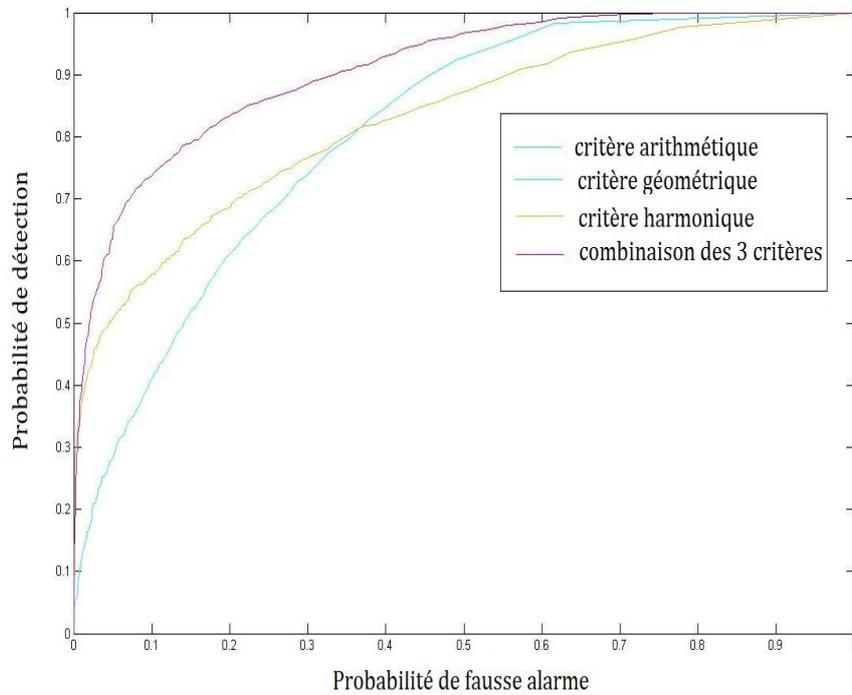


Figure II.4. Courbes COR associées aux différents critères (signal s_4)

Pour les autres signaux on obtient les résultats présentés par les figures II.5, II.6 et II.7.

Le filtre multicritère

II.2 Filtrage multicritère appliqué à des signaux simulés

- Détection du signal s_3 :

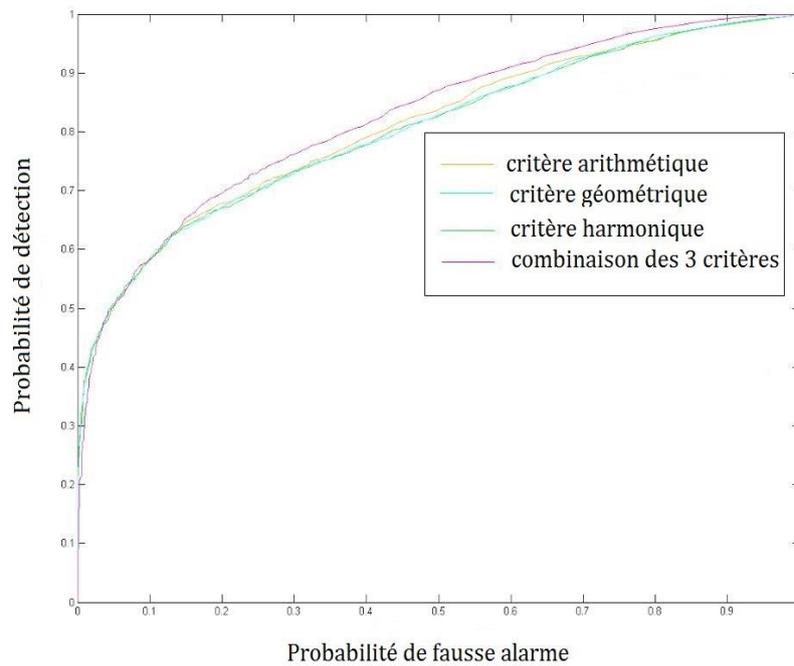


Figure II.5. Courbes COR associées aux différents critères (signal s_3)

- Détection du signal s_2 :

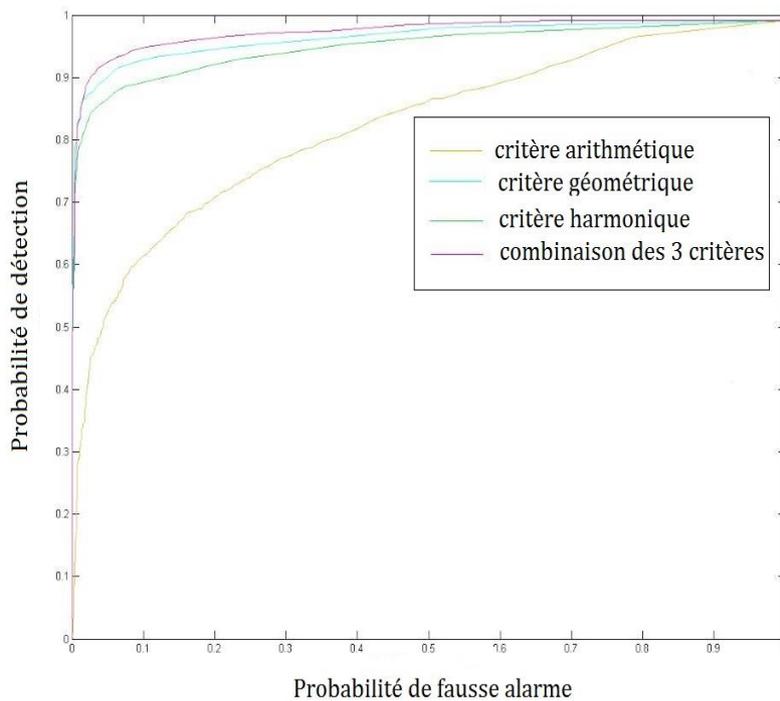


Figure II.6. Courbes COR associées aux différents critères (signal s_2)

Le filtre multicritère

II.2 Filtrage multicritère appliqué à des signaux simulés

- **Détection du signal s_1 :**

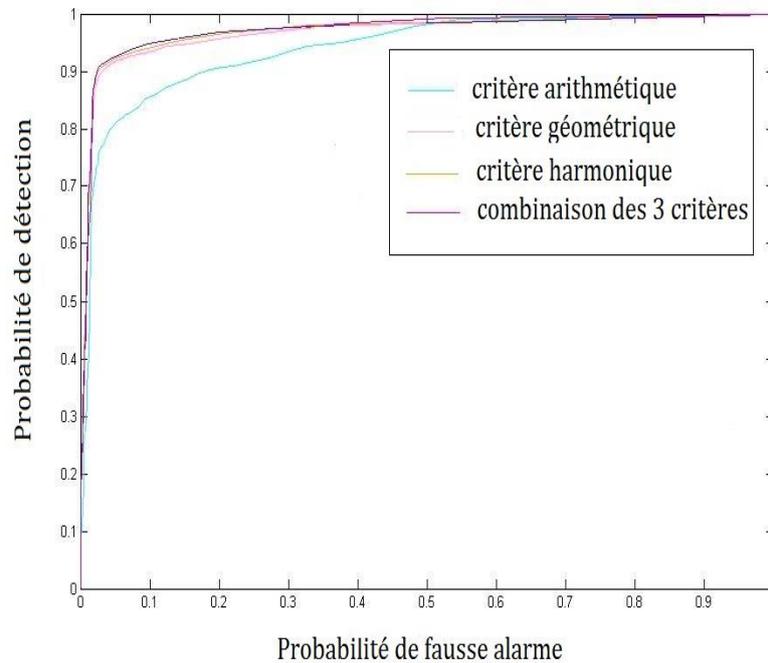


Figure II.7. Courbes COR associées aux différents critères (signal s_1)

L'étude des courbes COR des différents résultats de filtrages obtenus montre que la détection est systématiquement meilleure (ou au moins équivalente à la meilleure) dans le cas où l'on fusionne les trois critères.

Courbes COR : Variation du RSB

Les résultats présentés précédemment sur la figure II.7, ont été obtenus pour un RSB de 1 (avant filtrage). Ici nous proposons de présenter les courbes COR pour différents RSB. Pour cela, nous multiplions le signal d'intérêt par un coefficient variant de 0.2 à 1 par pas de 0.1.

On trouve sur la figure II.8 les résultats de la détection pour ces différents RSB.

On remarque qu'en augmentant la valeur du coefficient multiplicateur du signal d'intérêt α et donc du RSB, les courbes COR se rapprochent du point optimal (0,1) : ceci confirme que le résultat est d'autant meilleur que le RSB en entrée est grand.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

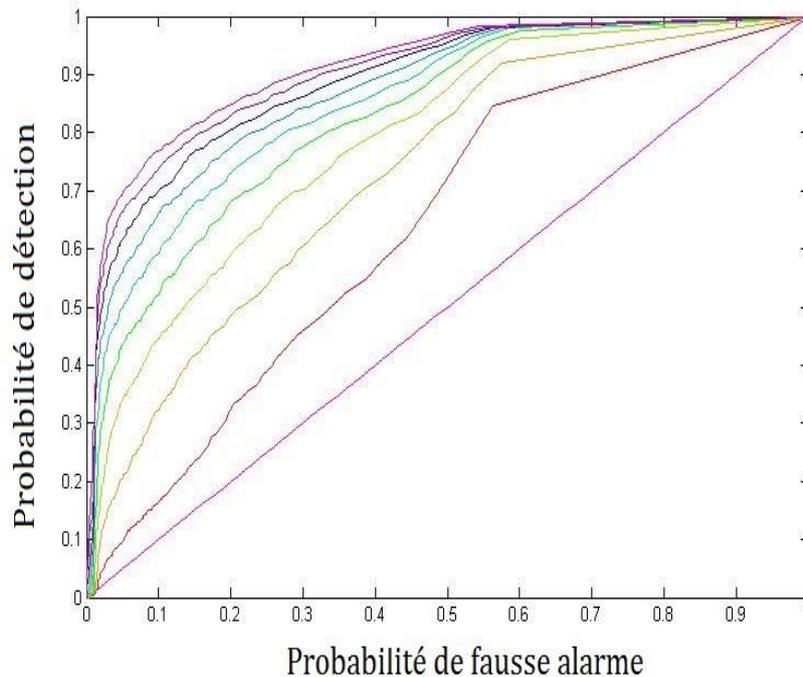


Figure II.8. détection du signal s_1 : résultats des courbes COR pour différents RSB

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

II.3.1 Nature des signaux

Dans cette partie, on travaille sur des signaux réels d'acoustique sous-marine : les clics des cétacés.

Les cétacés produisent des émissions sonores dans une très large bande de fréquence, entre 10 Hertz et 150 kHz environ. Mais seule la partie inférieure de ce spectre -en deçà de 20 kHz- nous est audible. Les infrasons ne sont pas audibles mais sont très bien captés par les instrumentations militaires ou industrielles. Ces vocalisations sont produites par des circulations d'air dans le système respiratoire supérieur des animaux, entre deux poches, par des résonances, ou le pincement d'écoulement à travers un diaphragme. A la différence de l'homme, la production d'un son n'est pas généralement associée à un échappement d'air, en raison de l'adaptation nécessaire à l'apnée.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

Les sons produits par le dauphin, peuvent être de type impulsif (clics, tics, bourdon ...) ou continu (sifflements, cris, mugissement) ou une succession des deux. Seul le Cachalot n'émet qu'une seule catégorie de son : des clics. [40]

Pour cette raison précise, on s'intéresse à la détection des clics pour les deux espèces : des clics de dauphins (Stenelle) et de cachalots (Physeter macrocephalus) se présentent comme suit :

- Des clics de dauphins.
- Des clics de cachalots vus de face par rapport à la position de l'hydrophone.
- Des clics de cachalots vus de dos par rapport à la position de l'hydrophone.

Le tout étant noyé dans du bruit de mer.

Ces signaux nous ont été fournis par l'équipe du laboratoire LSIS de l'université de Toulon, ils ont été réalisés en juillet 2014 dans le cadre d'un projet collaboratif entre les laboratoires du CNRS LSIS et MIO qui consistait à enregistrer les activités des baleines au sud du parc national de Port-Cros [41].

Un extrait de cet enregistrement sur deux jours, nous permet d'obtenir des réalisations différentes qui correspondent à chaque type de signal.

Le problème, est que le nombre des signaux renseignés est faible : on n'est pas capable alors d'utiliser la totalité des signaux de la base de données.

Notre objectif est donc de détecter soit les dauphins soit les cachalots (vus de face ou de dos) sur les parties renseignées de l'enregistrement.

On applique alors le CSMF sur les échantillons après normalisation.

Signaux réels utilisés

Quelques réalisations parmi celles obtenues, sont représentées par les figures II.9, II.10, II.11 et II.12.

Dans cette partie, on se propose d'utiliser des signaux sur une longueur $L=150$ points : c'est le nombre de points minimum qui nous permet d'observer le phénomène à détecter (les clics s'étendent sur une dizaine de kHz).

La fréquence d'échantillonnage des signaux est $F_s = 44 \text{ kHz}$, le nombre de points est en réalité le temps (chaque réalisation dure 3 millisecondes).

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

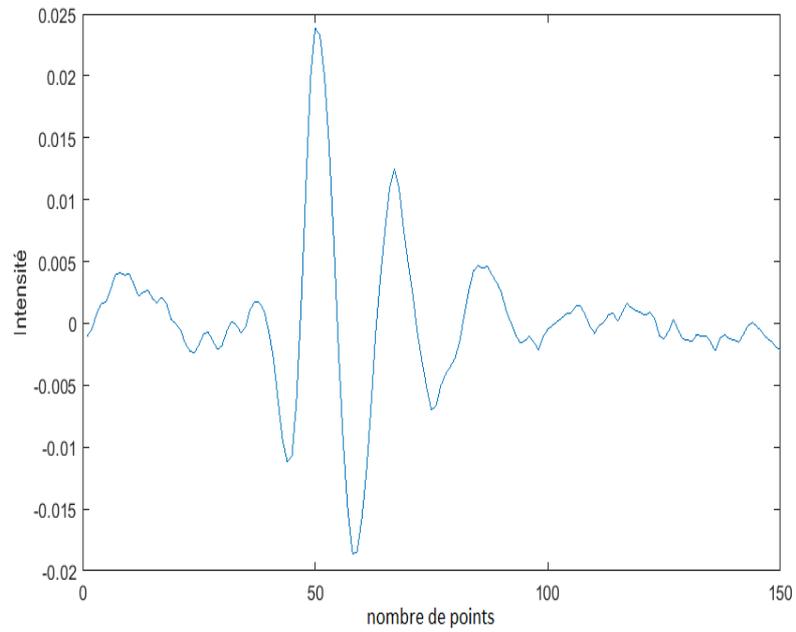


Figure II.9. clic de dauphin

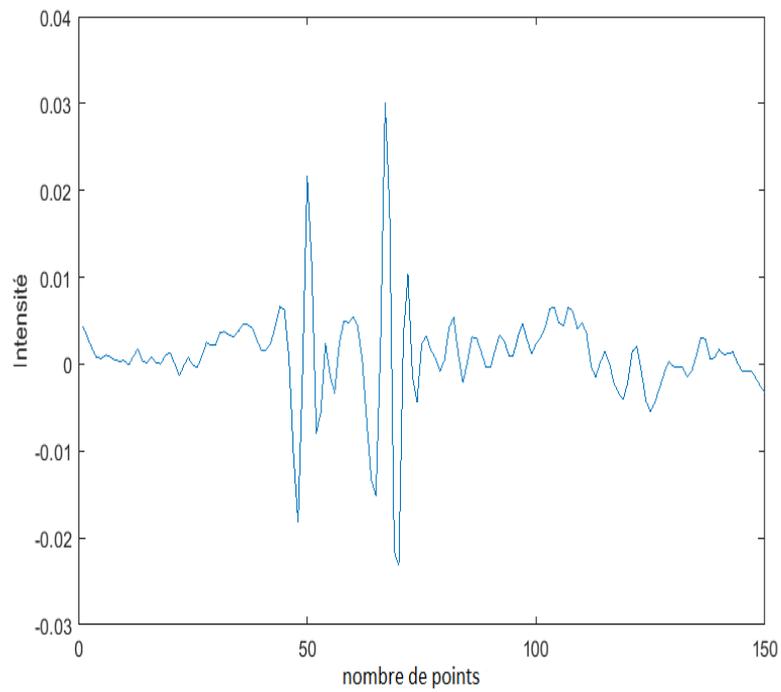


Figure II.10. clic de cachalot (face)

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

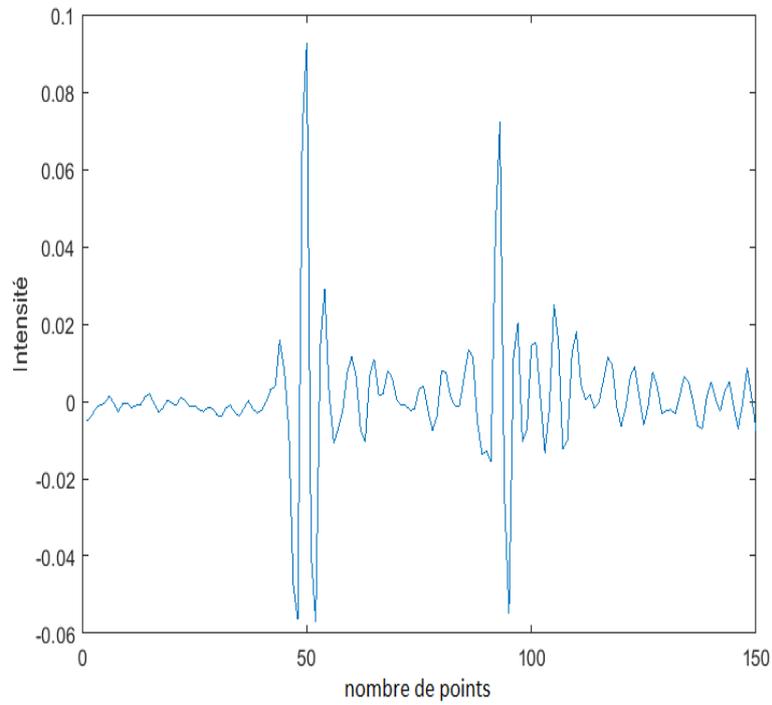


Figure II.11. clics de cachalot (dos)

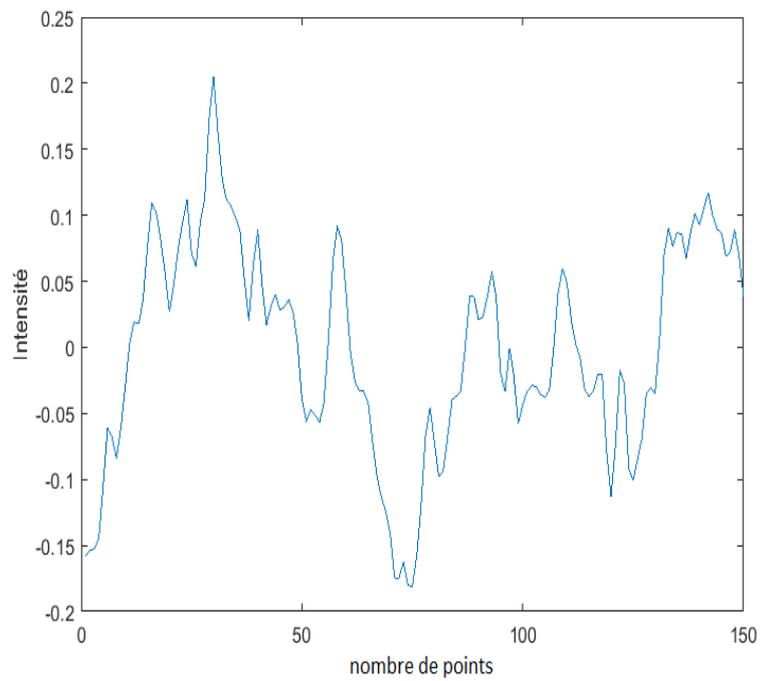


Figure II.12. Bruit de mer

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

II.3.2 Estimation des matrices de covariance

On dispose de clics de dauphins, de cachalots vus de dos et de cachalots vus de face sur des intervalles de $L = 150$ points par réalisation.

Vu que les signaux ne sont pas bien renseignés, on n'utilisera que 15 réalisations par type de signal dont 10 pour l'apprentissage.

Pour un processus donné \mathbf{s}_i , si on dispose de $R = 10$ réalisations, la matrice de covariance est estimée par :

$$\mathbb{E}(\mathbf{s}_i \mathbf{s}_i^T) = \frac{1}{R} \sum_{k=1}^R \mathbf{s}_{i,k} \mathbf{s}_{i,k}^T$$

où $\mathbf{s}_{i,k}$ représente la $k^{\text{ème}}$ réalisation de \mathbf{s}_i .

On remarque que les matrices obtenues de taille $(L \times L)$ sont de rang incomplet égal à R : ceci veut dire que l'information tirée de ces signaux peut être trouvée dans l'espace image de dimension R .

On peut retrouver cet espace image, en sélectionnant les R vecteurs associés aux plus grandes valeurs propres de la matrice de covariance estimée.

La projection des matrices initiales dans le sous espace image retrouvé nous permet d'obtenir des matrices de dimensions $(R \times R)$: ce sont ces matrices qu'on utilise par la suite pour le filtrage. On prend alors $n = 10$ et $p = 5$.

II.3.3 Détection des signaux

- **Détection d'un dauphin par rapport à un cachalot vu de face**

On extrait de l'enregistrement initial, un morceau (2000 points) dans lequel un clic de dauphins est présent et qui n'a pas servi dans la phase d'apprentissage, et on fait de même pour le clic de cachalot.

Le clic de dauphin est représenté sur la figure II.13 et le clic de cachalot sur la figure II.14.

Nous appliquons à ces deux signaux le filtre CSMF (calculé à partir des estimations des matrices de covariances précédentes afin de détecter le clic de dauphin) après avoir projeté dans le sous-espace image. Globalement, ceci revient à filtrer chaque réalisation de longueur L avec p vecteurs de même longueur. Le filtrage est réalisé par fenêtre glissante.

Nous obtenons les résultats présentés par les parties inférieures des figures II.13 et II.14.

On constate que le signal dauphin est bien détecté et que le signal cachalot n'a pas été interprété comme un signal dauphin.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

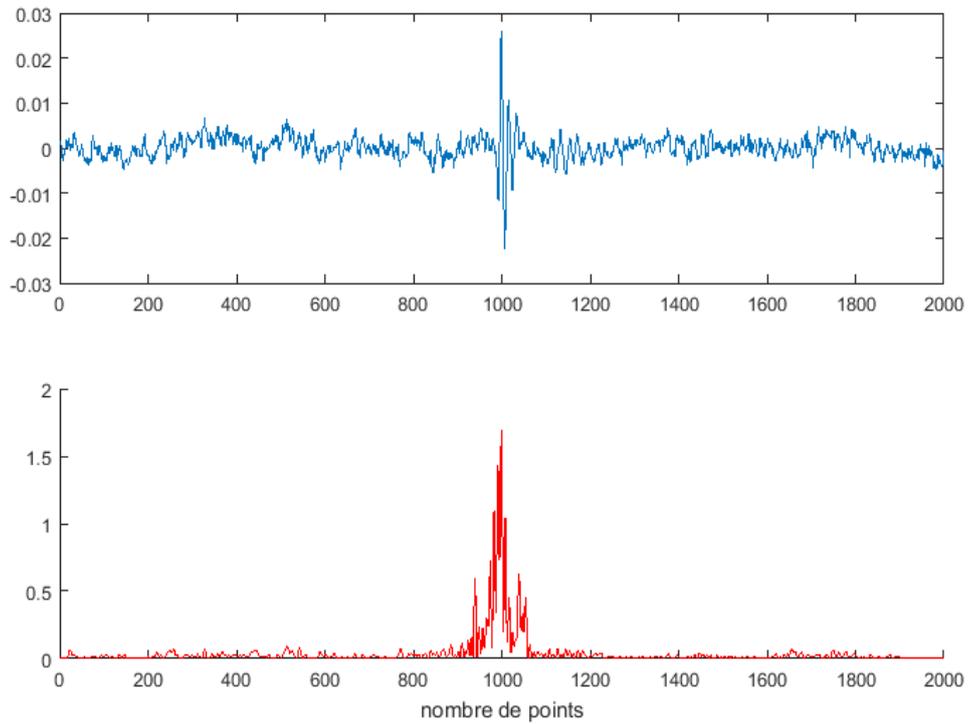


Figure II.13. Détection d'un dauphin dans un clic de dauphin à l'aide du CSMF

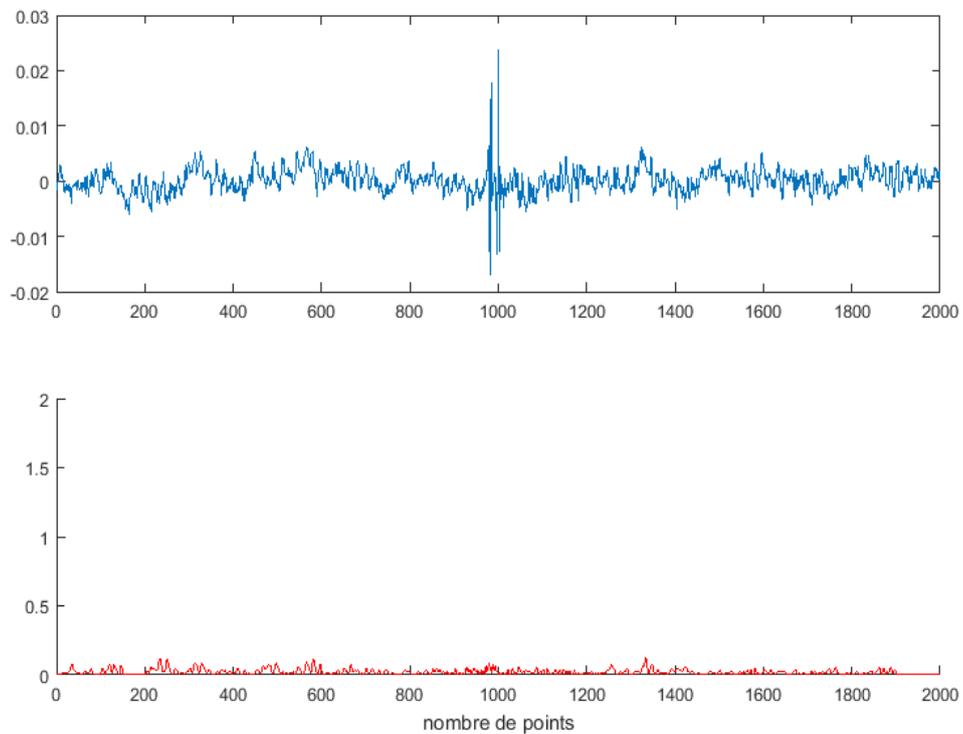


Figure II.14. Détection d'un dauphin dans un clic de cachalot à l'aide du CSMF

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

- **Détection d'un cachalot vu de face par rapport à un cachalot vu de dos**

On reprend le même signal du cachalot vu de face, et on sélectionne une réalisation d'un clic de cachalot vu de dos.

Afin de détecter le clic de cachalot vu de face, nous appliquons le CSMF par fenêtre glissante. On obtient les résultats des figures II.15 et II.16.

Sur les parties inférieures des figures II.15 et II.16, on constate que le signal d'intérêt est plus facile à détecter.

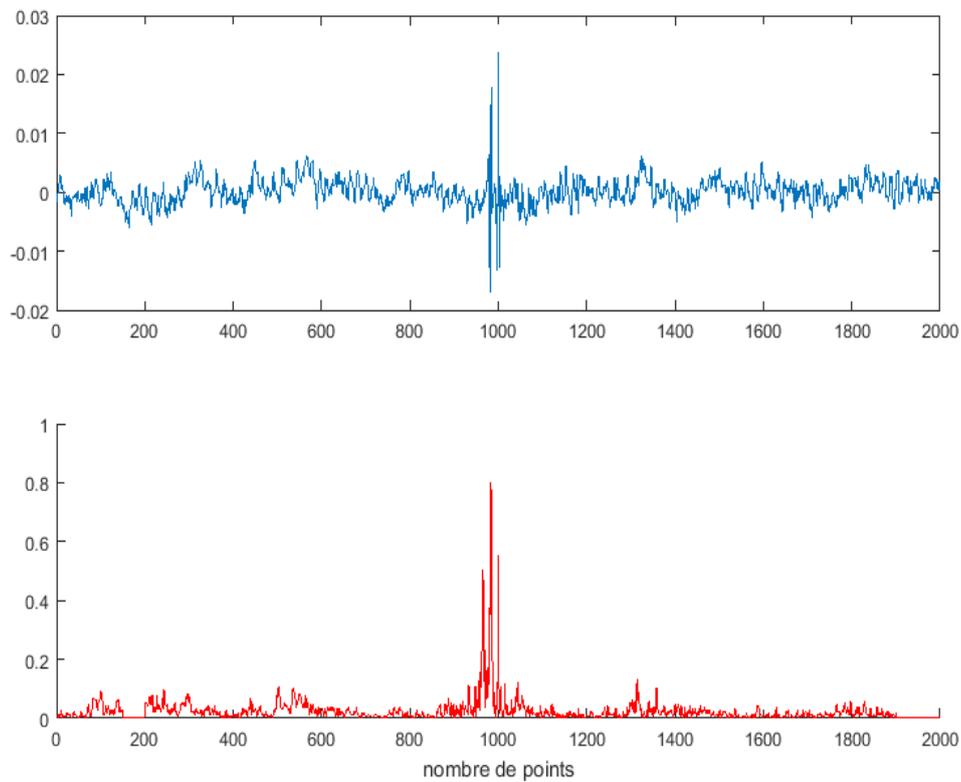


Figure II.15. Détection d'un cachalot vu de face dans un clic de cachalot vu de face à l'aide du CSMF

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

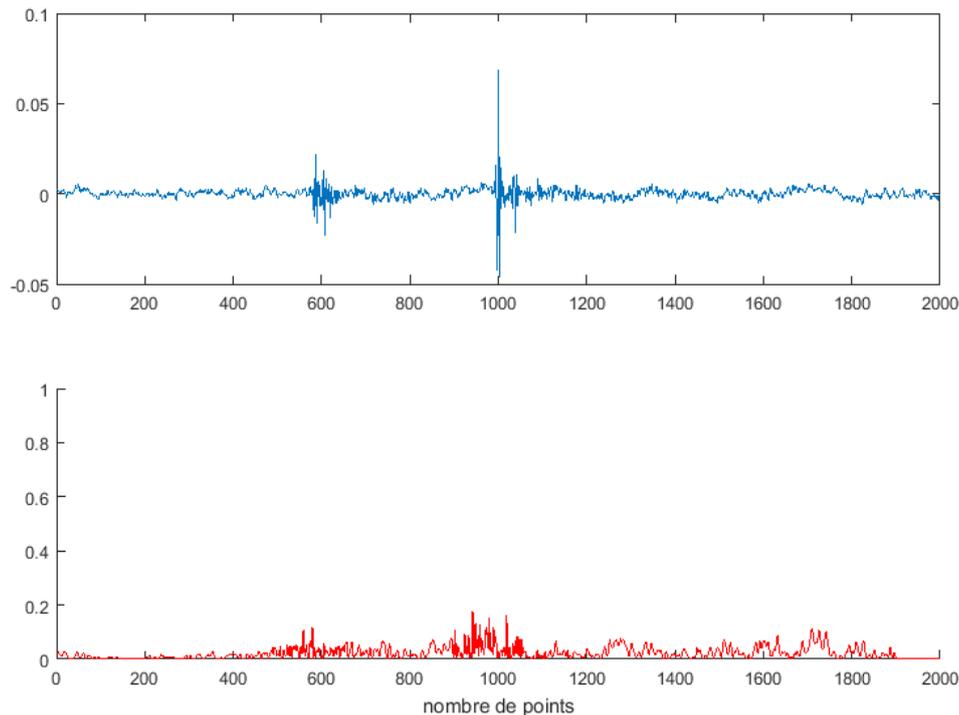


Figure II.16. Détection d'un cachalot vu de face dans un clic de cachalot vu de dos à l'aide du CSMF

On constate que le pic sur la figure II.16 n'a pas été détecté : le clic émis par le cachalot vu de dos n'a pas été interprété comme un clic émis par un cachalot vu de face.

II.3.4 Classification de signaux d'acoustique sous-marine

Afin de juger les performances du filtrage multicritère dans le cas où nous avons plusieurs bruits possibles, on construit un signal à l'aide d'échantillons de clics de dauphins, de cachalots vus de dos et de cachalots vus de face en faisant des mélanges de différents types de clics sur des intervalles de 150 points pour chaque réalisation.

Ainsi, le signal test obtenu de longueur $L' = 7200$ points, contient en tout 48 réalisations différentes mises bout à bout.

On alterne les signaux contenant des clics de dauphins et des signaux qui n'en contiennent pas comme suit :

Zone 1 : les 8 premières réalisations contiennent des clics de dauphins mélangées ou non avec des signaux de cachalots.

Zone 2 : les 8 réalisations suivantes, représentent uniquement des signaux de cachalots.

Zone 3 : les 8 réalisations suivantes, contiennent des clics de dauphins.

Zone 4 : les 8 réalisations qui suivent sont des clics de cachalot.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

Zone 5 : les 8 réalisations suivantes contiennent des clics de dauphins.

Zone 6 : les 8 dernières réalisations sont des clics de cachalot et du bruit de mer.

Le bruit de mer est présent dans toutes les zones.

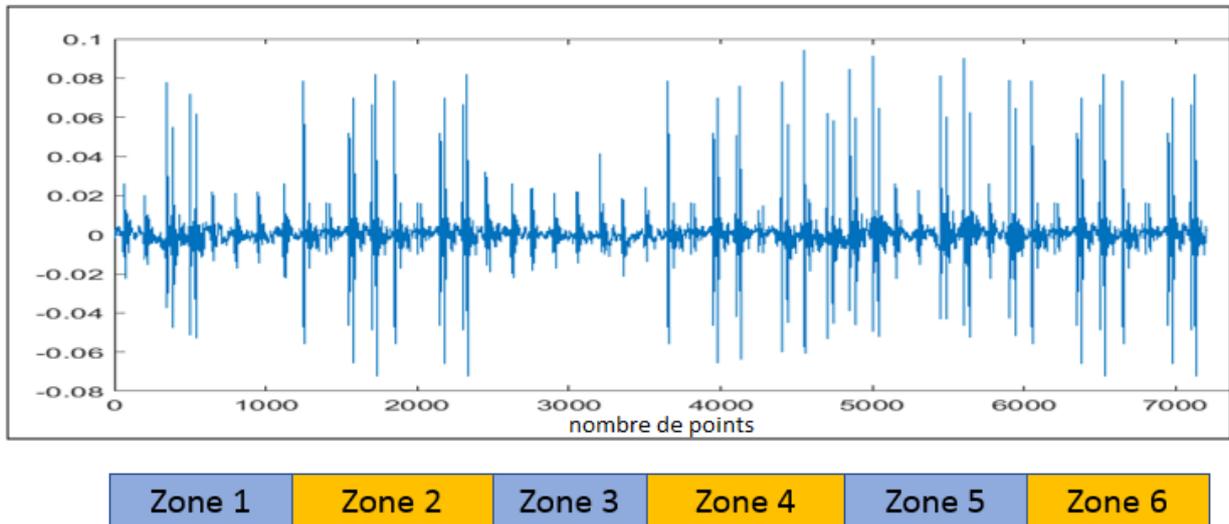


Figure II.17. signal test construit à partir de différents échantillons

Le but est de détecter uniquement les clics de dauphin qui sont présents ici dans les zones (en bleu dans la barre) 1,3 et 5 du signal test.

Détection des clics de dauphins ($p = 5$) :

Dans un premier temps, on s'intéresse à la détection de dauphins dans ce mélange de signaux. On a alors :

Signal : Dauphin

Bruits : Cachalot vu de dos, cachalot vu de face, bruit de mer.

On choisit $n = 10$ et $p = 5$: n est la dimension des signaux.

On applique le CSMF en utilisant les critères arithmétique, géométrique et harmonique puis on utilise le CSMF multicritère.

Le résultat du filtrage du signal test en considérant le critère harmonique est représenté sur la figure II.18.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

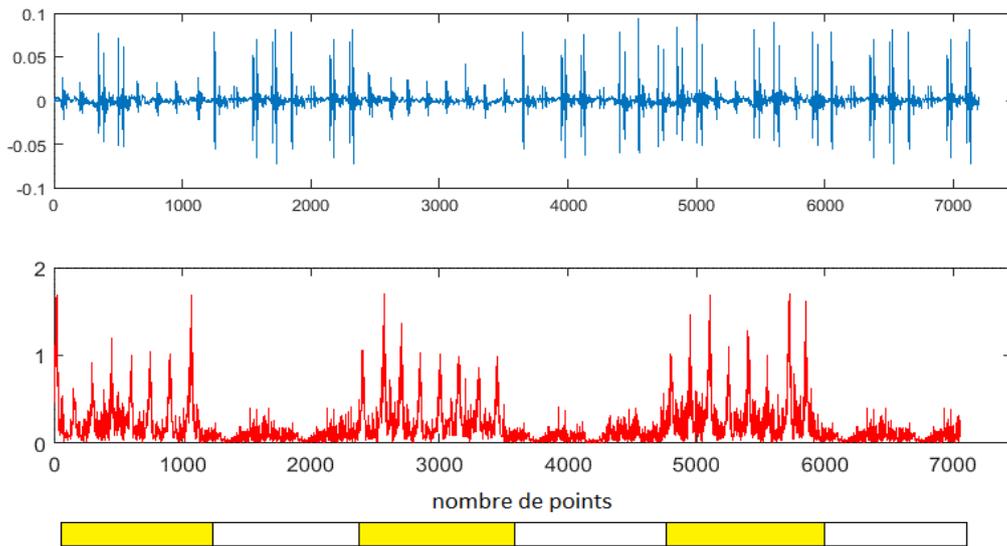


Figure II.18. résultat du filtrage du signal test en considérant le critère harmonique ($p = 5$)

On remarque que sur la Figure II.18, l'amplitude de la filtrée du signal test obtenue est beaucoup plus importante dans les zones délimitées par une barre de couleur jaune.

Ces zones-là correspondent aux échantillons du signal test dans lesquels on détecte la présence des clics de dauphins.

De la même manière, on présente le résultat du filtre multicritère sur la figure II.19

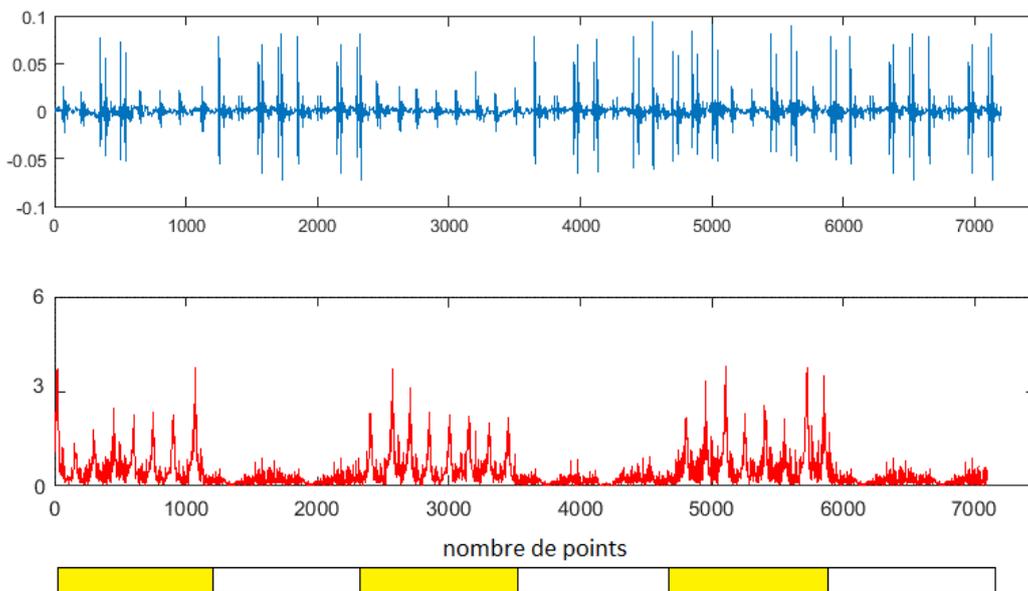


Figure II.19. résultat du filtrage du signal test en considérant le filtre multicritère ($p = 5$)

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

On remarque, en se basant sur ces exemples, qu'il est difficile de juger la performance du filtre à l'œil. On décide alors de tracer les courbes COR correspondantes.

Etude des courbes COR

Afin de mieux juger la performance du filtrage utilisant les différents critères déjà cités, on se propose de tracer les courbes COR correspondantes à chacun des critères employés.

Les courbes de la figure II.20 montrent que le filtrage multicritère présente une meilleure courbe COR que les critères utilisés chacun séparément.

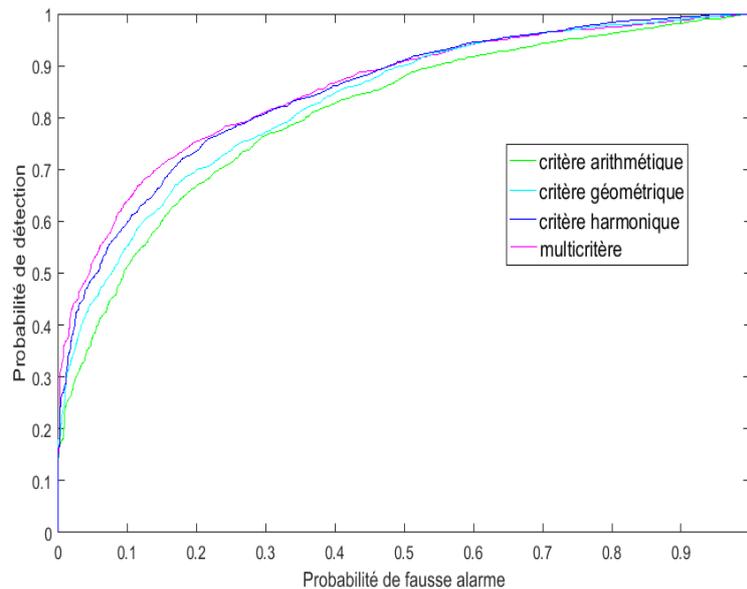


Figure II.20. courbe COR : résultat du filtrage du signal test en considérant le filtre multicritère

Importance de l'indication de tous les bruits :

Dans ce qui suit, on se propose de ne renseigner qu'un seul bruit à l'algorithme (le bruit de mer) tout en filtrant le signal test précédent qui contient plusieurs bruits (bruit de mer, cachalot vu de face, cachalot vu de dos). On cherche toujours à détecter la présence des clics de dauphins et on garde les mêmes valeurs de n et p .

Ceci revient à utiliser la version de l'algorithme CSMF conçue pour le mode détection (un seul bruit) sur un signal multi bruits.

Sur la figure II.21, on rajoute la nouvelle courbe COR obtenue aux courbes de la figure II.20.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

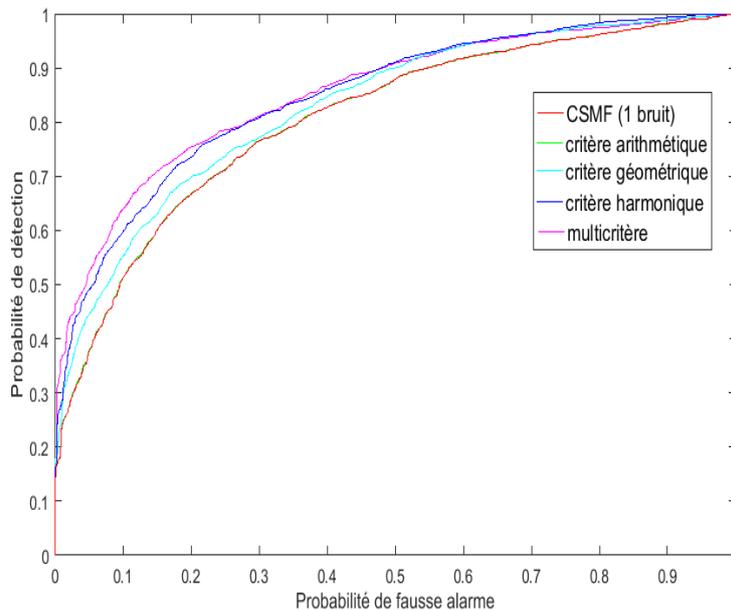


Figure II.21. courbe COR : résultat du filtrage par CSMF en mode détection

La nouvelle courbe COR obtenue (courbe rouge), est située au-dessous de toutes les autres courbes : elle présente donc une détection moins bonne.

Donc l'information apportée par la connaissance d'autres processus indésirables dans le filtre améliore nettement les performances.

La détection est meilleure dans le cas où on utilise la version du CSMF conçue pour la classification en spécifiant à l'algorithme les caractéristiques de chaque bruit (par sa matrice de covariance) dans le cas multi bruits.

Détection des clics de dauphins ($p = 2$) :

Pour ce même signal, et sous les mêmes hypothèses (détection du dauphin), on diminue la taille du sous-espace (initialement $p = 5$) à $p = 2$. On présente par la suite quelques résultats de détection.

Le résultat du filtrage du signal test en considérant le critère géométrique est représenté sur la figure II.22 et les courbes COR correspondantes à chacun des critères employés dont le multicritère sont présentées sur la figure II.23.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

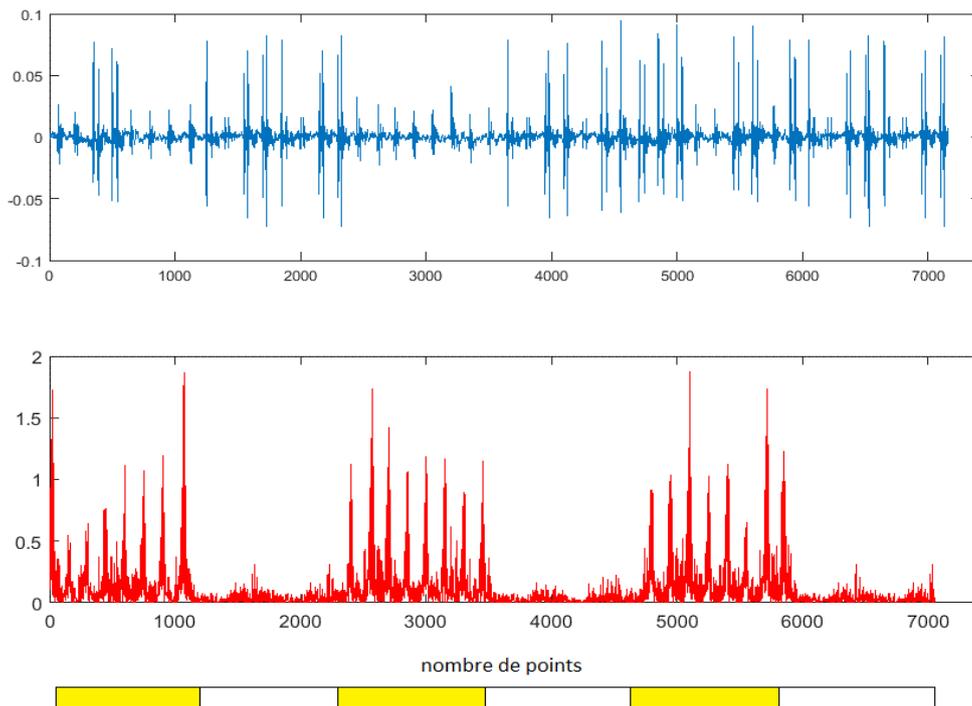


Figure II.22. résultat du filtrage du signal test en considérant le filtre géométrique ($p = 2$)

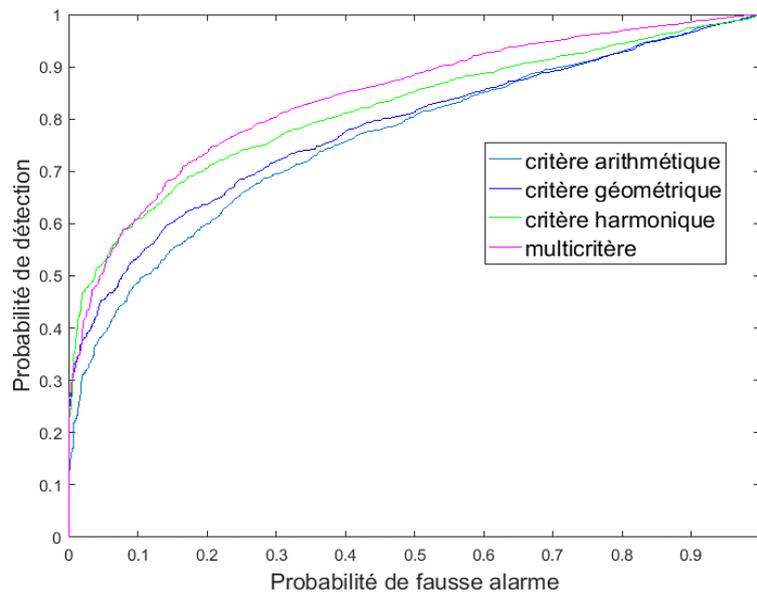


Figure II.23. courbe COR : résultat du filtrage en considérant le filtre multicritère ($p = 2$)

On remarque que la diminution de la valeur de p a modifié légèrement la performance du filtrage multicritère. C'est pour cette raison, qu'on se propose de tracer les courbes COR associés au filtrage multicritère à la recherche de la valeur optimale de p .

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

Détection des clics de dauphins (recherche du p optimal)

La figure II.24 présente les courbes COR obtenues pour différentes valeurs de p .

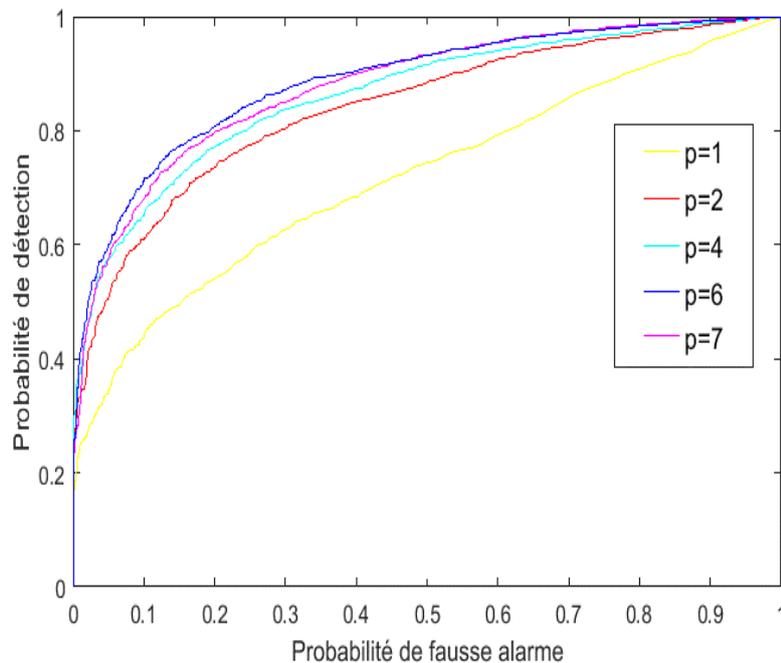


Figure II.24. courbe COR : résultat du filtre multicritère pour différentes valeur de p .

Ici, on n'a pas présenté toutes les courbes COR (pour tous les $p < 10$) pour ne pas encombrer la figure mais on a remarqué que la probabilité de détection augmente avec la valeur de p jusqu'à $p = 6$. Au-delà de cette valeur, cette probabilité diminue.

On en déduit alors, que la valeur optimale de p , dans ce cas particulier, est $p = 6$.

Détection des clics de cachalot ($p = 2$) :

Dans ce qui suit, on ne s'intéresse plus à la détection de clics de dauphins mais plutôt à la détection de cachalots vu de dos par rapport à la position de l'hydrophone. Dans ce cas-là on a :

Signal : Cachalot vu de dos

Bruits : Dauphin, cachalot vu de face, bruit de mer.

On choisit $p = 2$ pour la même valeur de n et on filtre le signal test avec le filtre CSMF.

Les résultats du filtrage multicritère sont donnés par les figures II.25 et Figure II.26.

Le filtre multicritère

II.3 Filtrage multicritère appliqué à des signaux réels d'acoustique sous-marine

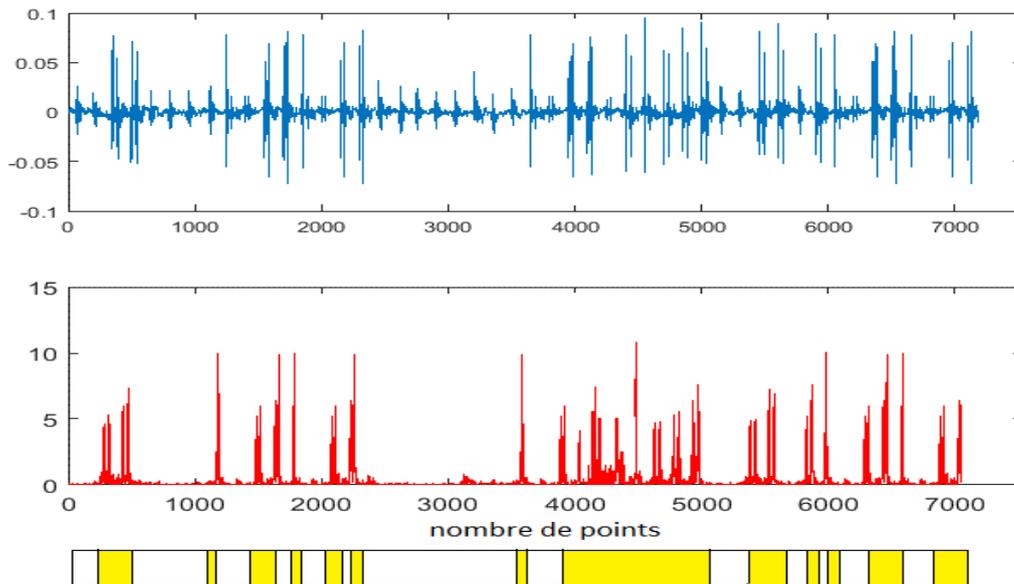


Figure II.25. Détection de cachalots en considérant le filtre multicritère ($p = 2$)

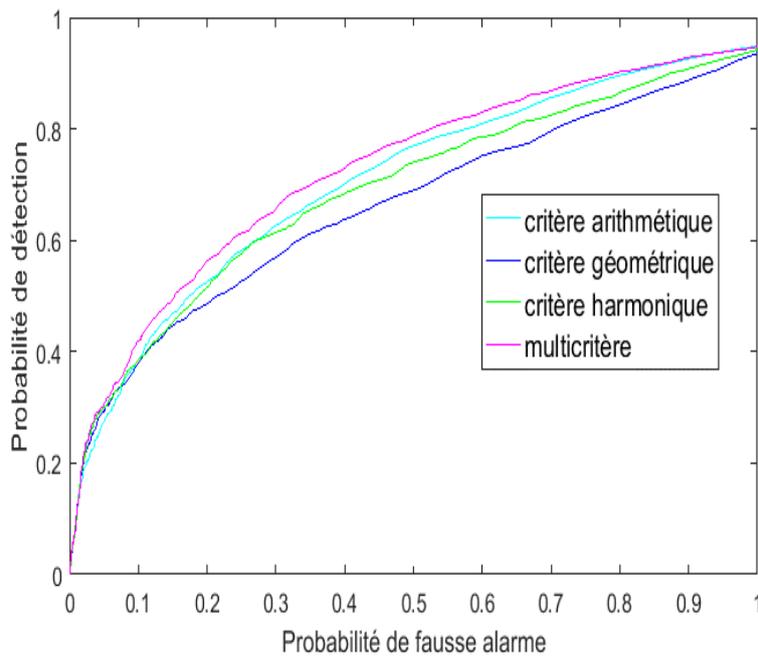


Figure II.26. courbe COR : résultat du filtrage en considérant le filtre multicritère ($p = 2$)

On remarque que les pics du cachalot sont bien détectés (barre jaune) et que la détection avec le filtre multicritère reste meilleure que la détection avec le filtre à critère unique même quand on change la valeur de p .

Détection des clics de cachalots (recherche du p optimal)

On change la valeur de p dans le but de trouver le p optimal pour la détection des signaux de cachalot.

On trace alors les différentes courbes COR données par la figure II.27, mais on remarque qu'à partir de $p = 7$, les courbes sont très proches : difficile alors de fixer une valeur optimale de p et de choisir entre les courbes de $p = 7, 8$ et 9 qui sont indistinguables.

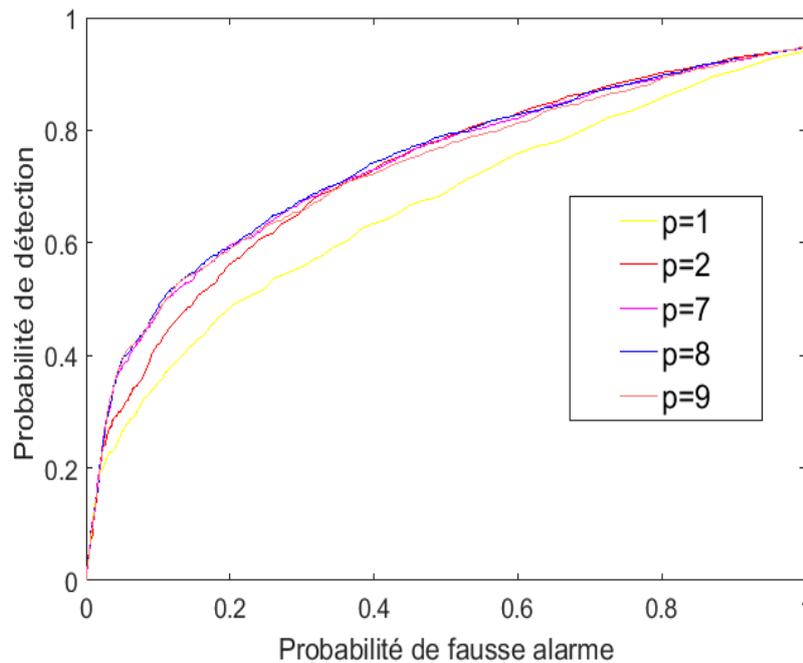


Figure II.27. courbe COR : résultat du filtre multicritère pour différentes valeur de p .

II.4 Conclusion

Dans ce chapitre nous avons traité le problème de classification de signaux en utilisant le filtrage multicritère qui prend en compte simultanément les critères cités précédemment pour le filtrage des signaux.

Cela nous a permis de tester ces extensions multicritères du filtre adapté stochastique sous contrainte sur des signaux simulés. Les résultats obtenus ont montré son efficacité.

Par la suite, nous avons testé cette méthode sur quelques signaux réels issus du monde de l'acoustique sous-marine ; ainsi, nous avons pu détecter des signaux de dauphins et de cachalots. L'étude des courbes COR établit une comparaison entre les différents résultats de filtrage obtenus en changeant à chaque fois le signal à détecter ainsi que les critères d'optimisation.

L'étude des signaux filtrés et des courbes COR ont révélé de meilleurs résultats pour la détection multi bruits en utilisant le filtre multicritère.

Le choix de la meilleure dimension p du sous-espace optimal est réalisée en testant plusieurs valeurs de p et en retenant celle qui donne la meilleure courbe COR.

En effet, il n'y a, à ce jour, pas d'autres méthodes permettant de trouver la valeur optimale de p .

Le problème rencontré dans ce chapitre est le manque de renseignements sur les signaux d'acoustique sous-marine. En effet, la base de données n'est pas toujours riche et ne permet pas de faire plusieurs tests.

Chapitre III :

Poursuite du sous-espace optimal

III .1 Introduction

L'estimation de sous-espaces joue un rôle important dans de nombreuses applications du traitement du signal. Les progrès récents des technologies de l'information ont conduit à un accroissement considérable de la quantité de données à traiter (sons, images, vidéos, textes, etc.). L'extraction de caractéristiques et la compression de données sont des techniques très employées dans l'archivage et la transmission des données en réduisant la quantité d'information.

De nombreuses méthodes ont été proposées pour détecter et extraire automatiquement les éléments caractéristiques composant un signal d'intérêt. La description d'un signal, constitué d'un très grand nombre d'échantillons, en un nombre limité d'éléments distincts en constitue une représentation nécessairement approchée mais fructueuse. De telles décompositions visent à fournir une approximation du signal, exprimée comme une combinaison linéaire d'un nombre réduit d'éléments.

Dans le domaine des statistiques, de nombreux travaux ont été réalisés sur la décomposition en composantes principales et sur la recherche des sous-espaces, en particulier la décomposition de Karhunen-Loève.

On utilise des techniques d'analyse matricielle consistant à calculer une approximation d'une matrice de données, s'exprimant comme un produit de matrices de rang inférieur : on parle d'approximation par réduction de rang. La réduction de rang est en effet une stratégie très utilisée pour extraire les principales caractéristiques d'un ensemble de données. Parmi les outils les plus répandus, l'analyse en composantes principales (ACP) fait appel à la décomposition en valeurs singulières (SVD) [42].

Par la suite, ces méthodes ont été appliquées avec succès au traitement du signal pour des problèmes tels que la compression de données, le filtrage, l'estimation de paramètres (direction d'arrivée DOA, ...), la détection de signaux, l'identification de systèmes ou la reconnaissance de formes : les solutions proposées reposent souvent sur une information de type sous-espace ou éléments propres qui peut être obtenue par une décomposition en éléments propres (EVD) d'une estimation de la matrice de covariance, ou par une décomposition en valeurs singulières (SVD) de la matrice correspondante des données.

L'objectif commun de ces algorithmes est d'estimer le sous-espace principal de dimension p , c'est-à-dire celui qui est généré par les p vecteurs propres associés aux plus grandes valeurs propres de la matrice de covariance du signal.

Pour les signaux dont les propriétés statistiques varient avec le temps, la matrice de covariance nécessite une mise à jour permanente de son estimation. On est alors confronté à un problème de poursuite de sous-espace, ou de tracking.

Cependant, le calcul complet et systématique de ces décompositions devient rapidement prohibitif et incompatible avec le temps réel. Aussi, très naturellement, les chercheurs se sont-ils ingéniés à proposer des techniques donnant une solution certes approchée mais beaucoup moins gourmande en temps de calcul.

Par exemple, Golub s'est intéressé au problème de tracking des SVD et EVD [43], suivi de Schreiber [44], Karasalo [45], DeGroat [46] ...

En 1998, une étude a été réalisée sur les algorithmes adaptatifs pour la poursuite du sous-espace source dans laquelle les différents algorithmes sont classés selon différents critères [47].

Depuis, de nombreux algorithmes ont été proposés pour l'estimation et la poursuite du sous-espace notamment dans le domaine de traitement d'antennes [48] [49] et dans les applications RADAR [50] [51].

Une catégorie particulière de ces algorithmes qui se base sur le principe de projection de sous-espace est nommée algorithme de type *Subspace Projection* (SP), et nous sera utile dans l'implémentation de notre algorithme.

Toutefois, notre objectif est différent puisqu'il s'agit de trouver, en temps réel, le sous-espace optimal en termes de rapport signal à bruit dans un contexte non stationnaire, ce qui induit une variation, à chaque instant de mesure du sous-espace signal et du sous-espace bruit pris séparément. Ceci entraîne, non seulement un changement du sous-espace optimal recherché mais également du rapport signal à bruit.

Une solution serait d'utiliser le CSMF à chaque instant, mais c'est un algorithme récursif lourd en termes de calculs ; pour cette raison, on en propose une version itérative plus rapide et plus légère que l'on appelle *CSMF Subspace Tracking* (CSMFST). L'avantage important de ce nouvel algorithme est qu'il remplace une convergence qui devrait être réalisée à chaque instant de mesure par une convergence dans le temps : on remplace l'index d'itération i par le temps t , de façon à n'effectuer qu'une itération à chaque instant.

Bien entendu, pour ce faire, les principes de l'algorithme CSMF seront largement utilisés, mais transformés en s'inspirant des algorithmes de type SP.

Il sera important de montrer que l'on retrouve, après convergence de ce nouvel algorithme, α minima dans le cas stationnaire, le sous-espace optimal pouvant être calculé avec le CSMF.

Ce chapitre est organisé comme suit. Tout d'abord on introduit la notion de poursuite de sous-espace (Tracking). Par la suite, on rappelle le principe des algorithmes de type Subspace-Projection (SP). Enfin, l'algorithme CSMFST ainsi que la preuve de sa convergence sont présentés.

III .2 Le Tracking : poursuite de sous-espace

III.2.1 Principe

Dans l'étude des phénomènes non stationnaires, une des méthodes les plus employées consiste à chercher, à chaque réception d'une nouvelle mesure, le sous-espace principal ou du moins une approximation de celui-ci.

Ce sous-espace est engendré par les vecteurs propres principaux, *i.e.* ceux associés aux valeurs propres principales (les plus grandes) de la matrice de covariance, qui évolue dans le temps, du processus étudié.

Le principe de la poursuite du sous-espace principal E^p de dimension p d'un signal de E^n où $p < n$ est le suivant : soit $\mathbf{x}(t) \in E^n$ un vecteur de données aléatoire observé à la date t et

$$\mathbf{C}(t) \equiv \mathbb{E}[\mathbf{x}(t)\mathbf{x}^T(t)] \in M_{n \times n}$$

sa matrice de covariance. Dans le cas d'un processus non stationnaire, $\mathbf{C}(t)$ varie avec le temps.

Dans le schéma adaptatif, une mise à jour classique de la matrice de covariance utilise une fenêtre à décroissance exponentielle :

$$\mathbf{C}(t) = \beta \mathbf{C}(t-1) + \mathbf{x}(t)\mathbf{x}^T(t)$$

où $0 < \beta \leq 1$ est le facteur d'oubli utilisé pour assurer la capacité de poursuite lorsque le processus est non stationnaire et $t-1$ la date d'estimation précédente.

Une autre méthode consiste à utiliser une fenêtre rectangulaire glissante de longueur L :

$$\mathbf{C}(t) = \mathbf{C}(t-1) + \frac{1}{L} (\mathbf{x}(t)\mathbf{x}^T(t) - \mathbf{x}(t-L)\mathbf{x}^T(t-L))$$

qui a l'inconvénient de devoir garder en mémoire un ensemble non négligeable de données.

Nous supposons qu'on est dans l'espace image de $\mathbf{C}(t)$ et que ses valeurs propres sont telles que :

$$\lambda_1(t) \geq \lambda_2(t) \geq \dots \geq \lambda_p(t) > \lambda_{p+1}(t) \geq \dots \geq \lambda_n(t) > 0.$$

A la date t , le sous-espace principal $E^p(t)$ est alors engendré par les vecteurs propres principaux, orthonormés, de $\mathbf{C}(t)$, notés $\mathbf{W}(t) = [\mathbf{w}_1(t) \ \mathbf{w}_2(t) \ \dots \ \mathbf{w}_p(t)] \in M_{n \times p}$ associés aux p plus grandes valeurs propres $\lambda_1(t), \dots, \lambda_p(t)$.

Poursuite du sous-espace optimal

III.2 Le Tracking : poursuite de sous-espace

On peut écrire,

$$W(t) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T \mathbf{H} = \mathbb{I}_p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H})$$

Cette solution peut être trouvée en faisant une décomposition en valeurs propres (EVD) de $\mathbf{C}(t)$ ou une décomposition en valeurs singulières (SVD) de la matrice des données et en sélectionnant par la suite les p vecteurs principaux (associés aux p plus grandes valeurs propres).

En stationnaire centré, on note $\mathbf{X}(t) = [\mathbf{x}(t) \dots \mathbf{x}(t - L + 1)] \in M_{n \times L}$ et la matrice de corrélation est, typiquement, estimée par :

$$\widehat{\mathbf{C}}(t) \equiv \frac{1}{L} \mathbf{X}(t) \mathbf{X}^T(t)$$

quand la vraie matrice de corrélation est $\mathbf{C}(t) = \mathbb{E}[\mathbf{x}(t) \mathbf{x}^T(t)] = \mathbb{E}[\widehat{\mathbf{C}}(t)]$.

La SVD de $\mathbf{X}(t) = \mathbf{U} \mathbf{S} \mathbf{V}^T$ avec $\mathbf{U} \in M_{n \times n}$ et $\mathbf{V} \in M_{L \times L}$ matrices unitaires est $\mathbf{S} \in M_{n \times L}$ matrice diagonale d'éléments non nuls positifs.

\mathbf{U} , \mathbf{S} et \mathbf{V} dépendent du temps, mais par souci de simplicité des écritures, on ne note pas cette dépendance.

La SVD donne :

$$\mathbf{X}(t) \mathbf{X}^T(t) = \mathbf{U} \mathbf{S} \mathbf{S}^T \mathbf{U}^T$$

ou

$$(\mathbf{X}(t) \mathbf{X}^T(t)) \mathbf{U} = \mathbf{U} \mathbf{S} \mathbf{S}^T$$

Les vecteurs singuliers à gauche de $\mathbf{X}(t)$, sont vecteurs propres de $\mathbf{X}(t) \mathbf{X}^T(t)$ donc de $\widehat{\mathbf{C}}(t)$.

L'information contenue dans une SVD de $\mathbf{X}(t)$ ou une EVD de $\widehat{\mathbf{C}}(t)$ est la même.

Notons que pour $\mathbf{H} \in M_{n \times p}$ vérifiant $\mathbf{H}^T \mathbf{H} = \mathbb{I}_p$,

$$\operatorname{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H}) = \operatorname{tr}(\mathbf{C}(t) \mathbf{H} \mathbf{H}^T)$$

et si on note $\mathbf{\Pi} = \mathbf{H} \mathbf{H}^T$ la projection orthogonale sur E^p (matrice de $M_{n \times n}$ de rang p telle que $\mathbf{\Pi}^2 = \mathbf{\Pi}$), alors on peut chercher

$$\mathbf{\Pi}(t) = \underset{\substack{\mathbf{\Pi} \in M_{n \times n} \\ \mathbf{\Pi}^2 = \mathbf{\Pi}, \operatorname{rang}(\mathbf{\Pi}) = p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{\Pi} \mathbf{C}(t))$$

et il existe alors une infinité de matrices $\mathbf{V}(t)$ telles que $\mathbf{\Pi}(t) = \mathbf{V}(t) \mathbf{V}^T(t)$ avec $\mathbf{V}(t)$ ne contenant pas nécessairement les vecteurs propres de $\mathbf{C}(t)$.

III.2.2 Nécessité de mettre en place des algorithmes

Dans de nombreuses situations, les phénomènes étudiés voient leurs propriétés statistiques évoluer avec le temps ; bien entendu, le sous-espace principal peut alors être modifié lui aussi. Afin d'atteindre les objectifs fixés (détection, ...), il est naturel de vouloir mettre en œuvre des techniques permettant de connaître en temps réel le sous-espace principal. Ces techniques constituent les algorithmes de tracking ou poursuite de sous-espace.

Le calcul systématique est coûteux puisque la complexité est $O(n^3)$, ce qui est pénalisant et inenvisageable pour le temps réel. De plus, cette méthode n'est pas adaptative c'est-à-dire que l'estimation du nouveau sous-espace ne tient pas compte de l'estimée précédente.

Il est donc nécessaire de mettre en œuvre des algorithmes adaptatifs plus rapides et dont la complexité est réduite, comme par exemple les algorithmes de type SP.

III.2.3 Algorithmes de poursuite du sous-espace principal

Une brève revue de la littérature, extrêmement riche, sur les principales contributions de l'estimation adaptative du sous-espace nous permet de constater que les approches proposées sont très diverses et de classer les algorithmes de tracking suivant différents critères : poursuite de sous-espace principal ou mineur, des vecteurs propres, des vecteurs singuliers à droite, orthonormalité de la base obtenue, complexité algorithmique, stabilité numérique.

Bien entendu, l'objectif ultime est d'améliorer la capacité de poursuite tout en réduisant la complexité algorithmique et en assurant la stabilité numérique.

On s'intéresse à leur classement par ordre de complexité en termes de calcul.

Les algorithmes proposés sont basés sur :

- La méthode d'itération bi-orthogonale : calcul de la SVD complète de la matrice des données qui permet une transposition aisée au traitement adaptatif en remplaçant l'index d'itération i par le temps t , de façon à n'effectuer qu'une itération à chaque instant. Cette idée forte sera reprise dans l'algorithme CSMFST, objet de ce chapitre. Initialement proposée par Strobach (algorithme *Bi-iteration SVD* ou Bi-SVD) [52] de complexité $O(np^2)$, cette méthode a été déclinée par Badeau, algorithme SWASVD [53], et d'autres auteurs comme dans [54] et [55].
- L'approximation par projection : c'est une méthode qui se base sur une interprétation nouvelle du sous-espace signal comme solution d'un problème de minimisation d'une projection des mesures. Les algorithmes de cette famille donnent alors une solution en faisant une approximation de l'espérance mathématique et en utilisant un facteur d'oubli.

Poursuite du sous-espace optimal

III .2 Le Tracking : poursuite de sous-espace

Cette catégorie se compose de L'algorithme *Projection Approximation Subspace Tracking* [56] (PAST) et ses variantes : OPAST [57], HPAST [58], NPAST [59], et FCPAST [60] qui sont des approches célèbres pour le suivi rapide du sous-espace.

Classement des algorithmes par ordre de complexité en termes de calcul

Si p désigne le rang du sous-espace principal, optimal ou mineur qu'on souhaiterait estimer, les algorithmes nécessitant $O(n^2p)$ ou $O(n^2)$ opérations par mise à jour sont classés dans la catégorie haute complexité comme Rank Revealing QR [61] ; les algorithmes avec $O(np^2)$ opérations sont de complexité moyenne et enfin, les algorithmes avec $O(np)$ opérations sont peu complexes. Cette dernière catégorie est la plus importante d'un point de vue de mise en œuvre en temps réel où la non stationnarité du système nécessite des mises à jour régulières pour un suivi rapide des sous-espaces, et les algorithmes de cette classe sont également connus dans la littérature sous le nom d'algorithmes de suivi de sous-espace rapides.

En particulier, la méthode Oja [62] a connu un intérêt particulier et a été utilisée dans certaines applications réelles.

Par la suite, d'autres algorithmes plus rapides ont été proposés tels que : PAST [56] et sa variante orthogonale OPAST [57].

La poursuite du sous-espace peut être aussi basée sur l'algorithme de puissance itérée, c'est le cas de LORAF2 [63] et FAPI [64] [65] et sur le principe de projection qui donne naissance à la famille d'algorithmes de type SP [66].

Ces différents groupes d'algorithmes de tracking différents seront présentés rapidement dans ce qui suit [67]

a) Algorithmes de tracking de la famille PAST

Le membre original de cette famille, l'algorithme PAST, a été proposé par Yang [56]. Cet algorithme ne fournit pas une solution orthonormale, cependant, d'autres algorithmes mentionnés ci-dessus donnent lieu à une estimation orthonormale.

Dans la littérature, l'algorithme NIC présenté dans [68] est interprété comme une généralisation du PAST. Les dernières variantes du PAST, qui présentent plusieurs modifications robustes du PAST original, sont présentés dans [69]. Il faut noter que, sauf pour certains algorithmes de la famille PAST, tous les algorithmes de suivi du sous-espace mentionnés fournissent une base orthonormale à chaque pas de temps.

Une brève revue de cette famille est présentée dans ce qui suit [70] :

- **L'algorithme PAST**

La méthode PAST utilise la même fonction d'erreur quadratique moyenne que la méthode Oja plus une approximation simplificatrice dite « approximation de projection ». La vitesse de convergence

Poursuite du sous-espace optimal

III .2 Le Tracking : poursuite de sous-espace

de cette méthode est en général largement supérieure à celle de la méthode Oja. Aussi, dans presque toutes les situations la méthode PAST converge vers une matrice orthogonale dont les vecteurs colonnes génèrent le sous-espace principal.

Cependant, cet algorithme ne garantit pas l'orthonormalité de la matrice liée au sous-espace estimée [71], bien que cette ortho-normalité puisse être souhaitée dans certaines applications [72].

C'est pour cette raison que l'algorithme OPAST qui est la version orthogonale du PAST a été mis en place.

Une extension de l'algorithme PAST appelée PASTd, a aussi été présentée par Yang. Il s'agit d'une méthode où la technique de déflation est utilisée pour l'estimation séquentielle des vecteurs propres et des valeurs propres.

- **L'algorithme OPAST**

Afin d'assurer la convergence globale de la méthode PAST et, plus important, pour garantir l'orthogonalité de la matrice de poids à chaque itération, une nouvelle méthode PAST orthogonale (algorithme OPAST [57]) a été proposée. L'algorithme OPAST s'implémente avec une complexité linéaire, i.e., $4np + O(p^2)$ opérations au lieu de $3np + O(p^2)$ opérations pour l'algorithme PAST. En outre, l'algorithme OPAST converge aussi rapidement (parfois même plus rapidement que l'algorithme PAST original.

L'algorithme OPAST est une modification de l'algorithme PAST où la matrice de poids utilisée est forcée à être orthonormale à chaque itération.

- **L'algorithme NPAST**

L'algorithme PAST normalisé NPAST [59] pour l'analyse des composantes principales a été développé, en introduisant à l'algorithme PAST, une étape de normalisation à chaque mise à jour [59,73]. L'algorithme NPAST est le cas particulier le plus simple de l'algorithme OPAST.

Cet algorithme s'est montré plus stable que l'algorithme PAST pour le tracking du sous-espace principal.

- **L'algorithme HPAST**

L'algorithme HPAST assure l'ortho-normalité du sous-espace associé à la matrice signal estimée. Cette méthode utilise des transformations Householder pour mettre à jour les lignes de la matrice sous-espace estimée à chaque instant.

Les simulations présentées dans [58], montrent un comportement stable de cet algorithme dans le tracking de sous-espace.

Poursuite du sous-espace optimal

III.2 Le Tracking : poursuite de sous-espace

- **L'algorithme FCPAST**

L'algorithme FCPAST [60] suit le sous-espace signal de manière récursive pour chercher la solution orthonormale à chaque étape.

Il est basé sur une contrainte de minimisation lié au sous-espace signal. Cet algorithme garantit l'ortho-normalité de la base du sous-espace signal estimé à chaque itération.

Ainsi, l'algorithme évite le processus d'ortho-normalisation après chaque mise à jour.

Cette méthode présente une complexité de $O(np)$.

b) Algorithmes de tracking basés sur le principe de puissance itérée

Le principe d'itération orthogonale récursive (puissance itérée) [74] est une méthode de calcul de vecteurs propres principaux de la matrice signal. Plusieurs algorithmes tels comme NP3 [71], LORAF [63] FAPI [65], FDPM [75], NDPMB3 [76], et OSD [77] sont basés sur ce schéma classique.

- **Algorithmes OSD ET FAPI**

Les deux algorithmes OSD et FAPI ont une performance globale acceptable. En effet, ils possèdent un bon taux de convergence avec une erreur d'état stable et raisonnable en comparaison avec d'autres algorithmes de tracking rapides.

En particulier, l'analyse théorique a révélé que, parmi les algorithmes orthogonaux existants basés sur l'itération, FAPI était optimal [77].

- **Algorithmes NP3, LORAF3, FDPM ET NDPMB**

Ces quatre algorithmes sont aussi basés sur le principe des puissances itérées.

Des simulations numériques ont montré que NP3 ne convergeait pas dans de nombreuses situations [65], et que FDPM et LORAF3 produisent des estimations de sous-espace dégradées [65,77].

Par ailleurs, quelques résultats expérimentaux présentés dans [65], montrent une bonne performance pour FDPM.

Une comparaison entre les algorithmes de la famille FAPI et celle des algorithmes PAST, a montré dans [78] que les algorithmes PAST et OPAST peuvent être vus comme des approximations de l'algorithme FAPI. Ceci justifie les performances dégradées de ces deux algorithmes comparés à FAPI.

L'algorithme NDPMB a une structure similaire à FDPM et a été développé dans le but de réduire simplement la complexité de FDPM au plus bas niveau.

c) Algorithmes de tracking basés sur la méthode de calcul itératif SVD

Certains autres algorithmes rapides reposent sur une méthode de calcul itératif SVD qui est appelé le principe de bi-itération.

Le Bi-SVD2 [52] et sa version modifiée Bi-LS3 [54] sont deux instances de ce type d'algorithme. Parmi ces algorithmes basés sur le calcul itératif de SVD, Bi-LS3, qui nécessite deux fois plus de calculs en comparaison avec l'algorithme FAPI, a une performance globale acceptable, et est similaire à ce dernier [67].

d) Algorithmes de tracking de type SP

Cette famille d'algorithme sera détaillée davantage dans le prochain sous-chapitre. Elle regroupe les algorithmes de type SP1 et SP2 [66]. On peut citer les algorithmes YAST ou NFYAST [79,80], SGYAST et GYAST [67].

Ces algorithmes présentent de très bonnes performances pour le tracking de sous-espace signal.

- **Algorithmes SP-1 et SP-2**

L'idée principale des algorithmes SP est de faire une sélection optimale de la matrice sous-espace à chaque mise à jour des données en sélectionnant le « meilleur » sous-espace de dimension p dans un espace augmenté. Ce sous-espace augmenté est de dimension $p + 1$ (SP-1) ou $p + 2$ (SP-2) et est construit à partir de l'estimée du sous-espace précédent à laquelle vient se rajouter une ou deux directions fonction du nouveau vecteur donné reçu à l'instant t . Sa complexité globale est $O(np^2)$ alors que d'autres algorithmes ne requièrent que $O(np)$ opérations à chaque instant (PAST, FST, NIC, OPAST).

- **Algorithme YAST**

L'algorithme YAST proposé par Badeau [79,80] est une implémentation rapide des algorithmes SP et calcule le même espace signal que l'algorithme SP mais requiert seulement $O(np)$ opérations. Il peut converger plus rapidement que les algorithmes FAPI et Bi-LS3. Plus précisément, l'algorithme YAST, a le meilleur taux de convergence parmi tous les algorithmes de tracking rapides.

Cette performance est due à l'approche optimale des algorithmes SP [66] et sera illustré par un exemple de comparaison d'algorithmes dans ce qui suit.

- **Algorithme NFYAST**

L'algorithme NFYAST est une version simplifiée du YAST qui a approximativement les mêmes performances que la méthode YAST.

Elle a été introduite pour gagner en termes de temps de calculs.

Poursuite du sous-espace optimal

III.2 Le Tracking : poursuite de sous-espace

- **Algorithme GYAST**

Le YAST généralisé est une version de l'algorithme YAST qui généralise l'application de l'algorithme YAST souvent limitée au domaine temporel.

La complexité de cet algorithme est de $3np$ opérations par mise à jour, ce qui présente le plus faible coût parmi tous les algorithmes de tracking rapides. Cependant, il a été prouvé théoriquement que cet algorithme pourrait avoir une instabilité numérique [67].

- **Algorithme SGYAST**

L'algorithme SGYAST est une version stable du GYAST qui vient corriger le problème d'instabilité numérique rencontré avec le GYAST tout en respectant l'ortho-normalité. L'algorithme SGYAST a une complexité de $6np$. Des applications montrent que la performance globale de SGYAST est similaire à celle de GYAST [67].

Résumé : Tableau de complexités [67]

Algorithme	NFYAST	SGYAST	GYAST	SP	FAP	Bi-LS3	OSD	LORAF3	FDPM	NDPM
Complexité par itération	$O(n^2)$	$6np$	$3np$	np^2	$3np$	$6np$	$3np$	$4np$	$6np$	$3np$

e) Comparaison entre certains algorithmes de tracking

N'ayant pas de références pour nos signaux d'acoustique sous-marine, on recherche dans la littérature une comparaison entre les algorithmes de tracking existants : en se référant alors aux travaux de M.Arjomandi et M. Karimi, on présente, dans cette partie, une simulation issue de [67], qui a pour but de comparer les performances des algorithmes de tracking cités précédemment en termes de vitesse de convergence et d'orthogonalité de la solution.

Cette comparaison sera réalisée dans un premier lieu dans un environnement stationnaire puis dans un cas non stationnaire.

Comparaison dans le cas stationnaire

L'exemple considère 3 sources fixes arrivant respectivement des directions 20° , 35° et 50° .

La figure III.1 (a) et (b) issues de [67] représentent l'erreur de projection et les erreurs d'orthogonalité des matrices dans le cas stationnaire.

Poursuite du sous-espace optimal

III.2 Le Tracking : poursuite de sous-espace

Le résultat de la Figure III.1 (a) montre que parmi les algorithmes de tracking rapides, NFYAST, GYAST et SGYAST ont des performances similaires et présentent le meilleur taux de convergence en atteignant les -50 dB.

Les méthodes FAPI et Bi-LS3 convergent presque vers la même valeur mais moins rapidement.

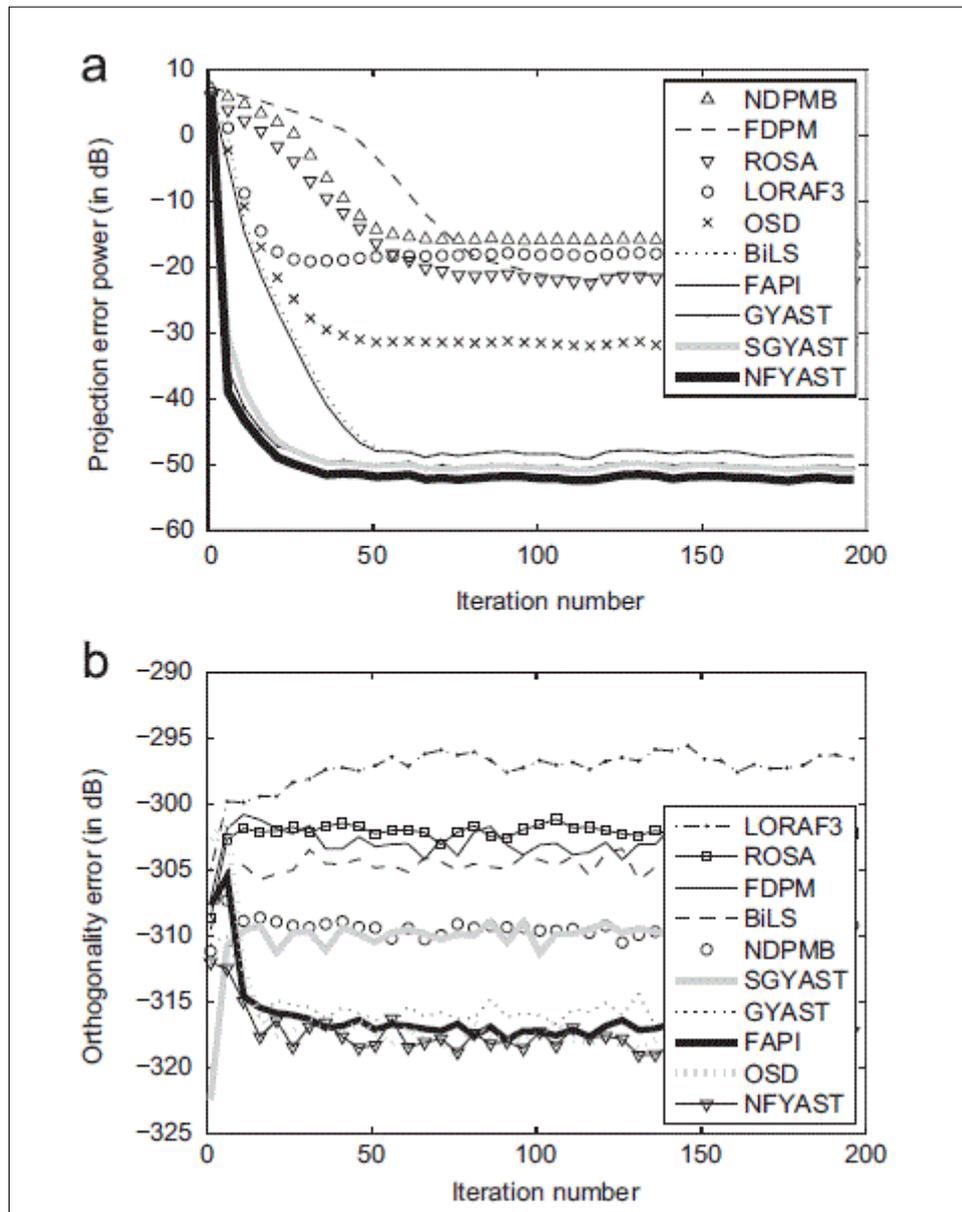


Figure III.1. Comparaison des algorithmes de tracking en termes de convergence (a) et d'orthogonalité (b) dans le cas stationnaire [67]

La Figure III.1 (b), montre que tous les algorithmes ont des erreurs d'ortho-normalité acceptables vu que toutes les courbes se situent au-dessous des -295 dB.

Poursuite du sous-espace optimal

III .2 Le Tracking : poursuite de sous-espace

Comparaison dans le cas non-stationnaire

La deuxième simulation considère deux sources mobiles pouvant se croiser et une source fixe. La trajectoire de ces sources est représentée sur la figure III.2 issue de [67]

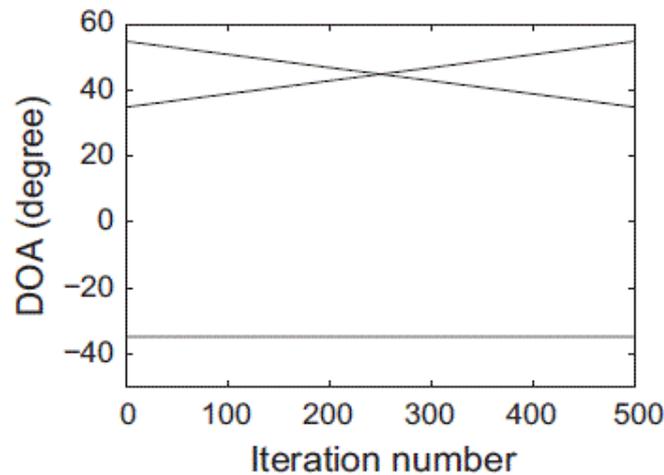


Figure III.2. trajectoire des sources [67]

Les résultats de cette simulation sont présentés sur la Figure III.3 (a) et (b) issues de [67].

Afin d'analyser les résultats de convergence donnés sur la Figure III.3 (a), il faut découper l'intervalle de temps en 3 sous-intervalles :

De 1-200 itérations : cet intervalle correspond à la zone où les sources mobiles ne se sont pas encore croisées et le comportement des algorithmes est similaire au cas stationnaire.

De 200-700 itérations : cet intervalle correspond à la zone où les sources se croisent ce qui explique l'augmentation de l'erreur de projection et on peut remarquer que plus les sources se rapprochent plus les performances des algorithmes se dégradent.

Les algorithmes reconvergent vers le bon sous- espace à partir de l'itération 550 où les sources commencent à s'éloigner l'une de l'autre.

De 700-1000 itérations : On retrouve les mêmes performances des algorithmes au début de l'intervalle de temps.

En ce qui concerne l'ortho-normalité des algorithmes, la Figure III.3 (b) montre que dans le deuxième intervalle, ce sont les algorithmes de type SP qui assure au mieux l'ortho-normalité vu que l'erreur de projection augmente peu pour ces algorithmes et reste inférieure à -305 dB.

Poursuite du sous-espace optimal

III.2 Le Tracking : poursuite de sous-espace

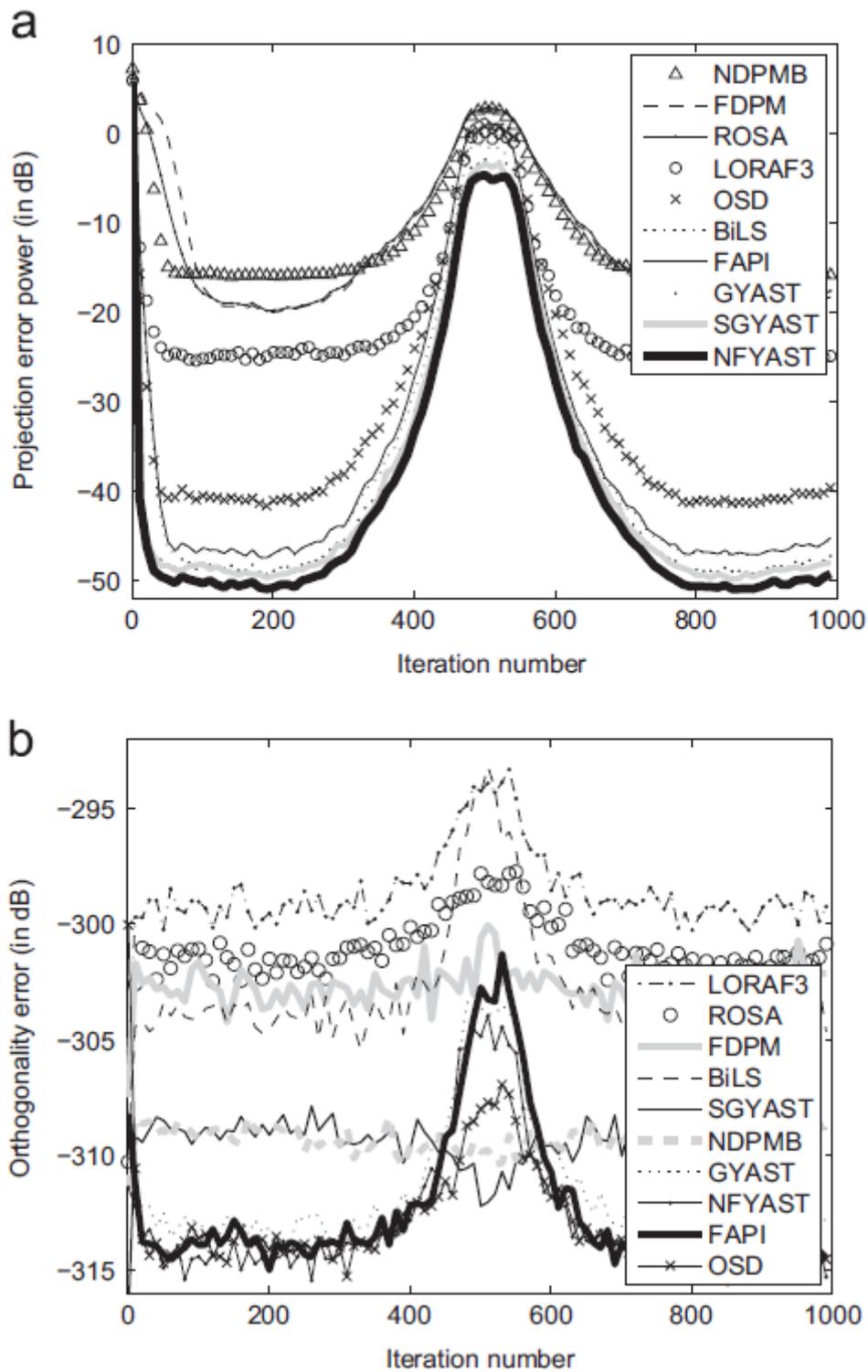


Figure III.3. Comparaison des algorithmes de tracking en termes de convergence (a) et d'orthogonalité (b) dans le cas non stationnaire [67]

III .3 Les algorithmes de type SP

On se propose de détailler ici l'algorithme SP-1 de Davila [66] puisqu'il est à la base de l'algorithme que nous proposons pour répondre à notre problématique. Il sera dit un mot de l'algorithme SP-2, mais son instabilité constatée nous dissuade de l'utiliser.

On dispose de mesures $\{\mathbf{x}(t)\}_{t \geq 0}$, vecteurs de E^n , qui constituent une classe de signaux tels que $E\{\mathbf{x}(t)\mathbf{x}^T(t)\} \equiv \mathbf{C}(t) \in M_{n \times n}$. t est une mesure du temps et on considère des signaux éventuellement non stationnaires, *i.e.* dont les propriétés statistiques évoluent dans le temps.

L'objectif est de trouver, à l'instant t , le sous-espace principal $E^p(t)$ ($p < n$) ; il est bien connu qu'une matrice orthonormée $\mathbf{W}(t) \in M_{n \times p}$ engendre le sous-espace principal de dimension p de $\mathbf{C}(t)$ si et seulement si elle maximise le critère $\text{tr}(\mathbf{W}^T(t)\mathbf{C}(t)\mathbf{W}(t))$:

$$\mathbf{W}(t) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T \mathbf{H} = \mathbb{I}_p}}{\text{argmax}} \text{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H}) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T \mathbf{H} = \mathbb{I}_p}}{\text{argmax}} \frac{\text{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H})}{\text{tr}(\mathbf{H}^T \mathbf{H})}$$

Alors, le maximum de ce critère est égal à la somme des p valeurs propres principales de $\mathbf{C}(t)$.

La matrice $\mathbf{W}(t)$ constituée des p vecteurs propres principaux de $\mathbf{C}(t)$ est une solution.

Malheureusement, l'implémentation de ce calcul est lourde (vecteurs propres d'une matrice $n \times n$) et doit être réalisé pour chaque valeur de t . De plus, elle ne conduit pas à une simple récursivité entre $\mathbf{W}(t)$ et $\mathbf{W}(t - 1)$.

Afin de réduire le coût, l'idée introduite dans [66] consiste à limiter cette recherche de la matrice $\mathbf{W}(t)$ au sous-espace image de $\mathbf{W}(t - 1)$ augmenté d'une ou plusieurs directions additionnelles fonction de la nouvelle donnée $\mathbf{x}(t)$.

Pour SP-1, on cherche donc, à l'instant t , une approximation de $\mathbf{W}(t)$, notée $\widehat{\mathbf{W}}(t)$ dans le sous-espace $\text{Span}\{\underline{\mathbf{W}}\}$ où on note :

$$\underline{\mathbf{W}} = [\widehat{\mathbf{W}}(t - 1) \ \mathbf{x}(t)] \in M_{n \times (p+1)}.$$

C'est-à-dire que l'on cherche $\mathbf{V}(t) \in M_{(p+1) \times p}$ tel que

$$\widehat{\mathbf{W}}(t) = \underline{\mathbf{W}} \mathbf{V}(t) \tag{III.1}$$

i.e.

$$\widehat{\mathbf{W}}(t) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T \mathbf{H} = \mathbb{I}_p \\ \mathbf{H} \in \text{Span}\{\underline{\mathbf{W}}\}}}{\text{argmax}} \text{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H})$$

Dire que $\widehat{\mathbf{W}}(t) \in \text{Span}\{\underline{\mathbf{W}}\}$ et $\widehat{\mathbf{W}}(t) \in M_{n \times p}$, revient à trouver :

Poursuite du sous-espace optimal

III.3 Les algorithmes de type SP

$\mathbf{V}(t) \in M_{(p+1) \times p} \mid \widehat{\mathbf{W}}(t) = \underline{\mathbf{W}}\mathbf{V}(t)$. Alors si l'on note $\mathbf{P}(t) = \underline{\mathbf{W}}^T \mathbf{C}(t) \underline{\mathbf{W}}$ et $\mathbf{Q}(t) = \underline{\mathbf{W}}^T \underline{\mathbf{W}}$ les matrices de $M_{(p+1) \times (p+1)}$, on a

$$\widehat{\mathbf{W}}^T(t) \mathbf{C}(t) \widehat{\mathbf{W}}(t) = \mathbf{V}^T(t) \underline{\mathbf{W}}^T \mathbf{C}(t) \underline{\mathbf{W}} \mathbf{V}(t) = \mathbf{V}^T(t) \mathbf{P}(t) \mathbf{V}(t)$$

et

$$\widehat{\mathbf{W}}^T(t) \widehat{\mathbf{W}}(t) = \mathbf{V}^T(t) \underline{\mathbf{W}}^T \underline{\mathbf{W}} \mathbf{V}(t) = \mathbf{V}^T(t) \mathbf{Q}(t) \mathbf{V}(t)$$

et le problème se ramène à trouver

$$\mathbf{V}(t) = \underset{\substack{\mathbf{V} \in M_{(p+1) \times p} \\ \mathbf{V}^H \mathbf{Q}_t \mathbf{V} = \mathbb{I}_p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{V}^T \mathbf{P}(t) \mathbf{V})$$

qui est un problème beaucoup plus léger en calculs : la solution est connue, il s'agit de prendre pour $\mathbf{V}(t)$ l'ensemble des p vecteurs propres principaux de $\mathbf{Q}(t)^{-1} \mathbf{P}(t)$, et $\mathbf{V}(t)$ diagonalise alors conjointement $\mathbf{P}(t)$ et $\mathbf{Q}(t)$.

On a

$$\mathbf{P}(t) = \begin{bmatrix} \widehat{\mathbf{W}}^T(t-1) \mathbf{C}(t) \widehat{\mathbf{W}}(t-1) & \widehat{\mathbf{W}}^T(t-1) \mathbf{C}(t) \mathbf{x}(t) \\ \mathbf{x}^T(t) \mathbf{C}(t) \widehat{\mathbf{W}}(t-1) & \mathbf{x}^T(t) \mathbf{C}(t) \mathbf{x}(t) \end{bmatrix} \in M_{(p+1) \times (p+1)}$$

et

$$\mathbf{Q}(t) = \begin{bmatrix} \widehat{\mathbf{W}}^T(t-1) \widehat{\mathbf{W}}(t-1) = \mathbb{I}_p & \widehat{\mathbf{W}}^T(t-1) \mathbf{x}(t) \\ \mathbf{x}^T(t) \widehat{\mathbf{W}}(t-1) & \mathbf{x}^T(t) \mathbf{x}(t) \end{bmatrix} \triangleq \begin{bmatrix} \mathbb{I}_p & \mathbf{y} \\ \mathbf{y}^T & a \end{bmatrix}$$

On a

$$\mathbf{Q}_k^{-1} \triangleq \begin{bmatrix} \mathbb{I}_p & \mathbf{y} \\ \mathbf{y}^T & a \end{bmatrix}^{-1} = \begin{bmatrix} \mathbb{I}_p + \kappa \mathbf{y} \mathbf{y}^T & -\kappa \mathbf{y} \\ -\kappa \mathbf{y}^T & \kappa \end{bmatrix},$$

avec

$$\kappa = \frac{1}{a - \mathbf{y}^T \mathbf{y}} = \frac{1}{\mathbf{x}_k^T \mathbf{x}_k - \mathbf{y}^T \mathbf{y}} = \frac{1}{\mathbf{x}_k^T \mathbf{x}_k - \mathbf{x}_k^T \widehat{\mathbf{W}}_{k-1} \widehat{\mathbf{W}}_{k-1}^T \mathbf{x}_k} = \frac{1}{\mathbf{x}_k^T (\mathbb{I}_n - \widehat{\mathbf{W}}_{k-1} \widehat{\mathbf{W}}_{k-1}^T) \mathbf{x}_k}$$

On note $\widehat{E}^p(t) = \operatorname{Span}\{\widehat{\mathbf{W}}(t)\}$ le sous-espace qui est une approximation de $E^p(t)$ à la date t . Bien entendu, il faut que l'algorithme converge vers le bon sous-espace principal, c'est-à-dire que, dans la situation d'un processus stationnaire

$$\lim_{t \rightarrow \infty} \widehat{E}^p(t) = E^p(t),$$

ce qui a été démontré dans [66].

Si l'on définit

$$\mathbf{y} = \widehat{\mathbf{W}}_{k-1}^T \mathbf{x}_k$$

alors

$$\mathbb{E}\{\mathbf{y} \mathbf{y}^T\} = \mathbb{E}\{\widehat{\mathbf{W}}_{k-1}^T \mathbf{x}_k \mathbf{x}_k^T \widehat{\mathbf{W}}_{k-1}\} = \widehat{\mathbf{W}}_{k-1}^T \mathbb{E}\{\mathbf{x}_k \mathbf{x}_k^T\} \widehat{\mathbf{W}}_{k-1} = \widehat{\mathbf{W}}_{k-1}^T \mathbf{C}_k \widehat{\mathbf{W}}_{k-1}$$

Poursuite du sous-espace optimal

III.3 Les algorithmes de type SP

et

$$\mathbf{P}_k(t) = \begin{bmatrix} \mathbb{E}\{\mathbf{y}\mathbf{y}^T\} & \widehat{\mathbf{W}}_{k-1}^T \mathbf{C}_k \mathbf{x}_k \\ \mathbf{x}_k^T \mathbf{C}_k \widehat{\mathbf{W}}_{k-1} & \mathbf{x}_k^T \mathbf{C}_k \mathbf{x}_k \end{bmatrix}$$

Badeau a proposé une solution alternative [80] qui consiste à augmenter la base non pas par la nouvelle mesure reçue $\mathbf{x}(t)$ mais par un vecteur $\mathbf{u}(t)$ comme suit :

$$\underline{\mathbf{W}} \equiv [\widehat{\mathbf{W}}(t-1) \quad \mathbf{u}(t)] \in M_{n \times (p+1)} \quad (\text{III.2})$$

où

$$\mathbf{u}(t) = \frac{\mathbf{x}_\perp(t)}{\|\mathbf{x}_\perp(t)\|} = \frac{\mathbf{e}(t)}{\|\mathbf{e}(t)\|}$$

Et

$$\mathbf{e}(t) = (\mathbb{I}_n - \widehat{\mathbf{W}}(t-1)\widehat{\mathbf{W}}^T(t-1))\mathbf{x}(t). \quad (\text{III.3})$$

i.e. $\mathbf{u}(t) \perp \widehat{\mathbf{W}}(t-1)$ ou $\widehat{\mathbf{W}}^T(t-1)\mathbf{u}(t) = 0$, et donc $\underline{\mathbf{W}}^T \underline{\mathbf{W}} = \mathbb{I}_{p+1}$.

On a alors

$$\mathbf{P}(t) = \begin{bmatrix} \widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1) & \widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\mathbf{u}(t) \\ \mathbf{u}^T(t)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1) & \mathbf{u}^T(t)\mathbf{C}(t)\mathbf{u}(t) \end{bmatrix}$$

et

$$\mathbf{Q}(t) = \mathbb{I}_{p+1}$$

et le problème se ramène à trouver

$$\mathbf{V}(t) = \underset{\substack{\mathbf{V} \in M_{(p+1) \times p} \\ \mathbf{V}^T \mathbf{V} = \mathbb{I}_p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{V}^T \mathbf{P}(t) \mathbf{V})$$

dont la solution consiste à prendre pour $\mathbf{V}(t)$ l'ensemble des p vecteurs propres principaux de

$$\mathbf{P}(t) = \underline{\mathbf{W}}^T \mathbf{C}(t) \underline{\mathbf{W}} \quad (\text{III.4})$$

La preuve de la convergence sera utile pour démontrer celle de notre algorithme et est donc donnée ci-après, dans le cadre du cas particulier énoncé ci-dessus et évidemment dans le cas de processus stationnaire (on prend donc $\beta = 1$).

On va utiliser le théorème suivant [81] :

Poursuite du sous-espace optimal

III .3 Les algorithmes de type SP

Théorème [81] :

Si on considère $\mathbf{A} \in M_{n \times n}$ une matrice hermitienne connue, $\mathbf{y} \in \mathbb{C}^n$ un vecteur donné et $a \in \mathbb{R}$ un nombre réel donné, on peut obtenir une matrice hermitienne $\hat{\mathbf{A}} \in M_{(n+1) \times (n+1)}$ en bordant \mathbf{A} avec \mathbf{y} et a comme suit :

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{y} \\ \mathbf{y}^H & a \end{bmatrix}$$

Si on note $\{\lambda_i\}$ et $\{\hat{\lambda}_i\}$ respectivement les valeurs propres de \mathbf{A} et $\hat{\mathbf{A}}$ et on considère qu'elles ont été rangées dans l'ordre croissant :

$$\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$$

Et

$$\hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \dots \leq \hat{\lambda}_n \leq \hat{\lambda}_{n+1}$$

Alors,

$$\hat{\lambda}_1 \leq \lambda_1 \leq \hat{\lambda}_2 \leq \lambda_2 \leq \dots \leq \lambda_{n-1} \leq \hat{\lambda}_n \leq \lambda_n \leq \hat{\lambda}_{n+1}.$$

Conséquence : on constate qu'alors, on a, si l'on prend $\hat{\mathbf{V}}_p$ les p vecteurs propres principaux de $\hat{\mathbf{A}}$:

$$\text{tr}(\hat{\mathbf{V}}_p^H \hat{\mathbf{A}} \hat{\mathbf{V}}_p) = \hat{\lambda}_{p+1} + \hat{\lambda}_p + \dots + \hat{\lambda}_2 \geq \lambda_p + \dots + \lambda_2 + \lambda_1 = \text{tr}(\mathbf{A})$$

Ce théorème s'applique aux matrices hermitiennes, vu que nos matrices sont réelles symétriques, on peut appliquer le théorème dans ce cas particulier : on a, en identifiant $\mathbf{P}(t)$ à $\hat{\mathbf{A}}$ et $\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1)$ à \mathbf{A} :

$$\text{tr}(\mathbf{V}^T(t)\mathbf{P}(t)\mathbf{V}(t)) = \text{tr}(\widehat{\mathbf{W}}^T(t)\mathbf{C}(t)\widehat{\mathbf{W}}(t)) \geq \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1)).$$

Or, si l'on prend $\mathbf{C}(t) = \mathbf{C}(t-1) + \mathbf{x}(t)\mathbf{x}^T(t)$, on a

$$\begin{aligned} \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1)) \\ = \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t-1)\widehat{\mathbf{W}}(t-1)) + \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{x}(t)\mathbf{x}^T(t)\widehat{\mathbf{W}}(t-1)) \end{aligned}$$

et puisque

$$\begin{aligned} \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{x}(t)\mathbf{x}^T(t)\widehat{\mathbf{W}}(t-1)) &= \text{tr}(\mathbf{x}^T(t)\widehat{\mathbf{W}}(t-1)\widehat{\mathbf{W}}^T(t-1)\mathbf{x}(t)) \\ &= \mathbf{x}^T(t)\widehat{\mathbf{W}}(t-1)\widehat{\mathbf{W}}^T(t-1)\mathbf{x}(t) \geq 0, \end{aligned}$$

avec inégalité stricte si $\mathbf{x}(t)$ n'est pas orthogonal à $\widehat{\mathbf{W}}(t-1)$, on a

$$\text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t)\widehat{\mathbf{W}}(t-1)) \geq \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t-1)\widehat{\mathbf{W}}(t-1))$$

Poursuite du sous-espace optimal

III .4 L'algorithme CSMFST

et donc

$$\text{tr}(\widehat{\mathbf{W}}^T(t)\mathbf{C}(t)\widehat{\mathbf{W}}(t)) \geq \text{tr}(\widehat{\mathbf{W}}^T(t-1)\mathbf{C}(t-1)\widehat{\mathbf{W}}(t-1)).$$

La suite $\text{tr}(\widehat{\mathbf{W}}^T(t)\mathbf{C}(t)\widehat{\mathbf{W}}(t))$ constitue donc une suite croissante qui est nécessairement majorée par la somme des p plus grandes valeurs propres de $\mathbf{C}(t)$.

Le **pseudo-code** d'un tel algorithme est le suivant :

A chaque date t , réception d'une nouvelle mesure $\mathbf{x}(t)$, alors que l'on dispose d'une estimation de $\hat{E}^p(t-1) = \text{Span}\{\widehat{\mathbf{W}}(t-1)\}$

1. Calcul de la base orthonormée $\underline{\mathbf{W}} = [\widehat{\mathbf{W}}(t-1) \mathbf{u}(t)]$ du sous-espace augmenté
2. Calcul de la matrice $\mathbf{P}(t) = \underline{\mathbf{W}}^T \mathbf{C}(t) \underline{\mathbf{W}}$
3. Trouver les p plus grands vecteurs propres $\mathbf{V}(t)$ de $\mathbf{P}(t)$ et générer $\widehat{\mathbf{W}}(t) = \underline{\mathbf{W}}\mathbf{V}(t)$.

III .4 L'algorithme CSMFST

L'algorithme CSMF a été présenté dans le premier chapitre.

Rappelons simplement que, partant de données dans un espace de dimension n , le but du CSMF est de trouver le sous-espace optimal \mathcal{E}_p^* de dimension p dans lequel le rapport signal à bruit est maximisé, la valeur de p étant fixée et vérifiant $p \leq n$.

Seules les matrices de covariance \mathbf{A} et \mathbf{B} du signal et du bruit de dimension $n \times n$ sont nécessaires et peuvent être estimées à partir des réalisations.

$\mathcal{E}_p^* = \text{Span}\{\mathbf{W}_p^*\}$ où $\mathbf{W}_p^* \in M_{n \times p}$ vérifie $\mathbf{W}_p^{*T} \mathbf{W}_p^* = \mathbb{I}_p$.

Dans \mathcal{E}_p^* , le rapport signal à bruit ρ défini par

$$\rho = \frac{\text{tr}(\mathbf{W}_p^{*T} \mathbf{A} \mathbf{W}_p^*)}{\text{tr}(\mathbf{W}_p^{*T} \mathbf{B} \mathbf{W}_p^*)}$$

est maximal et vaut ρ^* .

Désormais, on se place dans une situation où l'on dispose d'échantillons de mesures $\mathbf{x}_A(t)$ et $\mathbf{x}_B(t)$ dépendant du temps dans le cadre de processus stationnaires.

Il existe donc des matrices de covariance de dimension $n \times n$, \mathbf{A} et \mathbf{B} et un sous-espace optimal noté \mathcal{E}_p^* .

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

On se propose de mettre en place un nouvel algorithme qui converge vers \mathcal{E}_p^* , à partir des réalisations temporelles des deux processus.

Ensuite, cet algorithme sera étendu au cas de processus non stationnaires.

Cet algorithme est nommé *Constrained Stochastic Matched Filter Subspace Tracking* (CSMFST).

Une démonstration de la convergence de l'algorithme est donnée dans ce paragraphe.

Des simulations sont réalisées dans le cas de processus stationnaires puis non stationnaires.

Les résultats obtenus seront comparés à ceux de l'algorithme CSMF ou à d'autres algorithmes existants dans le cas particulier du bruit blanc, seul cas où il existe un algorithme avec lequel on peut faire des comparaisons.

III.4.1 Présentation de l'algorithme

Dans cette partie, on présente l'algorithme CSMFST qui permet de converger vers \mathcal{E}_p^* .

Nous avons montré dans le chapitre précédent que la maximisation du RSB conduit à l'équation :

$$\mathbf{M}(\rho)\mathbf{W}_p(\rho) = \mathbf{W}_p(\rho)\Delta_u(\rho)$$

avec $\mathbf{M}(\rho) = \mathbf{A} - \rho\mathbf{B}$.

L'optimisation consiste à trouver la valeur de ρ^* et les vecteurs propres de la matrice $\mathbf{M}(\rho^*)$. Néanmoins, ρ est une variable inconnue positive et $\mathbf{M}(\rho)$ n'est pas définie positive.

De plus, dans l'algorithme CSMF simplifié, il a été constaté qu'on obtient le sous-espace optimal en sélectionnant à chaque itération k les p vecteurs propres principaux de la matrice $\{\mathbf{M}(\rho^{(k)})\}$.

On peut donc envisager une méthodologie basée sur le principe des algorithmes de type SP.

La version présentée de l'algorithme est basée sur la connaissance des réalisations du signal et du bruit séparément.

On verra par la suite, d'autres utilisations possibles de ce type d'algorithme.

A la date t , on dispose des réalisations du signal et du bruit $\mathbf{x}_A(t)$ et $\mathbf{x}_B(t)$.

Il serait donc possible à partir des matrices $\mathbf{A}(t)$ et $\mathbf{B}(t)$ d'appliquer l'algorithme récursif CSMF pour obtenir le sous espace optimal $\mathcal{E}_p^*(t)$ à chaque date t .

Ceci étant lourd en termes de calcul, on se propose de mettre en œuvre un algorithme qui remplace, dans un contexte stationnaire, la convergence par récursivité à chaque date t par une convergence dans le temps.

Bien évidemment, à la date t , on n'aura pas le sous espace optimal $\mathcal{E}_p^*(t)$ mais une estimation de celui-ci notée $\mathcal{E}_p(t)$.

$\mathcal{E}_p(t)$ est engendré par p vecteurs notés $\mathbf{W}(t)$.

L'objectif de l'algorithme est, connaissant $\mathbf{W}(t-1)$ et les mesures $\mathbf{x}_A(t)$ et $\mathbf{x}_B(t)$, d'en déduire $\mathbf{W}(t)$ et donc $\mathcal{E}_p(t)$ avec pour objectif que $\mathcal{E}_p(t)$ tende vers \mathcal{E}_p^* quand t tend vers « l'infini ».

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

A la date t , on dispose de $\mathbf{W}(t-1)$. En s'inspirant des algorithmes de type SP, le problème est d'augmenter $\mathbf{W}(t-1)$ pour définir $\underline{\mathbf{W}}(t)$ et par suite en déduire $\mathbf{W}(t)$.

En utilisant $\mathbf{x}_A(t)$ et $\mathbf{x}_B(t)$, on peut calculer les matrices de covariance :

$$\mathbf{A}(t) = \beta \mathbf{A}(t-1) + \mathbf{x}_A(t) \mathbf{x}_A(t)^T$$

et

$$\mathbf{B}(t) = \beta \mathbf{B}(t-1) + \mathbf{x}_B(t) \mathbf{x}_B(t)^T.$$

Et définir également le rapport signal à bruit intermédiaire :

$$\rho(t-1, t) = \frac{\text{tr}(\mathbf{W}(t-1)^T \mathbf{A}(t) \mathbf{W}(t-1))}{\text{tr}(\mathbf{W}(t-1)^T \mathbf{B}(t) \mathbf{W}(t-1))} \quad (\text{III.5})$$

Le double indilage indique clairement que l'on dispose de données à la date t et d'autres à la date $t-1$.

En se basant sur l'équation (III.3), on définit :

$$\mathbf{e}_A(t) = \mathbf{x}_A(t) - \mathbf{W}(t-1) \mathbf{W}(t-1)^T \mathbf{x}_A(t)$$

et

$$\mathbf{e}_B(t) = \mathbf{x}_B(t) - \mathbf{W}(t-1) \mathbf{W}(t-1)^T \mathbf{x}_B(t).$$

On cherche alors les p vecteurs propres principaux de $\mathbf{C}_{xx}(t)$ jouant le rôle de la matrice $\mathbf{M}(\rho)$ du CSMF :

$$\mathbf{C}_{xx}(t) \equiv \mathbf{A}(t) - \rho(t-1, t) \mathbf{B}(t). \quad (\text{III.6})$$

Afin d'augmenter d'une dimension $\mathbf{W}(t-1)$, on définit la matrice $\mathbf{M}(t)$:

$$\mathbf{M}(t) \equiv \mathbf{e}_A(t) \mathbf{e}_A(t)^T - \rho(t-1, t) \mathbf{e}_B(t) \mathbf{e}_B(t)^T$$

qui est au plus de rang deux.

On recherche le vecteur propre $\mathbf{u}(t)$ associé à la plus grande valeur propre de $\mathbf{M}(t)$, qui nous permettra de définir, comme dans l'équation (III.2), la matrice augmentée

$$\underline{\mathbf{W}}(t) \equiv [\mathbf{W}(t-1) \mathbf{u}(t)].$$

Comme pour les algorithmes de type SP :

$$\mathbf{u}(t) \perp \mathbf{W}(t-1)$$

Afin d'atteindre la matrice $\mathbf{W}(t) \in M_{n \times p}$, on sélectionne $\mathbf{U}(t)$ les p plus grands vecteurs propres de $\mathbf{P}(t)$ définie dans l'équation (III.4). D'où

$$\mathbf{W}(t) = \underline{\mathbf{W}}(t) \mathbf{U}(t).$$

On peut alors mettre à jour le rapport signal à bruit à la date t ,

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

$$\rho(t, t) \equiv \frac{\text{tr}(\mathbf{W}(t)^T \mathbf{A}(t) \mathbf{W}(t))}{\text{tr}(\mathbf{W}(t)^T \mathbf{B}(t) \mathbf{W}(t))} \quad (\text{III.7})$$

où tout dépend de t et le double indiçage peut devenir simple.

L'initialisation des matrices de covariance peut se faire de deux manières différentes.

Le pseudo-code de l'algorithme CSMFST est présenté dans la table 1.

TABLE 1 : Description de l'algorithme CSMFST

1) Initialisation

- $\mathbf{A}(t-1) = \mathbf{B}(t-1) = \mathbb{O}_{n \times n}$
- $\mathbf{W}(t-1) = \mathbb{I}_{n \times p}$

Ou

- $\mathbf{A}(t-1)$ et $\mathbf{B}(t-1)$ sont calculées à partir de N échantillons.
- $\mathbf{W}(t-1)$ est initialisé à l'aide des p plus grands valeurs propres associées aux p plus grandes valeurs propres de $\mathbf{B}(t-1)^{-1} \mathbf{A}(t-1)$ ou $\mathbf{W}(t-1) = \mathbb{I}_{n \times p}$.

Pour $t \geq 1$, mesures $\mathbf{x}_A(t)$ et $\mathbf{x}_B(t)$

2) Mise à jour des matrices :

$$\mathbf{A}(t) = \beta \mathbf{A}(t-1) + \mathbf{x}_A(t) \mathbf{x}_A(t)^T$$

$$\mathbf{B}(t) = \beta \mathbf{B}(t-1) + \mathbf{x}_B(t) \mathbf{x}_B(t)^T$$

et

$$\rho(t-1, t) = \frac{\text{tr}(\mathbf{W}(t-1)^T \mathbf{A}(t) \mathbf{W}(t-1))}{\text{tr}(\mathbf{W}(t-1)^T \mathbf{B}(t) \mathbf{W}(t-1))}$$

3) Calcul de la matrice

$$\mathbf{C}_{xx}(t) = \mathbf{A}(t) - \rho(t-1, t) \mathbf{B}(t)$$

4) Calcul de

$$\mathbf{e}_A(t) = \mathbf{x}_A(t) - \mathbf{W}(t-1) \mathbf{W}(t-1)^T \mathbf{x}_A(t)$$

$$\mathbf{e}_B(t) = \mathbf{x}_B(t) - \mathbf{W}(t-1) \mathbf{W}(t-1)^T \mathbf{x}_B(t)$$

5) Calcul de

$$\mathbf{M}(t) = \mathbf{e}_A(t) \mathbf{e}_A(t)^T - \rho(t-1, t) \mathbf{e}_B(t) \mathbf{e}_B(t)^T$$

6) Calcul du plus grand vecteur propre $\mathbf{u}(t)$ de $\mathbf{M}(t)$ [voir en annexe 3 le calcul théorique]

7) Calcul de la matrice $\underline{\mathbf{W}}(t)$:

$$\underline{\mathbf{W}}(t) = [\mathbf{W}(t-1) \mathbf{u}(t)]$$

8) Calcul de la matrice

$$\mathbf{P}(t) = \underline{\mathbf{W}}(t)^T \mathbf{C}_{xx}(t) \underline{\mathbf{W}}(t)$$

9) Sélection de $\mathbf{U}(t)$ les p plus grands vecteurs propres de $\mathbf{P}(t)$.

10) Mise à jour $\mathbf{W}(t)$ et calcul de $\rho(t, t)$

$$\mathbf{W}(t) = \underline{\mathbf{W}}(t)\mathbf{U}(t)$$

Et

$$\rho(t, t) = \frac{\text{tr}(\mathbf{W}(t)^T \mathbf{A}(t) \mathbf{W}(t))}{\text{tr}(\mathbf{W}(t)^T \mathbf{B}(t) \mathbf{W}(t))}$$

$$t = t + 1$$

Si $\{|\rho(t, t) - \rho(t - 1, t)| < \varepsilon\}$ on s'arrête,

sinon

Aller à 2)

Fin

III.4.2 Convergence de l'algorithme

Le but de cette section est de prouver, dans un cas stationnaire, que le rapport signal à bruit $\rho(t, t)$ converge vers ρ^* et que $\mathcal{E}_p(t)$ converge vers le sous-espace optimal \mathcal{E}_p^* .

Pour cela, on va utiliser le théorème [81] déjà cité et le corollaire suivant :

Corollaire [82]

Soit $\mathbf{C} \in M_{n \times n}$ une matrice hermitienne, et r un nombre entier donné avec $1 \leq r \leq n$, et $\mathbf{u}_1, \dots, \mathbf{u}_r \in \mathbb{C}^n$, r vecteurs orthonormaux donnés.

Si

$$\hat{\mathbf{A}} = [\mathbf{u}_i^H \mathbf{C} \mathbf{u}_j]$$

Et les vecteurs propres de \mathbf{C} et $\hat{\mathbf{A}}$ sont rangés dans l'ordre croissant, on aura :

$$\lambda_k(\mathbf{C}) \leq \lambda_k(\hat{\mathbf{A}}) \leq \lambda_{k+n-r}(\mathbf{C}), \quad k = 1, 2, \dots, r$$

a) Convergence du rapport signal à bruit $\rho(t, t)$

A chaque instant t , et selon les équations (III.2) et (III.4), on peut écrire :

$$\mathbf{P}(t) = \begin{bmatrix} \mathbf{W}(t-1)^T \mathbf{C}_{xx}(t) \mathbf{W}(t-1) & \mathbf{y}(t) \\ \mathbf{y}^T(t) & a(t) \end{bmatrix} \quad (\text{III.8})$$

Avec $a(t)$ un scalaire

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

$$a(t) = \mathbf{u}(t)^T \mathbf{C}_{xx}(t) \mathbf{u}(t)$$

Et

$$\mathbf{y}(t) = \mathbf{W}(t-1)^T \mathbf{C}_{xx}(t) \mathbf{u}(t)$$

En utilisant le théorème, où $\mathbf{A} = \mathbf{W}(t-1)^T \mathbf{C}_{xx}(t) \mathbf{W}(t-1)$ et $\hat{\mathbf{A}} = \mathbf{P}(t)$, on a :

$$\lambda_1(t) \leq \lambda_2(t) \leq \dots \leq \lambda_p(t),$$

$$\hat{\lambda}_1(t) \leq \hat{\lambda}_2(t) \leq \dots \leq \hat{\lambda}_{p+1}(t)$$

Et

$$\hat{\lambda}_1(t) \leq \lambda_1(t) \leq \hat{\lambda}_2(t) \leq \dots \leq \hat{\lambda}_p(t) \leq \lambda_p(t) \leq \hat{\lambda}_{p+1}(t).$$

Cela veut dire, qu'à chaque itération t , on a :

$$\hat{\lambda}_{i+1}(t) \geq \lambda_i(t), \quad i = 1, \dots, p \quad (\text{III.9})$$

Si on considère la matrice diagonale $\mathbf{\Lambda}(t)$ composée par les p valeurs propres $\lambda_i(t)$ et la matrice diagonale $\hat{\mathbf{\Lambda}}_{pp}(t)$ qui contient les p plus grandes valeurs propres $\hat{\lambda}_i(t)$ associées aux vecteurs propres $\mathbf{U}(t)$ de $\mathbf{P}(t)$, on peut écrire :

$$\mathbf{U}(t)^T \mathbf{P}(t) \mathbf{U}(t) = \hat{\mathbf{\Lambda}}_{pp}(t)$$

On a,

$$\text{tr}(\hat{\mathbf{\Lambda}}_{pp}(t)) \geq \text{tr}(\mathbf{\Lambda}(t)) \quad (\text{III.10})$$

Vu que,

$$\text{tr}(\mathbf{\Lambda}(t)) = \text{tr}(\mathbf{W}(t-1)^T \mathbf{C}_{xx}(t) \mathbf{W}(t-1)) \quad (\text{III.11})$$

Si on remplace $\mathbf{C}_{xx}(t)$ par sa valeur selon l'équation (III.6) en considérant $\rho(t-1, t)$ donné par l'équation (III.5), on retrouve,

$$\text{tr}(\mathbf{\Lambda}(t)) = 0 \quad (\text{III.12})$$

Donc,

$$\text{tr}(\mathbf{U}(t)^T \underline{\mathbf{W}}(t)^T \mathbf{C}_{xx}(t) \underline{\mathbf{W}}(t) \mathbf{U}(t)) \geq 0$$

De l'équation (III.4), on trouve

$$\text{tr}(\mathbf{U}(t)^T \underline{\mathbf{W}}(t)^T \mathbf{C}_{xx}(t) \underline{\mathbf{W}}(t) \mathbf{U}(t)) \geq 0$$

Et selon l'équation (III.1), on peut écrire

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

$$\text{tr}(\mathbf{W}(t)^T \mathbf{C}_{xx}(t) \mathbf{W}(t)) \geq 0.$$

Si on remplace $\mathbf{C}_{xx}(t)$ par son expression (III.6), on trouve

$$\rho(t, t) - \rho(t - 1, t) \geq 0$$

Soit

$$\rho(t, t) \geq \rho(t - 1, t)$$

Quand les matrices de covariance ont convergé, $\mathbf{A}(t) = \mathbf{A}$ et $\mathbf{B}(t) = \mathbf{B}$ ne sont plus fonction de t , on peut écrire :

$$\rho(t - 1) \equiv \rho(t - 1, t) \quad \text{et} \quad \rho(t) \equiv \rho(t, t).$$

D'où,

$$\rho(t) \geq \rho(t - 1) \tag{III.13}$$

Donc, après convergence des matrices de covariances, l'évolution dans le temps du rapport signal à bruit $\rho(t)$ représente une suite croissante qui est bornée par les plus grandes valeurs propres de la matrice $\mathbf{B}^{-1}\mathbf{A}$. On peut dire alors que $\rho(t)$ converge vers une valeur ρ_c .

Après la convergence, on aura une suite $\{\mathbf{u}(t)\}$ qui nous conduira à la valeur ρ_c de $\rho(t)$.

b) Convergence du sous espace engendré par $\mathbf{W}(t)$

Après avoir démontré la convergence de $\rho(t)$, liée à une suite $\{\mathbf{u}(t)\}$ vers la valeur ρ_c , montrons la convergence de $\mathbf{W}(t)$.

On peut donc écrire $\mathbf{C}_{xx}(\rho_c) = \mathbf{A} - \rho_c \mathbf{B}$ et $\mathbf{C}_{xx} \mathbf{W}_c = \mathbf{W}_c \mathbf{D}_c$ avec

$$\rho_c = \frac{\text{tr}(\mathbf{W}_c^T \mathbf{A} \mathbf{W}_c)}{\text{tr}(\mathbf{W}_c^T \mathbf{B} \mathbf{W}_c)}$$

On utilise alors le corollaire avec $\hat{\mathbf{A}} = \mathbf{P}(t)$ et $\mathbf{C} = \mathbf{C}_{xx}(\rho_c)$

$\mathbf{C}_{xx}(\rho_c) \in M_{n \times n}$ est symétrique réelle, et comme dans l'équation (III.4)

$$\mathbf{P}(t) = \mathbf{W}(t)^T \mathbf{C}_{xx}(\rho_c) \mathbf{W}(t) = [\mathbf{w}_i^T(t) \mathbf{C}_{xx} \mathbf{w}_j],$$

Les valeurs propres de $\mathbf{C}_{xx}(\rho_c)$ et $\mathbf{P}_t(t)$, notées respectivement $\lambda_i(\mathbf{C}_{xx}(\rho_c))$ et $\lambda_i(\mathbf{P}(t))$ sont rangées dans l'ordre croissant :

$$\lambda_1(\mathbf{C}_{xx}) \leq \lambda_2(\mathbf{C}_{xx}) \leq \dots \leq \lambda_n(\mathbf{C}_{xx})$$

et

$$\lambda_1(\mathbf{P}(t)) \leq \lambda_2(\mathbf{P}(t)) \leq \dots \leq \lambda_{p+1}(\mathbf{P}(t))$$

Alors,

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

$$\lambda_k(\mathbf{C}_{xx}) \leq \lambda_k(\mathbf{P}(t)) \leq \lambda_{k+n-(p+1)}(\mathbf{C}_{xx}), \quad k = 1, \dots, p+1$$

aussi

$$\lambda_k^c \leq \hat{\lambda}_k(t) \leq \lambda_{k+n-(p+1)}^c, \quad k = 1, \dots, p+1.$$

En reprenant l'équation (III.9) à l'instant $t+1$, on trouve :

$$\hat{\lambda}_{k+1}(t+1) \geq \lambda_k(t+1)$$

avec $\lambda_k(t+1)$ les valeurs propres de $\underline{\mathbf{W}}(t)^T \mathbf{C}_{xx}(t) \underline{\mathbf{W}}(t) = \mathbf{P}(t)$.

D'où

$$\lambda_k(t+1) = \hat{\lambda}_{k+1}(t)$$

Alors,

$$\hat{\lambda}_{k+1}(t+1) \geq \hat{\lambda}_{k+1}(t), \quad k = 1, \dots, p$$

$\hat{\lambda}_k(t)$ est une suite croissante bornée par $\lambda_{k+n-(p+1)}^c$ avec $k = 2, \dots, p+1$.

Montrons par la suite que $\mathbf{W}(t) = \mathbf{W}(t-1)$

On rappelle qu'on se met dans les conditions suivantes :

\mathbf{A} et \mathbf{B} ont convergé, le rapport signal à bruit $\rho(t)$ a aussi convergé vers $\rho_c \neq \rho^*$, et les $\hat{\lambda}_k(t)$ ont convergé vers $\hat{\lambda}_k^c$, $k = 2, \dots, p+1$.

$$\mathbf{P}(t) = \begin{bmatrix} \mathbf{W}^T(t-1) \mathbf{C}_{xx} \mathbf{W}(t-1) & \mathbf{y}(t) \\ \mathbf{y}^T(t) & a(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Lambda}(t) & \mathbf{y}(t) \\ \mathbf{y}^T(t) & a(t) \end{bmatrix} \quad (\text{III.14})$$

Vu que les valeurs propres ont convergé, on a

$$\boldsymbol{\Lambda}(t) = \hat{\boldsymbol{\Lambda}}_p(t-1) = \hat{\boldsymbol{\Lambda}}_p^c = \begin{bmatrix} \hat{\lambda}_{p+1}^c & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \hat{\lambda}_2^c \end{bmatrix}.$$

Donc

$$\mathbf{P}(t) \mathbf{U}(t) = \mathbf{U}(t) \begin{bmatrix} \hat{\boldsymbol{\Lambda}}_p^c & \mathbf{0} \\ \mathbf{0}^H & \hat{\lambda}_1(t) \end{bmatrix},$$

avec, nécessairement $\hat{\lambda}_1(t) < \hat{\lambda}_2$. On a

$$\text{tr}(\mathbf{P}(t)) = \text{tr}(\hat{\boldsymbol{\Lambda}}_p^c) + \hat{\lambda}_1(t),$$

Et selon l'équation (III.14), on a aussi

$$\text{tr}(\mathbf{P}(t)) = \text{tr}(\hat{\boldsymbol{\Lambda}}_p^c) + a(t),$$

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

avec

$$a(t) = \hat{\lambda}_1(t).$$

Si on décompose la matrice $\mathbf{U}(t)$ de dimension $(p+1) \times (p+1)$ qui contient les vecteurs propres de $\mathbf{P}(t)$:

$$\mathbf{U}(t) = \begin{bmatrix} \mathbf{U}_s(t) \\ \mathbf{u}_{p+1}^T(t) \end{bmatrix},$$

On pourrait écrire

$$\mathbf{y}^T(t)\mathbf{U}_s(t) + a(t)\mathbf{u}_{p+1}^T(t) = \hat{\lambda}_1(t)\mathbf{u}_{p+1}^T(t).$$

D'où,

$$\mathbf{y}^T(t)\mathbf{U}_s(t) = \mathbf{0}^T. \quad (\text{III.15})$$

De plus,

$$\mathbf{U}(t)\mathbf{U}^T(t) = \mathbb{I}_{p+1} = \begin{bmatrix} \mathbf{U}_s(t)\mathbf{U}_s^T(t) & \mathbb{O} \\ \mathbb{O}^T & \mathbf{u}_{p+1}^T(t)\mathbf{u}_{p+1}(t) \end{bmatrix}.$$

Alors

$$\mathbf{u}_{p+1}^T(t)\mathbf{u}_{p+1}(t) = 1,$$

et

$$\mathbf{U}_s(t)\mathbf{U}_s^T(t) = \mathbb{I}_p.$$

Si on multiplie l'équation (III.15) par $\mathbf{U}_s^T(t)$, on trouve

$$\mathbf{y}^T(t)\mathbf{U}_s(t)\mathbf{U}_s^T(t) = \mathbb{O}^T.$$

D'où,

$$\mathbf{y}(t) = \mathbf{0}.$$

La matrice $\mathbf{P}(t)$ est diagonale et $\mathbf{U}(t) = \mathbb{I}_{p+1}$, et $[0 \dots 0 \ 1]^T$ est associé à la plus petite valeur propre de $\mathbf{P}(t)$, et

$$\mathbf{W}(t) = \underline{\mathbf{W}}(t)\mathbf{U}_p(t) = \mathbf{W}(t-1),$$

On le notera \mathbf{W}_c .

Poursuite du sous-espace optimal

III.4 L'algorithme CSMFST

Donc les valeurs propres de $\mathbf{P}(t)$ forment une matrice \mathbb{I}_{p+1} et on a

$$\mathbf{W}^T \mathbf{C}_{xx}(\rho_c) \mathbf{W} = \widehat{\Lambda}_{pp} \quad (\text{III.16})$$

D'où,

$$\rho_c = \frac{\text{tr}(\mathbf{W}^T \mathbf{A} \mathbf{W})}{\text{tr}(\mathbf{W}^T \mathbf{B} \mathbf{W})}$$

La matrice \mathbf{W} décrit le même sous espace que \mathbf{W}_c et

$$\mathbf{W} \mathbf{W}^T = \mathbf{W}_c \mathbf{W}_c^T \neq \mathbf{W}^* \mathbf{W}^{*T}$$

La démonstration ici, a été faite pour une suite $\{\mathbf{u}(t)\}$ donnée qui vérifie

$$\forall t, \mathbf{u}(t) \in \mathbf{W}_\perp \quad \text{et} \quad \mathbf{u}(t) \in \mathbf{W}_\perp^*.$$

Elle est donc réalisée pour les $\{\mathbf{x}_A(t)\}$ et les $\{\mathbf{x}_B(t)\}$ qui explorent une partie de $\text{Span}(\mathbf{W}^*)$.

Admettons maintenant que le dernier $\mathbf{u}(t) \in \mathbf{W}^*$ et $\mathbf{u}(t) \in \mathbf{W}_{c_\perp}$ et montrons que la nouvelle valeur de $\rho(t)$ est supérieure à ρ_c .

Selon l'équation (III.12)

$$\text{tr}(\mathbf{\Lambda}(t)) = \text{tr}(\widehat{\Lambda}_p^c) = 0$$

Et vu que

$$\mathbf{P}(t) = \begin{bmatrix} \mathbf{\Lambda}(t) = \widehat{\Lambda}_p^c & \mathbf{y}(t) \\ \mathbf{y}^T(t) & a(t) \end{bmatrix}$$

Avec les valeurs propres de la matrice $\widehat{\Lambda}(t) = \begin{bmatrix} \widehat{\Lambda}_p(t) & 0 \\ 0^T & \hat{\lambda}_1(t) \end{bmatrix}$,

On a

$$\text{tr}(\mathbf{P}(t)) = a(t) = \text{tr}(\widehat{\Lambda}_p(t)) + \hat{\lambda}_1(t).$$

Alors

$$\text{tr}(\widehat{\Lambda}_p(t)) \neq 0$$

Mais d'après l'équation (III.10),

$$\text{tr}(\widehat{\Lambda}_p(t)) \geq \text{tr}(\widehat{\Lambda}_p^c),$$

On a donc

$$\text{tr}(\widehat{\Lambda}_p(t)) > 0.$$

A partir de l'équation suivante :

$$\mathbf{W}^T(t)\mathbf{C}_{xx}(\rho_c)\mathbf{W}(t) = \mathbf{U}_p^T(t)\mathbf{P}(t)\mathbf{U}_p(t) = \widehat{\Lambda}_p(t),$$

On écrit

$$\frac{\text{tr}(\mathbf{W}^T(t)\mathbf{C}_{xx}(\rho_c)\mathbf{W}(t))}{\text{tr}(\mathbf{W}^T(t)\mathbf{B}\mathbf{W}(t))} = \frac{\text{tr}(\widehat{\Lambda}_p(t))}{\text{tr}(\mathbf{W}^T(t)\mathbf{B}\mathbf{W}(t))},$$

avec

$$\mathbf{C}_{xx}(\rho_c) = \mathbf{A} - \rho_c\mathbf{B}.$$

D'où

$$\rho(t) - \rho_c = \frac{\text{tr}(\widehat{\Lambda}_p(t))}{\text{tr}(\mathbf{W}^T(t)\mathbf{B}\mathbf{W}(t))} > 0.$$

Donc,

$$\rho(t) > \rho_c.$$

En conclusion, on peut dire que pour une suite $\{\mathbf{u}(t)\} \in \text{Span}(\mathbf{W}^*)$, le rapport signal à bruit $\rho(t)$ converge vers sa valeur optimale ρ^* .

Par conséquent,

$$\mathbf{W}^T(t)\mathbf{C}_{xx}(\rho^*)\mathbf{W}(t) = \widehat{\Lambda}_p(\rho^*)$$

et $\mathbf{W}(t)\mathbf{W}(t)^T = \mathbf{W}(t)^*\mathbf{W}(t)^{*T}$ puisque $\mathbf{W}(t)$ diagonalise $\mathbf{C}_{xx}(\rho^*)$ et $\widehat{\Lambda}_p(\rho^*)$ renferme les plus grandes valeurs propres.

Nous avons développé [annexe 3] le calcul des vecteurs propres de $\mathbf{M}(t)$ afin d'éviter la diagonalisation de cette matrice.

III .5 Conclusion

Dans ce chapitre, nous avons rappelé le problème classique d'estimation du sous-espace principal, de dimension connue, dans le cas d'un signal d'intérêt noyé dans un bruit blanc. Une technique bien connue de détermination du sous-espace principal consiste à faire des décompositions de type SVD ou EVD : toutefois, pour les systèmes non stationnaires (variant dans le temps), les variations de la matrice de corrélation entraînent une nécessaire mise à jour permanente de l'estimation de celle-ci. On est alors confronté à un problème de poursuite de sous-espace, ou de tracking.

Le calcul complet et systématique des décompositions devient rapidement prohibitif et incompatible avec le temps réel. Aussi, très naturellement, les chercheurs se sont ingéniés à proposer des techniques donnant une solution certes approchante mais beaucoup moins

gourmande en temps de calcul. A cette fin, de nombreux algorithmes ont été proposés pour l'estimation et la poursuite du sous-espace principal.

Par la suite nous avons rappelé quelques algorithmes de tracking tels que ceux de la famille PAST, ceux basés sur le principe de puissance itérée, ceux basés sur la méthode de calcul itératif SVD et ceux de type *Subspace Projection* (SP).

Cette dernière catégorie nous a aidés dans l'implémentation de notre algorithme.

Nous avons proposé un algorithme adaptatif appelé *Constrained Stochastic Matched Filter Subspace Tracking* (CSMFST), en nous inspirant des méthodes de poursuite du sous-espace principal, tels que les algorithmes de type *Subspace Projection* (SP), afin de faire la poursuite du sous-espace optimal en temps réel, dans le cadre de la détection de signaux dont les propriétés statistiques, voire le RSB, varient dans le temps.

Notre objectif est donc différent puisqu'il s'agit de trouver, en temps réel, le sous-espace optimal en termes de rapport signal à bruit dans un contexte non stationnaire, ce qui induit une variation, à chaque instant de mesure, non seulement du sous-espace recherché mais également du rapport signal à bruit.

Bien entendu, pour ce faire, les principes de l'algorithme CSMF ont été largement utilisés, mais transformés en s'inspirant des algorithmes de type SP.

A la fin du chapitre, nous avons démontré que cet algorithme converge et que l'on retrouve, après convergence de ce nouvel algorithme, *a minima* dans le cas stationnaire, le sous-espace optimal pouvant être calculé avec le CSMF.

Il faut noter ici, que notre objectif était de montrer la faisabilité de la poursuite et de trouver un algorithme en ce sens ; nous n'avons pas cherché à optimiser la complexité algorithmique : dans les exemples présentés on s'est inspiré du principe du GYAST qui est un algorithme de type SP moins complexe que l'algorithme SP-1.

Notre algorithme peut être ramené à la même complexité que l'algorithme GYAST qui est de $3np$ opérations par mise à jour ce qui représente le plus faible coût parmi tous les algorithmes de tracking rapides.

Chapitre IV :
Applications : simulations et
signaux réels

IV.1 Introduction

Dans ce chapitre, on teste l'algorithme CSMFST dans les cas stationnaire et non stationnaire.

IV.2 Conditions de simulations

Dans toutes les simulations qui suivent, on prend comme valeurs $n = 8$ et $p = 3$.

Génération des réalisations :

Pour chaque scénario on fixe les valeurs propres ainsi que les vecteurs propres des différentes matrices \mathbf{A} et \mathbf{B} .

On crée des signaux stationnaires par zone, donc constitués de plusieurs zones stationnaires (instants différents).

$\lambda_{i,j}^S$ et $\lambda_{i,j}^B$ représentent respectivement les valeurs propres du signal et du bruit, où i représente le numéro de la valeur propre et j le numéro de la zone stationnaire.

Les vecteurs propres associés sont notés respectivement par \mathbf{V}^S et \mathbf{V}^B .

Les matrices contenant les vecteurs propres du signal et du bruit sont orthogonalisées à l'aide du procédé d'ortho-normalisation de Gram-Schmidt.

A chaque instant t , on génère une réalisation du signal $\mathbf{x}_A(t)$ par

$$\mathbf{x}_A(t) = \sum_{i=1}^n a_i(t) \mathbf{v}_i$$

où $a_i(t)$ est une variable aléatoire gaussienne centrée de variance λ_i la valeur propre associée au vecteur propre \mathbf{v}_i de \mathbf{A} .

On fait évidemment de même pour $\mathbf{x}_B(t)$.

On suppose, que dans la phase d'apprentissage, on accède à un certain nombre d'échantillons du signal \mathbf{x}_A et du bruit \mathbf{x}_B séparément : ceci nous permettra d'estimer à partir de leurs matrices de covariances, le sous-espace optimal.

La détection est réalisée par la suite sur le reste des échantillons.

Comparaison des sous-espaces :

Le critère qualitatif qui va nous permettre de juger la performance de notre algorithme est la **distance** en dB [66] [83] entre les deux sous-espaces E_1 et E_2 avec $E_1 = \text{Span}\{\mathbf{W}_1\}$ et $E_2 = \text{Span}\{\mathbf{W}_2\}$ qu'on appellera **erreur de projection** définie par :

$$d(E_1, E_2) = 10 \log_{10} \|\mathbf{\Pi}_1 - \mathbf{\Pi}_2\|_F$$

où $\mathbf{\Pi}_i = \mathbf{W}_i \mathbf{W}_i^H$ et $\|\cdot\|_F$ est la norme de Frobenius définie par $\|\mathbf{C}\|_F = \text{tr}(\mathbf{C}\mathbf{C}^*)^{\frac{1}{2}} = \sqrt{\sum_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}} |\mathbf{C}_{ij}|^2}$

avec \mathbf{C}^* désigne la matrice adjointe de \mathbf{C} .

Un autre critère, équivalent, est l'angle principal θ entre les sous-espaces, donné par :

$$\theta(E_1, E_2) = \text{ArcSinus}(\|\mathbf{\Pi}_2 - \mathbf{\Pi}_1(\mathbf{\Pi}_1^H \mathbf{\Pi}_2)\|).$$

IV.3 Performances du CSMFST dans le cas stationnaire

Dans cette partie, on compare, à chaque date t , le sous-espace $E_p(t)$ retourné par l'algorithme CSMFST avec le sous-espace $E_p^*(t)$ retourné par l'algorithme CSMF puis avec le sous-espace théorique ξ_p calculé directement à partir de l'ensemble des échantillons du signal et du bruit.

Puisque, dans cette partie, les tests sont réalisés dans un environnement stationnaire, le facteur d'oubli β est choisi égal à 1.

Dans toutes les simulations qui suivent, on observe les variations instantanées de la distance et du rapport signal-à-bruit. Ceci nous permettra de suivre l'évolution de l'algorithme en temps réel.

IV.3.1 Premier scénario : cas stationnaire, bruit blanc

► Dans la première simulation [Annexe 5 scénario1], il n'y a qu'une seule zone et le bruit est blanc. On construit des échantillons du signal $\mathbf{x}_A(t)$ à partir d'une matrice de covariance \mathbf{A} dont trois des valeurs propres $\lambda_{i,1}^S$ sont supérieures à 1 et les autres égales à 1. Le bruit est blanc obtenu en fixant une matrice de covariance de bruit scalaire \mathbf{B} avec toutes les valeurs propres $\lambda_{i,1}^B = 1$.

Les figures 1a et b représentent respectivement les erreurs de projection, c'est-à-dire la distance entre $E_p(t)$ et $E_p^*(t)$ et entre $E_p(t)$ et ξ_p .

Les résultats observés sur les courbes montrent que l'algorithme CSMFST présente une bonne convergence car $E_p(t)$ est proche de $E_p^*(t)$ (-60 dB) et de ξ_p (-35 dB).

La figure 1c montre la différence entre le rapport signal à bruit théorique et celui calculé par l'algorithme CSMFST à chaque date t , nous pouvons remarquer qu'ils sont presque identiques.

Applications : simulations et signaux réels

Vu que le test correspond à un cas stationnaire avec un bruit blanc, nous pouvons également comparer le résultat obtenu avec un algorithme de type SP (NFYAST). La figure 2 présente l'erreur de projection entre les deux algorithmes.

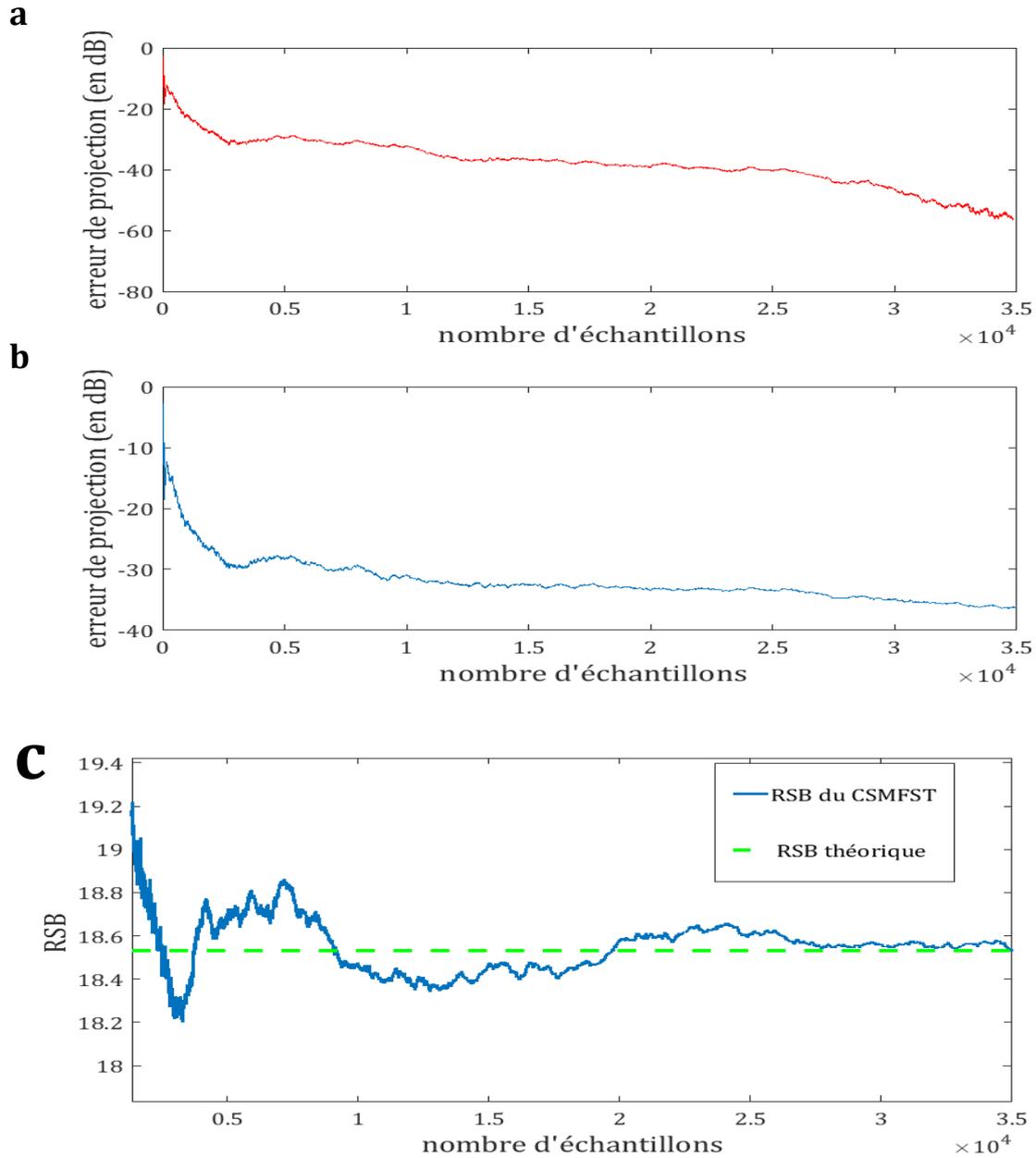


Figure IV.1. Performance de l'algorithme CSMFST dans le cas stationnaire : $d(E_p(t), E_p^*(t))$ (a), $d(E_p(t), \xi_p)$ (b) et RSB (c) [Annexe 5 scénario 1].

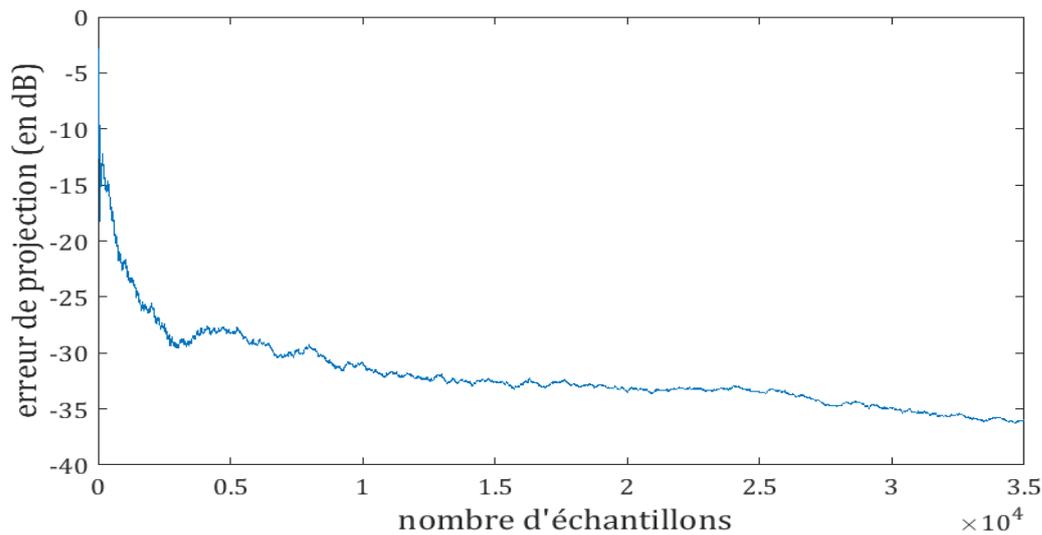


Figure IV.2. Comparaison dans le cas stationnaire entre un algorithme de type SP (NFYAST) et le CSMFST en présence d'un bruit blanc [Annexe 5 scénario 1].

IV.3.2 Deuxième scénario : cas stationnaire, bruit non blanc

► Dans un second scénario [Annexe 5 scénario 2], on garde la même matrice de covariance \mathbf{A} pour la partie signal et on construit la matrice \mathbf{B} qui a les mêmes vecteurs propres que \mathbf{A} ; seules les valeurs propres sont changées.

On donne ici les valeurs propres correspondantes aux matrices pour pouvoir calculer la valeur du RSB dans ce qui suit :

$$[\lambda_{i,1}^S] = [20,18,16,1,1,1,1,1]^H$$

pour le signal et

$$[\lambda_{i,1}^B] = [2,3,4,1,2,3,4,5]^H$$

pour le bruit.

Dans cet exemple, $\xi_p = \text{Span}(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_4)$ avec une valeur maximale de $\rho = 6.5$ même si une analyse rapide et fausse conduirait à croire que $\xi_p = \text{Span}(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)$ pour une valeur de $\rho = 6$.

Ceci s'explique par le fait qu'on doit maximiser $\frac{\sum \lambda_{i,1}^S}{\sum \lambda_{i,1}^B}$ et non pas $\sum \frac{\lambda_{i,1}^S}{\lambda_{i,1}^B}$.

En effet les gains $\frac{\lambda_{i,1}^S}{\lambda_{i,1}^B}$ valent respectivement 10, 6, 4, 1 pour $i = 1, 2, 3, 4$ donc $\sum \frac{\lambda_{i,1}^S}{\lambda_{i,1}^B}$ est maximal

pour $i = 1, 2, 3$, tandis que $\frac{\sum \lambda_{i,1}^S}{\sum \lambda_{i,1}^B}$ est maximal pour $i = 1, 2, 4$.

Applications : simulations et signaux réels

Les résultats correspondants à cet exemple sont donnés par la figure 3 et montrent que l'algorithme converge : $d(E_p(t), E_p^*(t))$ atteint -40 dB et $d(E_p(t), \xi_p)$ atteint -30dB et les valeurs des RSB (du CSMFST et théorique) se suivent.

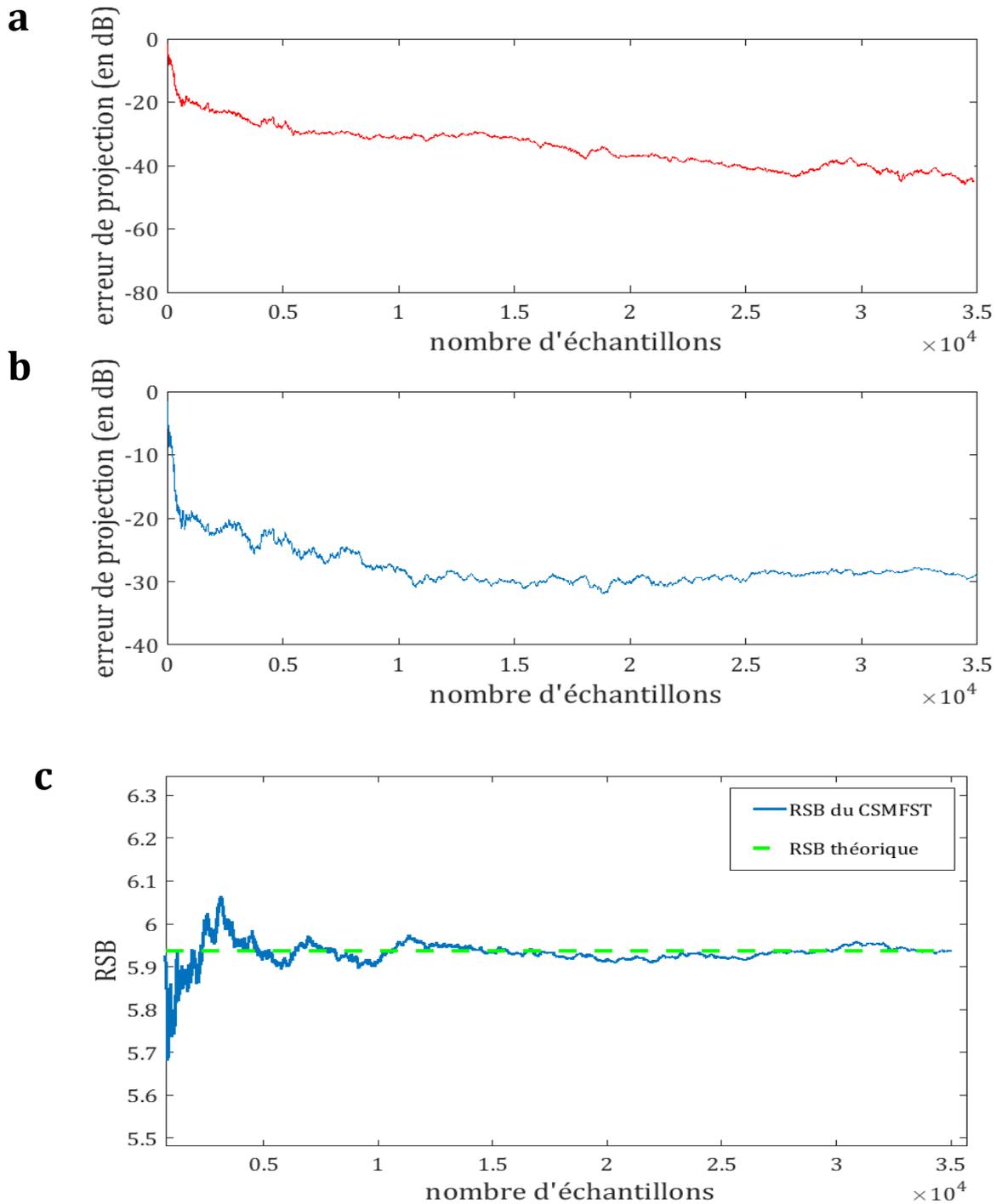


Figure IV.3. Performance de l'algorithme CSMFST dans le cas stationnaire : $d(E_p(t), E_p^*(t))$ (a), $d(E_p(t), \xi_p)$ (b) et RSB (c) [Annexe 5 scénario 2].

IV.4 Performances du CSMFST dans le cas non stationnaire

Dans cette seconde partie, on teste l'algorithme dans un environnement non stationnaire avec un facteur d'oubli β égal à 0.999.

IV.4.1 Troisième scénario : cas non stationnaire, bruit blanc

► Dans ce troisième scénario [Annexe 5 scénario 3], on crée quatre zones stationnaires différentes.

Dans la première zone, les matrices de covariances du signal et du bruit (blanc) sont identiques à celles du premier scénario. Pour chaque nouvelle zone, on échange un vecteur propre du sous-espace principal du signal avec un autre vecteur propre choisi en dehors du sous-espace principal : ceci nous permettra d'imposer un changement brusque des matrices et donc du sous-espace optimal et de forcer l'algorithme à s'adapter à cette nouvelle situation.

En répétant la même procédure dans chaque zone, on obtient quatre zones stationnaires qui, mises bout à bout, constituent un signal non stationnaire c'est-à-dire soumis à des variations brusques.

Ce type de scénario est plus pénalisant qu'un scénario où les transitions, au lieu d'être instantanées, seraient à évolution lente dans le temps.

C'est par ailleurs ce qui est fait dans de nombreux travaux (changements brusques de fréquence par exemple).

Les valeurs propres restent inchangées, le rapport signal à bruit reste donc constant. La figure 4b représente la distance $d(E_p(t), E_p^*(t))$.

On remarque que l'algorithme CSMFST présente une bonne convergence puisque le sous-espace résultant est assez proche de $E_p^*(t)$, en effet, la puissance d'erreur de projection atteint presque -30 dB. La même figure montre que nous avons des pics obtenus exactement au début de chaque zone en raison du changement de sous-espace.

La figure 4a montre la différence entre le rapport signal à bruit théorique et estimé. On constate également des variations de ce RSB estimé au début de chaque nouvelle zone.

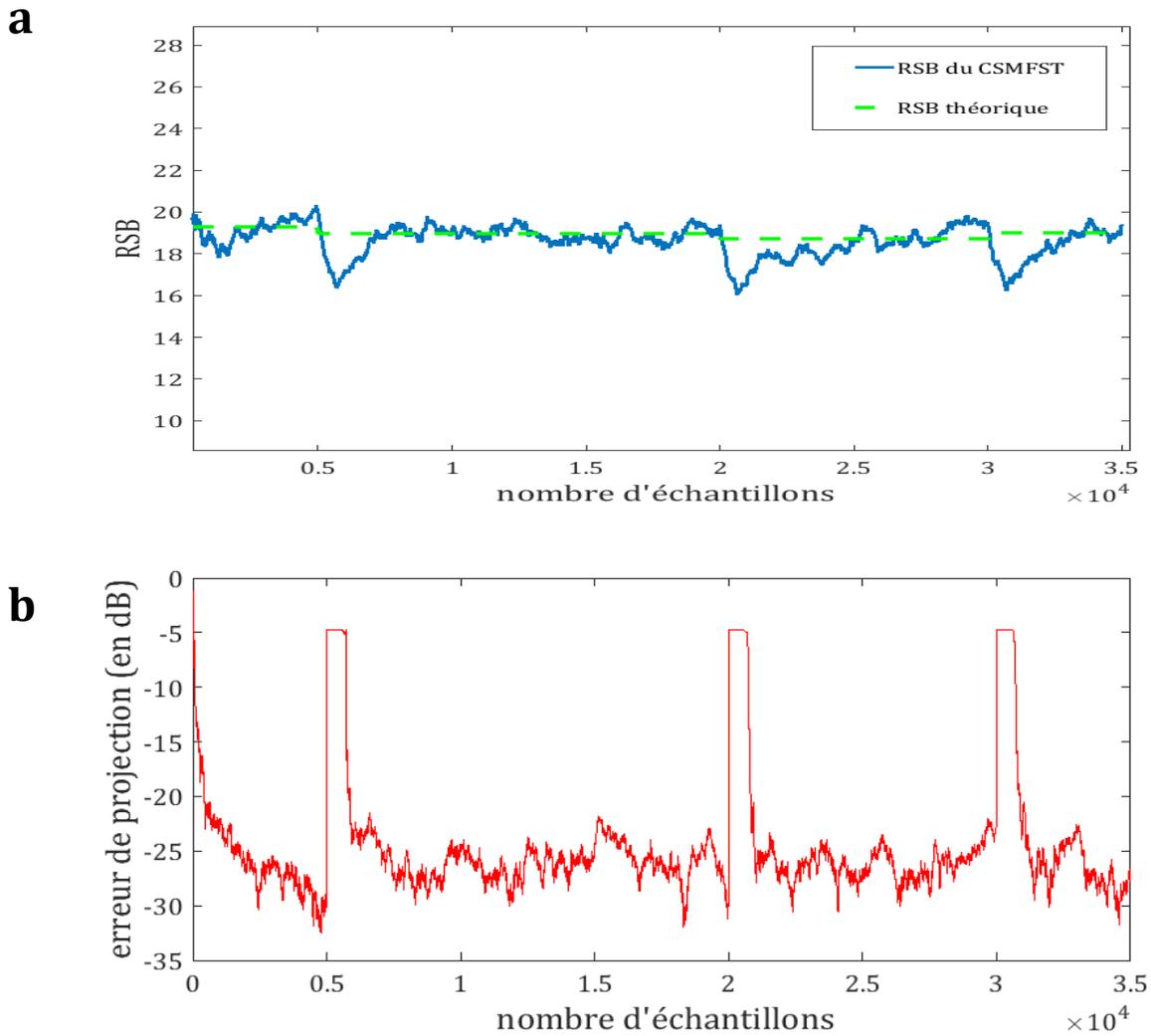


Figure IV.4. Performance du CSMFST dans le cas non stationnaire en termes de $d(E_p(t), E_p^*(t))$ (a) et du RSB (b) [Annexe 5 scénario 3].

Variation du facteur d'oubli β :

Pour le même exemple (scénario 3), on va varier les valeurs de β et tracer pour chaque valeur, les distance $d(E_p(t), E_p^*(t))$.

Le résultat obtenu est présenté dans la figure IV.5.

Applications : simulations et signaux réels

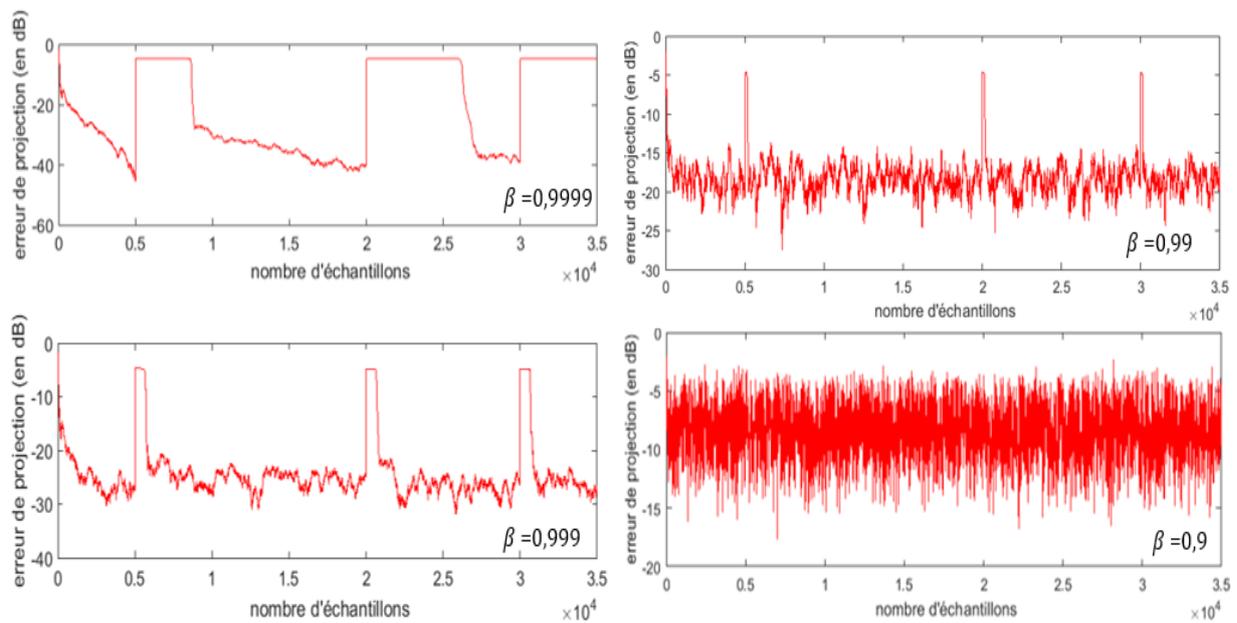


Figure IV.5 Distance $d(E_p(t), E_p^*(t))$ pour différentes valeurs de β

On remarque que la valeur de β a une influence directe sur la vitesse de convergence et donc sur la largeur des pics observés. Plus on est proche de 1, plus la convergence est lente mais plus on s'approche des valeurs données par le CSMF (dans le cas où $\beta = 0,9999$, on arrive à -40 dB de convergence mais lentement).

Dans le cas contraire, on converge très vite mais pas assez (pour $\beta = 0,9$, on n'atteint que -15 dB mais très rapidement).

Ceci veut dire qu'il faut trouver un compromis entre la vitesse de convergence et le degré de convergence qu'on arrive à atteindre.

C'est pour cette raison qu'on choisit un facteur de 0.999 dans nos simulations.

IV.4.2 Quatrième scénario : cas non stationnaire

► Pour le quatrième scénario [Annexe 5 scénario 4], nous créons quatre zones dans lesquelles nous modifions pour chacune d'entre elles, le sous-espace principal en changeant à la fois les valeurs et les vecteurs propres associés à la matrice de covariance du signal. Le bruit est non blanc (sauf dans la première et la 3^{ème} zone).

Le rapport signal à bruit varie d'une zone à l'autre. Le résultat donné par la figure IV.6, montre que l'erreur de projection est de l'ordre de -30 dB et que le RSB $\rho(t)$ est bien estimé.

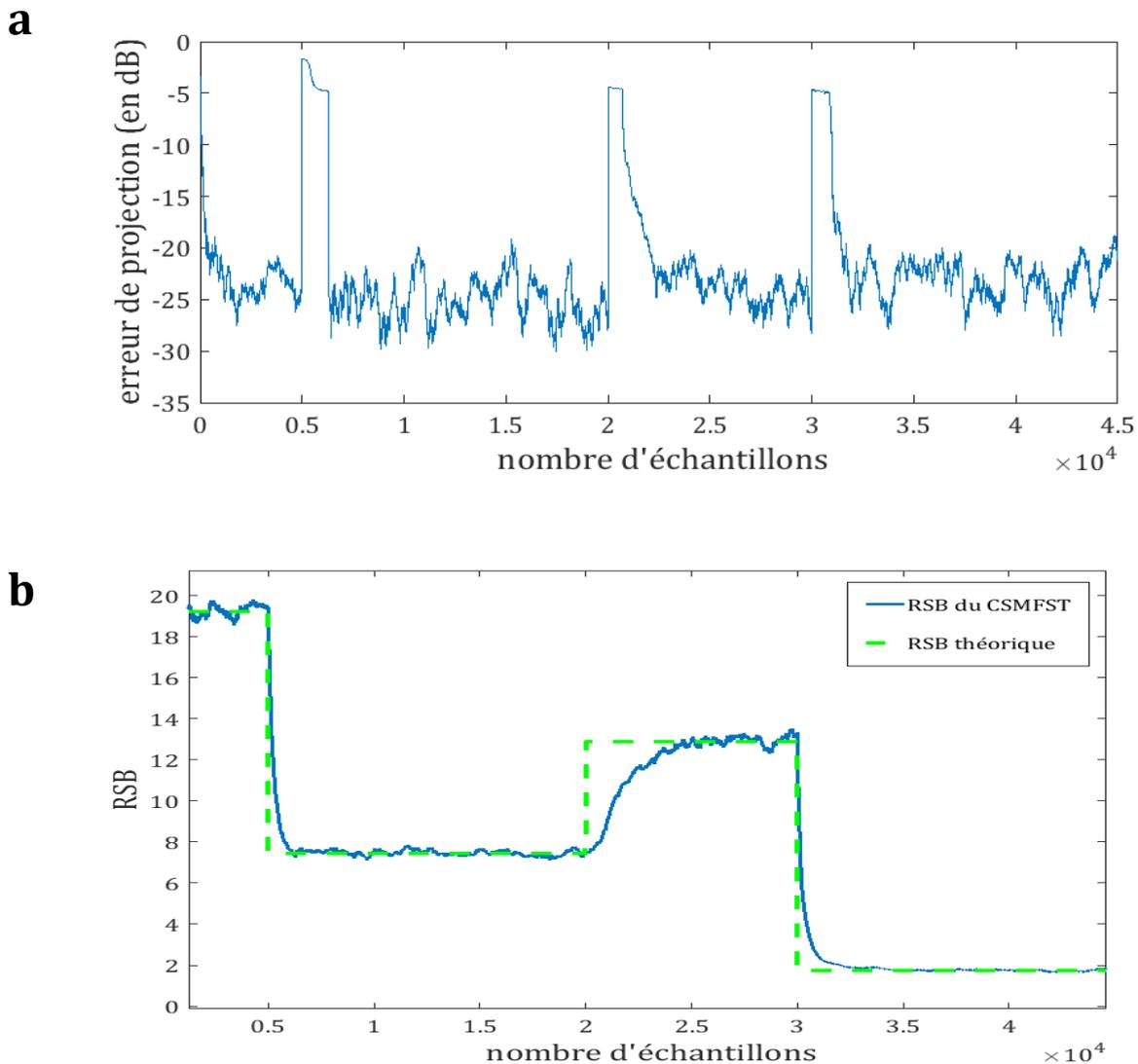


Figure IV.6. Performance de l'algorithme CSMFST dans le cas non stationnaire en termes de distance $d(E_p(t), E_p^*(t))$ (a) et du RSB (b) [Annexe 5 scénario 4].

IV.4.3 Cinquième scénario : cas non stationnaire, bruit non blanc

► Dans le cinquième exemple [Annexe 5 scénario 5], nous conservons les mêmes valeurs propres et vecteurs propres du signal que l'exemple précédent.

Le bruit est stationnaire non-blanc.

Les résultats de la figure 7 montrent que l'algorithme CSMFST présente une bonne convergence (-25 dB) même dans un environnement non stationnaire.

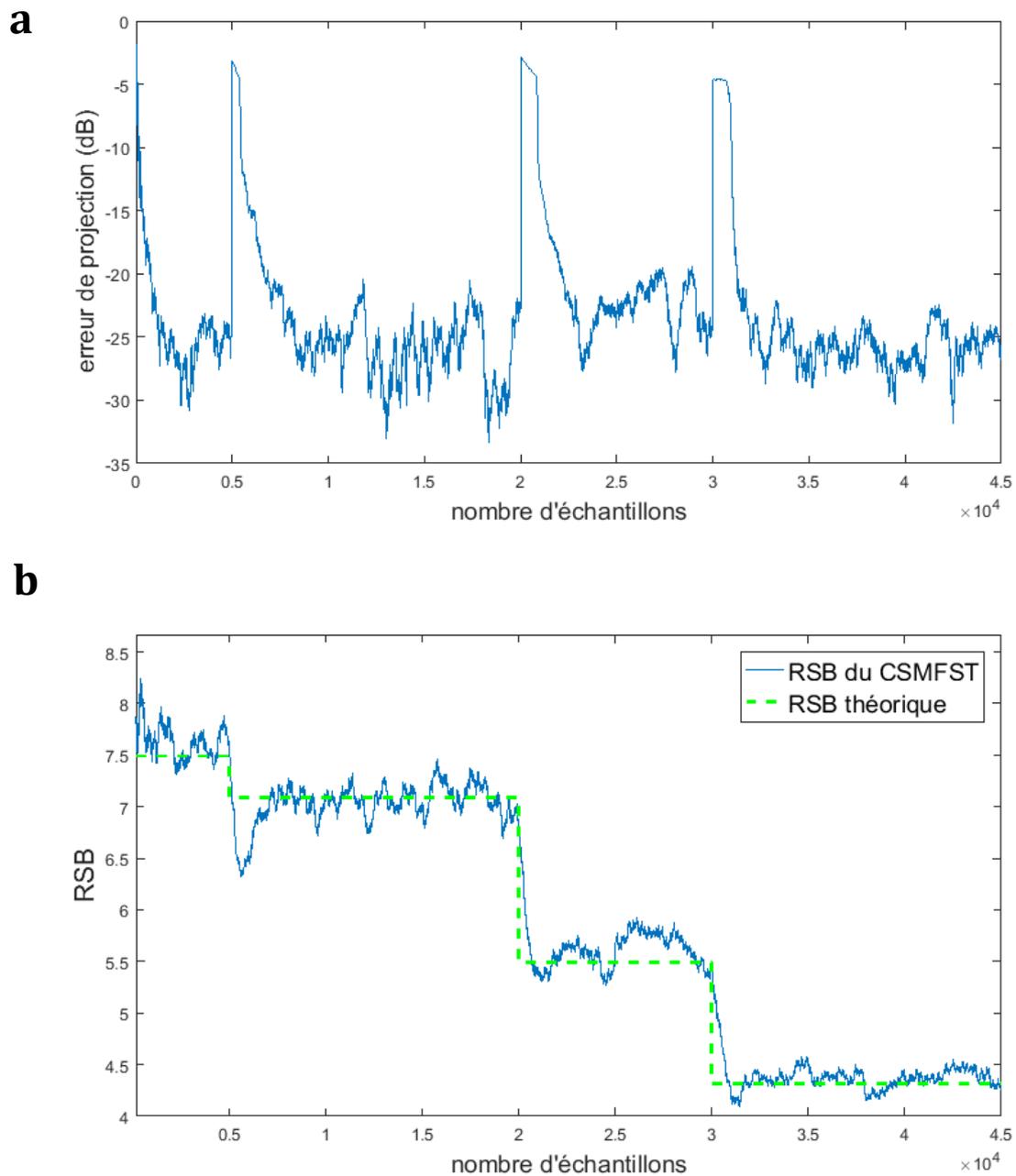


Figure IV.7. Performance de l'algorithme CSMFST dans le cas non stationnaire en termes de distance $d(E_p(t), E_p^*(t))$ (a) et RSB (b) [Annexe 5 scénario 5].

IV.5 Version modifiée du CSMFST : modélisation additive (signal plus bruit)

En pratique, la matrice de covariance du bruit est estimée *a priori* par un modèle donné.

Applications : simulations et signaux réels

Nous proposons de réaliser des scénarios dans lesquels le bruit est stationnaire de matrice de covariance estimée *a priori*. Dans ces simulations, les observations reçues seront, en fonction des scénarios, soit des échantillons du signal soit la superposition additive des échantillons du signal et du bruit.

Nous présentons les modifications apportées dans le pseudo-code de l'algorithme CSMFST de la table 1 qui sont liées au fait que la matrice de covariance du bruit $\mathbf{B}(t)$ est fixée à \mathbf{B} dans l'algorithme :

- Puisqu'on ne reçoit plus d'échantillons du bruit, $\mathbf{x}_B(t) = 0$ donc $\mathbf{e}_B(t) = 0$,
- $\mathbf{M}(t) = \mathbf{e}_A(t)\mathbf{e}_A^H(t)$,
- $\mathbf{u}(t) = \mathbf{e}_A(t)$.
- La matrice du bruit \mathbf{B} , est divisée par le nombre d'échantillons.

► Dans le premier cas [Annexe 5 scénario 6], l'algorithme CSMFST reçoit uniquement les échantillons d'un signal non stationnaire (le même que le scénario précédent). Les résultats présentés dans la figure 8 montrent la bonne convergence de l'algorithme (-30 dB) en termes de puissance d'erreur de projection et on trouve approximativement la valeur théorique du rapport signal sur bruit.

L'importance de ce scénario consiste à vérifier le bon fonctionnement de la version classique de l'algorithme avant d'appliquer la version orientée application et de donc comparer les deux résultats.

► Dans un autre scénario [Annexe 5 scénario 7], on conserve les mêmes conditions que le scénario précédent, la seule différence étant que les échantillons observés sont la superposition additive du signal et du bruit.

Les résultats sont présentés à la figure 9 et montrent que l'algorithme converge avec une puissance d'erreur de projection d'environ -25 dB.

On retrouve également les valeurs précédentes du rapport signal-bruit auquel on ajoute un dans chaque zone.

En effet, dans le scénario 6, on note $\mathbf{x}_A(t)$ les échantillons du signal, alors $\mathbf{A} = \mathbf{C}$ est la matrice de corrélation du signal et \mathbf{B} est la matrice de covariance du bruit stationnaire.

Dans le scénario 7, nous avons la superposition additive du signal et du bruit, si nous considérons de plus que le signal et le bruit sont non corrélés, nous avons :

$$\mathbf{A} = \mathbf{C} + \mathbf{B}.$$

Pour les scénarios 6 et 7, on note respectivement \mathbf{W} optimal par \mathbf{W}_{prev} et \mathbf{W}_{new} et le rapport signal à bruit ρ par ρ_{prev} et ρ_{new} .

Applications : simulations et signaux réels

D'où

$$\rho_{new} = \frac{\text{tr}(\mathbf{W}_{new}^H (\mathbf{C} + \mathbf{B}) \mathbf{W}_{new})}{\text{tr}(\mathbf{W}_{new}^H \mathbf{B} \mathbf{W}_{new})} = \frac{\text{tr}(\mathbf{W}_{new}^H \mathbf{C} \mathbf{W}_{new})}{\text{tr}(\mathbf{W}_{new}^H \mathbf{B} \mathbf{W}_{new})} + 1.$$

La valeur maximale de ρ_{new} est atteinte lorsque

$$\frac{\text{tr}(\mathbf{W}_{new}^H \mathbf{C} \mathbf{W}_{new})}{\text{tr}(\mathbf{W}_{new}^H \mathbf{B} \mathbf{W}_{new})}$$

est maximal, c'est-à-dire pour $\mathbf{W}_{new} = \mathbf{W}_{prev}$. On a alors $\rho_{new} = \rho_{prev} + 1$.

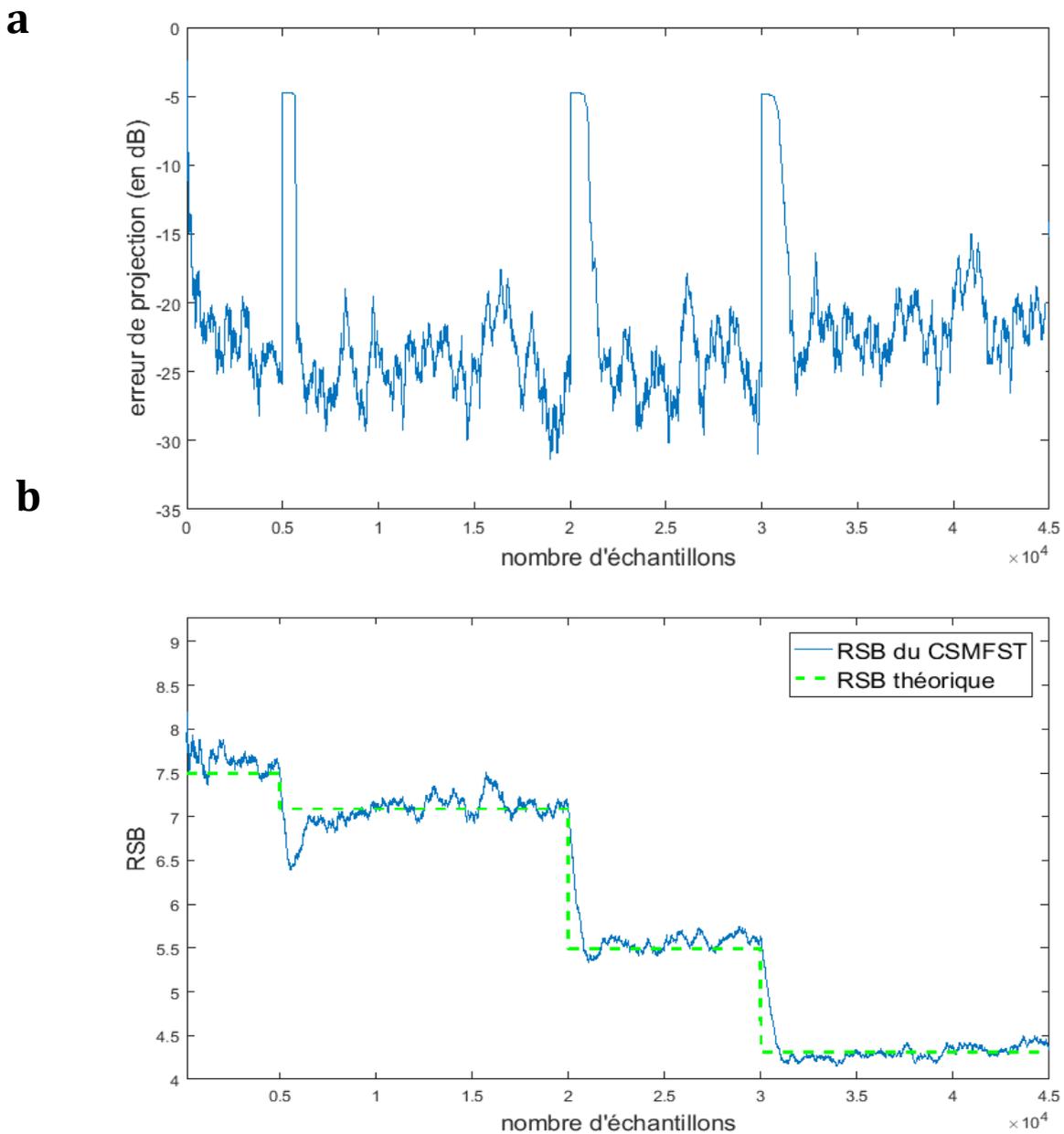
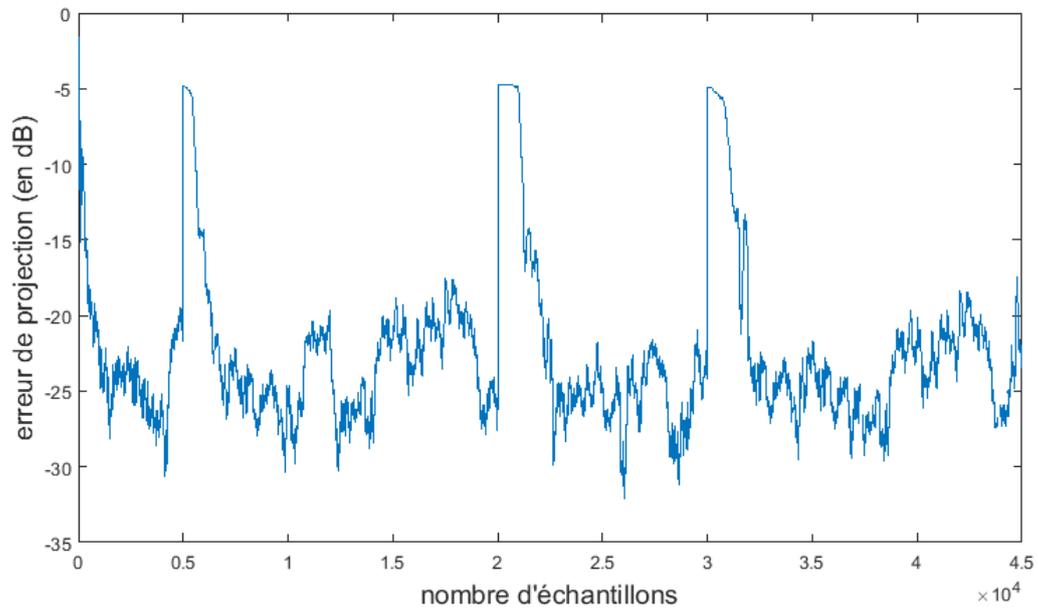


Figure. IV.8. Performance du CSMFST en termes de distance $d(E_p(t), E_p^*(t))$ (a) et de RSB (b) dans le cas d'une matrice de bruit fixée [Annexe 5 scénario 6]

a



b

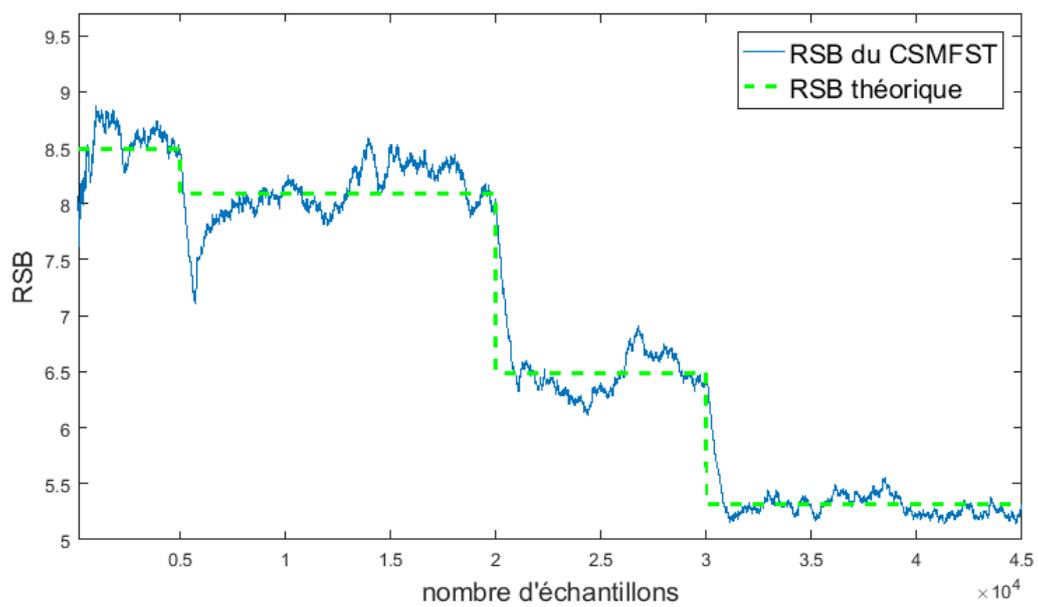


Figure IV.9. Performance du CSMFST en termes de distance $d(E_p(t), E_p^*(t))$ (a) et de RSB (b) dans le cas d'une matrice de bruit fixée [Annexe 5 scénario 7].

IV.6 Application aux signaux bioacoustiques réels

Dans cette partie, on teste l'algorithme CSMFST dans le cas non-stationnaire en prenant pour signal des clics d'orques et pour bruit le bruit de mer dont la puissance instantanée varie.

On possède des enregistrements sur 2 millions de points. Les signaux sont échantillonnés à une fréquence $F_s = 48 \text{ kHz}$. Sachant que la taille minimale des signaux qui nous permet d'observer le clic est de 100 points, on prend $n = 100$.

On choisit $p = 60$. Le but est donc de détecter les clics non stationnaires des orques avec le filtre CSMFST.

Les signaux bioacoustiques manipulés sont présentés par la figure IV.10.

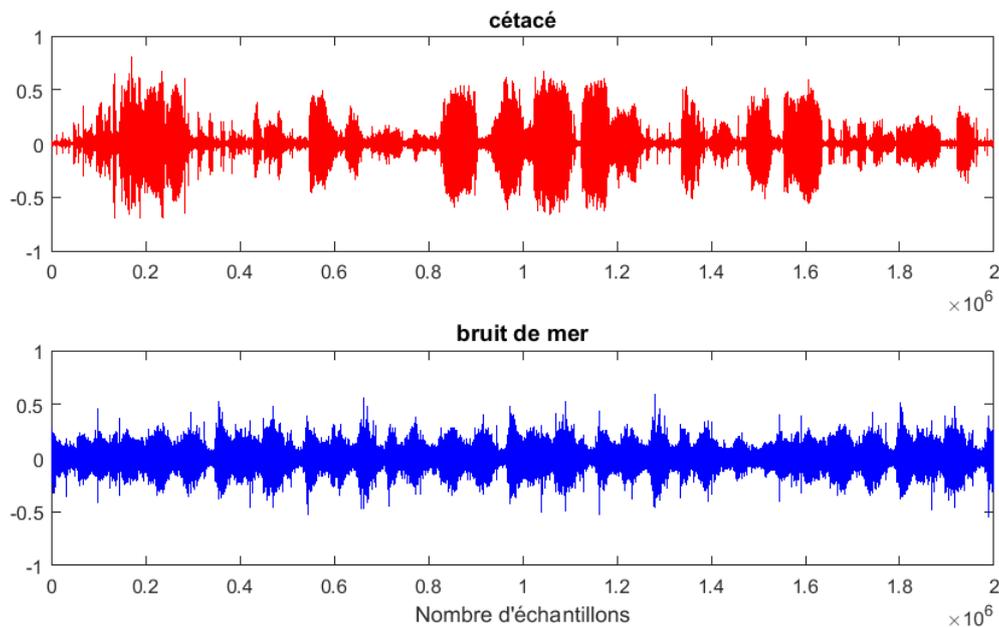


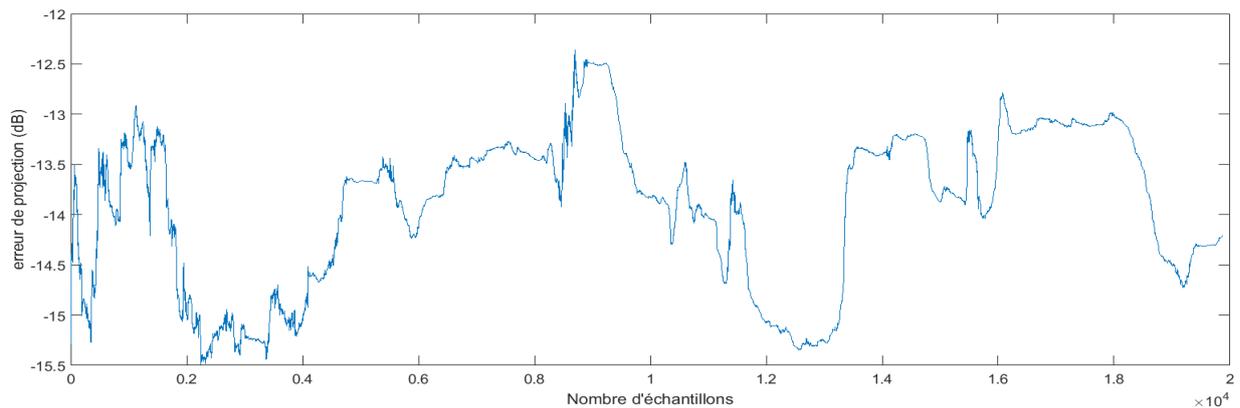
Figure IV.10. Signaux bioacoustiques

L'estimation des matrices de covariance de chaque processus a été réalisée avec 2000 réalisations (phase d'apprentissage) : ces signaux n'ont pas été utilisés dans la phase de détection.

Quelques résultats :

La figure IV.11 présente la distance $d(E_p(t), E_p^*(t))$ et le rapport $\rho(t)$ estimé par l'algorithme CSMFST à chaque étape sur les 20000 réalisations ainsi que celui estimé par l'algorithme CSMF $\rho^*(t)$.

a



b

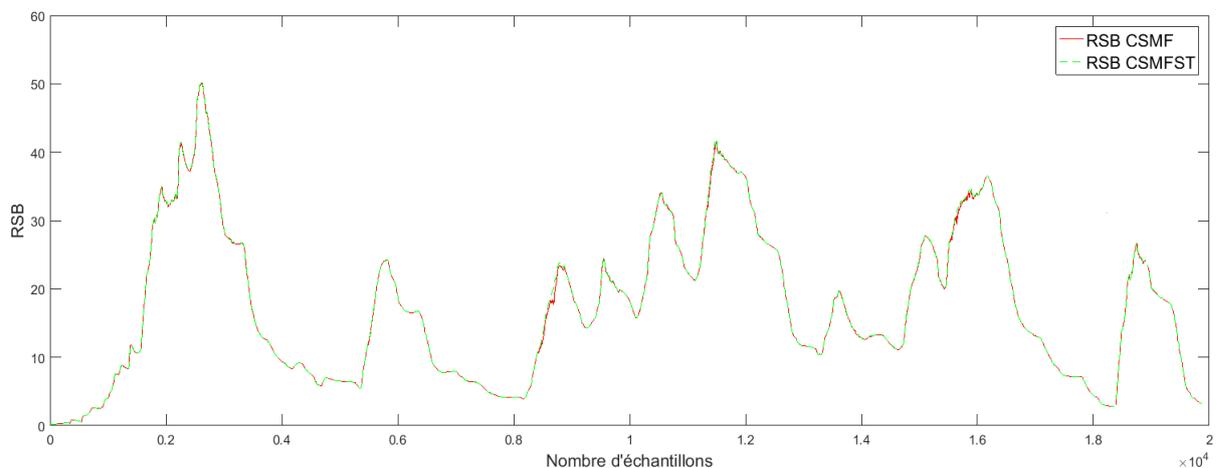


Figure IV.11. Performances du CSMFST en non stationnaire : $d(E_p(t), E_p^*(t))$ (a) et RSB (b)

Interprétations :

On remarque ici qu'il y a plusieurs variations du sous-espace et du RSB au cours du temps. Les deux algorithmes donnent le même résultat avec une précision de -15 dB sur la convergence et des RSB identiques.

Le problème ici, est que les signaux présentent une très grande non-stationnarité qui fait que l'algorithme ne peut pas converger rapidement. Ceci veut dire qu'il n'y a pas de zones stationnaires suffisamment longues (comme dans nos simulations) pour que le filtre ait le temps de converger.

Détection :

On construit par la suite, un signal test où on mélange les clics d'orques au bruit de mer. On crée ainsi 4 zones dans lesquelles on alterne les clics d'orques dans un bruit de mer et le bruit de mer seul. Le signal obtenu est présenté par la figure IV.12.

Applications : simulations et signaux réels

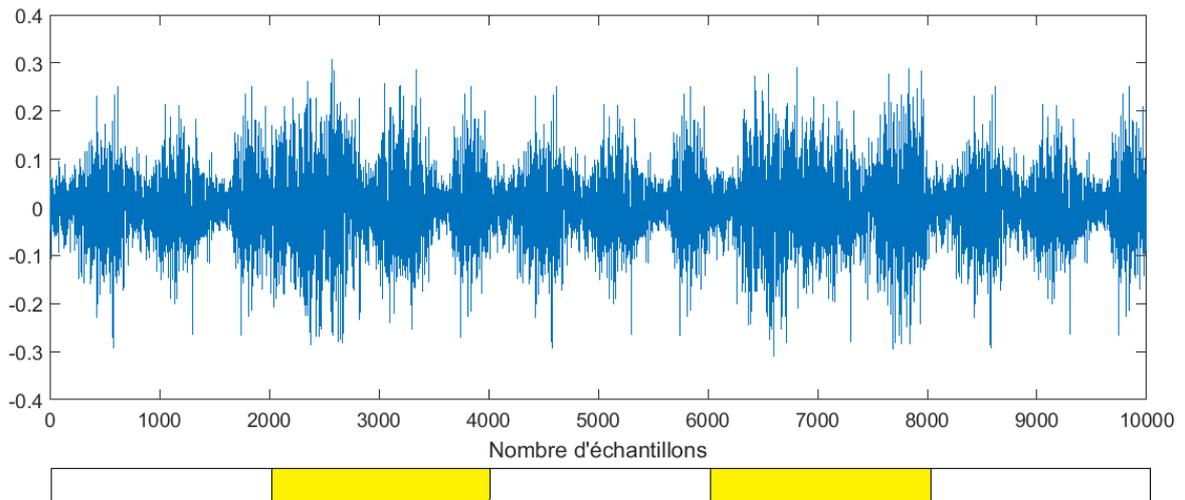


Figure IV.12. Signal test

Le résultat du filtrage multicritère est présenté sur la figure IV.13

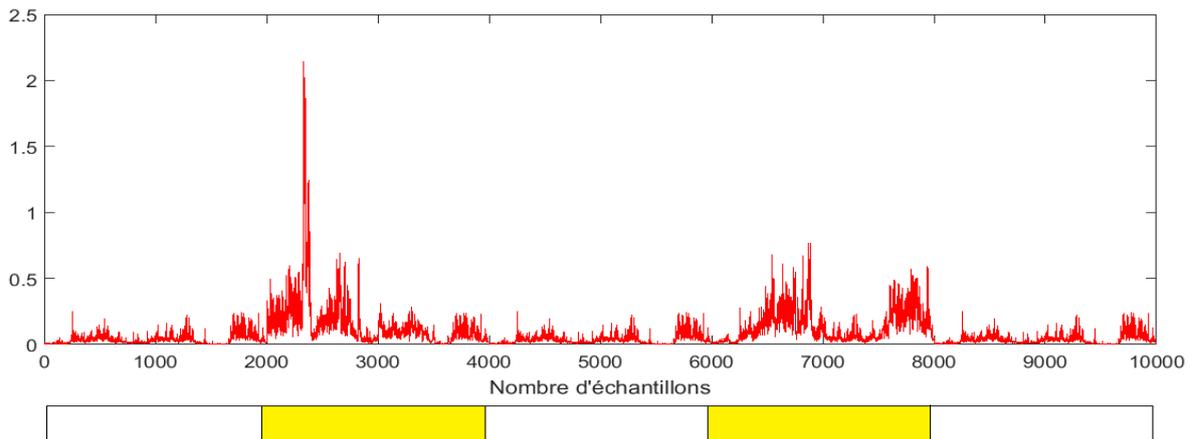


Figure IV.13. Résultat du filtrage multicritère du signal test : détection d'orques

On remarque qu'on détecte la majorité des clics d'orques dans le signal test.

Cet exemple illustre les limites de la méthode dans le cas de signaux insuffisamment échantillonnés : on note quand même que les tendances à la convergence restent observables.

On se propose dans la partie suivante, de simuler les clics de cétacés (pour limiter les non-stationnarités) en gardant un bruit de mer réel.

IV.7 Application au bruit de mer

Dans cette partie, on teste l'algorithme CSMFST dans le cas non-stationnaire en prenant pour signal une somme de trois sinusoides de fréquences différentes et pour bruit le son des vagues (bruit de mer réel) dont la puissance instantanée varie.

L'expression du signal est la suivante :

$$x(k) = 0.4 \cos(0.08 \times 2\pi k) + 0.5 \cos(0.2 \times 2\pi k) + 0.3 \cos(0.35 \times 2\pi k)$$

où k représente le temps ($t_k = k \Delta t$). On a donc $\text{rang}(\mathbf{C}_{xx}) = 6$.

Dans cette simulation, on choisit $n = 50$, $k = [1: 2050]$ et $p = 4$, les fréquences sont normalisées $\Delta f = \Delta t = 1$. L'objectif étant de montrer que l'on retrouve bien les 2 fréquences qui ont le meilleur RSB.

On utilise la version classique de l'algorithme CSMFST en initialisant les matrices de covariance des données sur la base de 100 échantillons.

La figure IV.14 présente la distance $d(E_p(t), E_p^*(t))$ et la figure IV.15 illustre le rapport $\rho(t)$ calculé par l'algorithme CSMFST à chaque étape pour 20000 réalisations ainsi que celui calculé par l'algorithme CSMF $\rho^*(t)$.

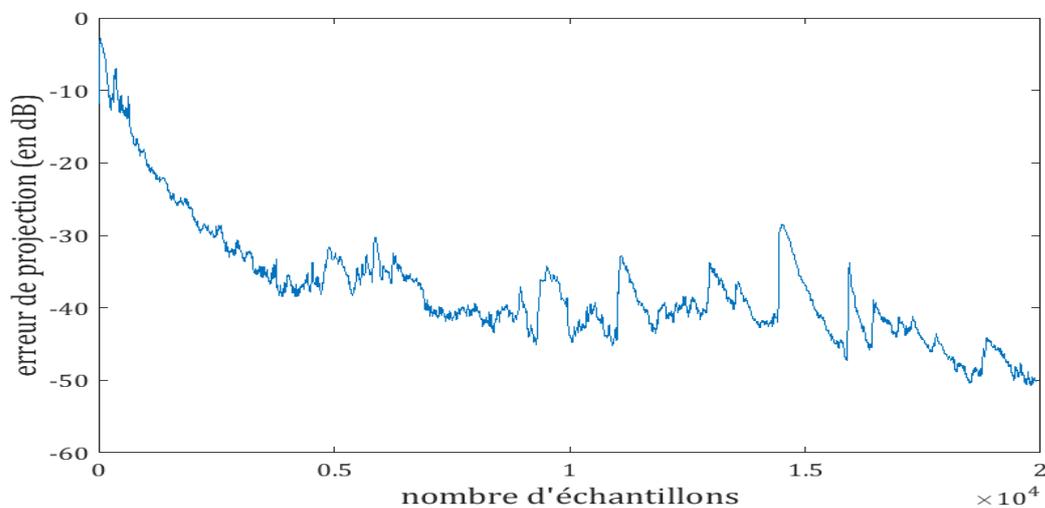


Figure IV.14. Distance $d(E_p(t), E_p^*(t))$

Applications : simulations et signaux réels

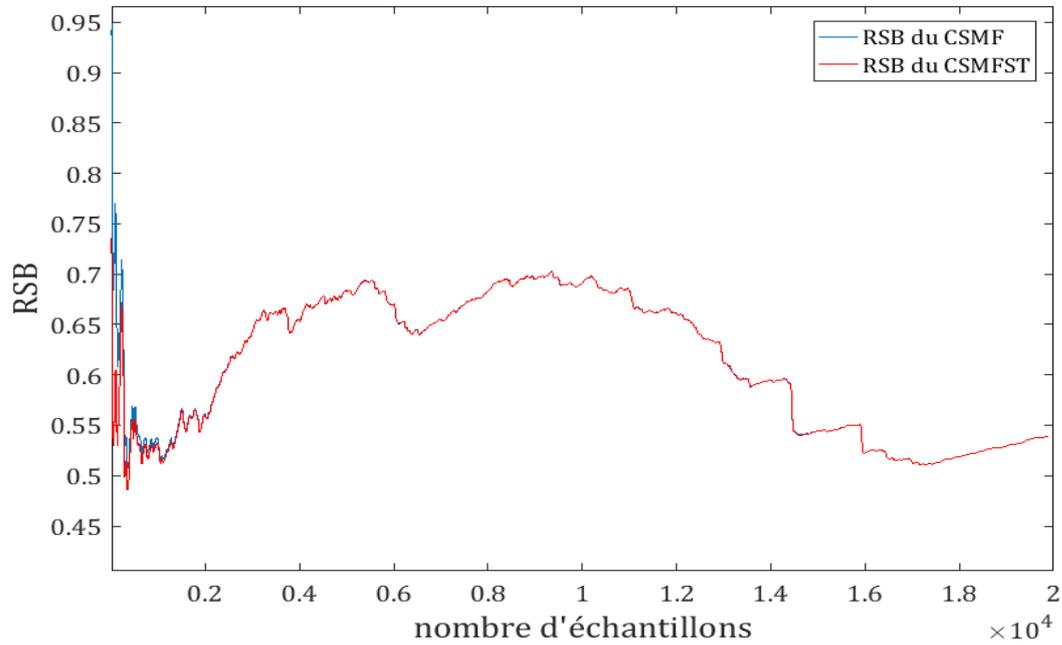


Figure IV.15. $\rho(t)$ et $\rho^*(t)$

L'étape suivante consiste à filtrer 20000 réalisations de signaux et de bruit en utilisant le sous-espace renvoyé par l'algorithme CSMFST et calculé avec toutes les réalisations du signal et du bruit.

La figure IV.16 présente la densité spectrale de puissance du signal et du bruit avant filtrage.

La figure IV.17 présente la densité spectrale de puissance des mêmes signaux après filtrage.

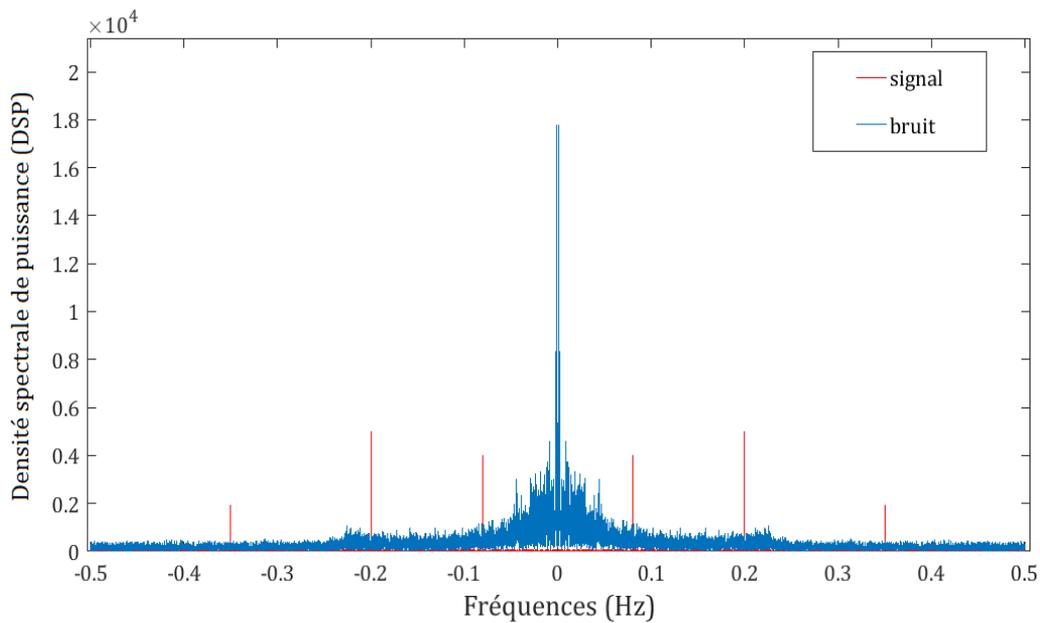


Figure IV.16. La densité spectrale de puissance des signaux avant filtrage.

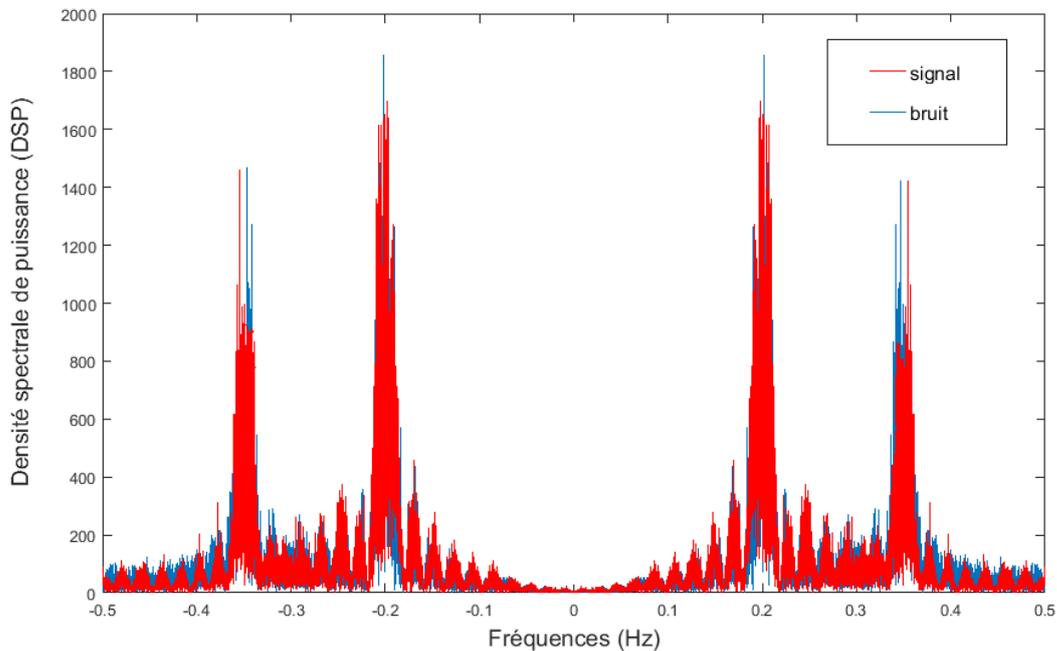


Figure IV.17. La densité spectrale de puissance des signaux après filtrage.

Si le bruit était blanc, on aurait trouvé les deux pics correspondant aux fréquences 0,1 Hz et 0,2 Hz, mais comme il s'agit d'un bruit non-blanc, on retrouve ceux qui correspondent aux fréquences 0,2 Hz et 0,35 Hz car ils présentent le meilleur rapport signal à bruit pour $p = 4$.

Les pics sont moins importants que sur la figure IV.16 puisque le calcul s'est fait sur 10000 échantillons.

IV.8 Conclusion

Dans ce dernier chapitre, des applications numériques ont été réalisées sur des simulations et sur des signaux réels.

Nous avons testé l'algorithme CSMFST dans le cas stationnaire puis dans le cas non stationnaire, ainsi nous avons comparé le sous-espace obtenu avec l'optimal (donné par l'algorithme CSMF) en calculant la distance $d(E_p(t), E_p^*(t))$ entre les deux.

Nous avons aussi testé la version modifiée du CSMFST en réalisant des scénarios dans lesquels le bruit est stationnaire de matrice de covariance estimée *a priori*. Dans ces simulations, les observations étaient, en fonction des scénarios, soit des échantillons du signal soit la superposition additive des échantillons du signal et du bruit.

Les résultats obtenus, ont montré une convergence rapide de l'algorithme même dans le cas de changements brusques et un rapport signal à bruit toujours proche de sa valeur théorique.

Applications : simulations et signaux réels

Concernant les signaux réels, on remarque que l'algorithme converge mais pas assez rapidement, ceci est dû au fait que les signaux présentent une très grande non-stationnarité : il n'y a pas de zones stationnaires suffisamment longues (comme dans nos simulations) pour que le filtre ait le temps de converger.

Conclusion générale

Les travaux présentés dans ce manuscrit portent sur le filtrage adapté stochastique sous contrainte qui repose sur deux principes : la projection de signaux initialement présents dans un espace de dimension « n » dans un espace de dimension fixée " $p < n$ " et la maximisation du rapport signal-à-bruit après filtrage.

L'objectif étant de chercher parmi tous les sous-espaces de dimension p , le « sous-espace optimal » dans lequel le rapport signal-à-bruit est maximal.

Nous avons rappelé, dans le premier chapitre, l'algorithme CSMF qui peut être appliqué dans deux situations :

- Détection : quand il y a un signal d'intérêt et un bruit.
- Classification : quand on est en présence d'un signal d'intérêt et de plusieurs perturbations potentiellement présentes additivement, chacune pouvant devenir le signal d'intérêt par permutations.

Dans le cas de la détection, le rapport signal-à-bruit élémentaire à optimiser constitue un critère unique tandis que dans celui de la classification, nous avons présenté plusieurs critères possibles à optimiser basés sur les moyennes des RSB élémentaires. Ces différents critères ne sont pas toujours possibles à classer en termes de performances optimales.

C'est pourquoi ma première contribution est développée dans le chapitre II. C'est la technique du filtrage multicritère qui consiste à combiner plusieurs critères afin d'améliorer les performances dans le cas de la détection multi-bruits.

Cette méthode a été appliquée sur des signaux simulés et réels (dauphins, cachalot, bruit de mer). Les courbes COR obtenus ont montré son efficacité.

Dans le chapitre III, nous avons traité le problème du tracking ou poursuite de sous-espace, pour la détection en se basant sur les méthodes de poursuite du sous-espace principal, tels que les algorithmes de type *Subspace Projection* (SP) et nous avons proposé, un algorithme adaptatif appelé *Constrained Stochastic Matched Filter Subspace Tracking* (CSMFST) afin de faire la poursuite du sous-espace optimal en temps réel, dans le cadre de la détection de signaux dont les propriétés statistiques, voire le RSB, varient dans le temps.

A la fin de ce chapitre, nous avons aussi démontré la convergence de notre algorithme vers le sous-espace optimal pouvant être calculé par le CSMF.

Conclusion générale

L'amélioration de la complexité de l'algorithme et de sa performance en termes de vitesse de convergence n'était pas notre objectif principal : on voulait démontrer, dans un premier temps, la faisabilité de l'algorithme.

Dans le dernier chapitre, nous avons testé l'algorithme CSMFST dans le cas stationnaire puis dans le cas non stationnaire en simulant différents scénarios avec des jeux de données puis en utilisant des signaux réels.

Afin de juger les performances de l'algorithme, nous avons comparé le sous-espace obtenu avec l'optimal (donné par l'algorithme CSMF) en calculant la distance entre les deux.

Les résultats obtenus étaient satisfaisants puisque :

- les distances obtenues variaient entre -30 dB et pouvaient aller jusqu'à -60 dB ce qui prouve une bonne convergence de l'algorithme.
- nous avons retrouvé en moyenne les bonnes valeurs du RSB : les mêmes que le CSMF.

Ces différents tests ont été réalisés en deux phases distinctes :

- la phase d'apprentissage qui consiste à prendre un certain nombre d'échantillons du signal et du bruit qui servent à initialiser les matrices de l'algorithme.
- la phase de faisabilité qui consiste à appliquer l'algorithme sur des échantillons autres que les signaux d'apprentissage.

Cette méthode suppose qu'on reçoit les échantillons du signal et les échantillons du bruit séparément, ce qui peut être problématique dans certaines situations concrètes.

Il a fallu, donc, mettre en place une version modifiée du CSMFST en réalisant des scénarios dans lesquels le bruit est stationnaire de matrice de covariance estimée *a priori* (le bruit pouvant être non blanc).

Dans ces expériences, les observations étaient, en fonction des scénarios, soit des échantillons du signal soit la superposition additive des échantillons du signal et du bruit.

Les résultats de cette méthode, ont montré une bonne convergence de l'algorithme et un rapport signal à bruit proche de sa valeur théorique.

Perspectives

Concernant les perspectives, des améliorations sont envisageables en termes de complexité algorithmique et de vitesse d'exécution de l'algorithme. En effet, notre objectif été de montrer la faisabilité de la poursuite et de trouver un algorithme en ce sens sans chercher à optimiser la complexité algorithmique.

La deuxième perspective, se présente dans le cas d'application du CSMFST aux signaux bioacoustiques. En effet, l'exemple traité a révélé les limites de l'algorithme quand les signaux

Conclusion générale

présentent une trop grande non-stationnarité : c'est-à-dire qu'il n'y a pas de zones stationnaires suffisamment longues (comme dans nos simulations) pour que le filtre ait le temps de converger. Une solution envisageable pour résoudre ce problème serait de suréchantillonner les signaux afin d'avoir beaucoup plus de points dans chaque zone stationnaire.

D'autres perspectives seraient d'appliquer l'algorithme CSMFST à des signaux de nature différente et qui présentent moins de non stationnarités que ce soit dans le domaine de l'acoustique sous-marine (on cite par exemple les sifflements des dauphins) ou encore dans d'autres domaines tels que l'acoustique de l'air ou les signaux biologiques ...

Annexe 1 :

Calculs des dérivées des critères

Le lagrangien a été défini par :

$$\mathcal{L}(\mathbf{X}_p, \boldsymbol{\Omega}) = \rho + \text{tr}(\boldsymbol{\Omega}(\mathbf{X}_p^T \mathbf{X}_p - \mathbb{I}_p))$$

où ρ est le critère à maximiser sous la contrainte $\mathbf{X}_p^T \mathbf{X}_p = \mathbb{I}_p$. On a $\boldsymbol{\Omega}$ symétrique réelle de dimension $p \times p$ et $\mathbf{X}_p \in M_{n \times p}$.

Il faut donc vérifier

$$\frac{\partial \mathcal{L}}{\partial \mathbf{X}_p} = \mathbb{O}.$$

On définit également les gains élémentaires :

$$\rho_i = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}.$$

L'expression de ρ peut prendre plusieurs formes, dont les suivantes.

A - Critère géométrique :

Pour la moyenne géométrique des gains élémentaires, le critère à maximiser est donné par :

$$\rho_G = \sqrt[K]{\rho} = \sqrt[K]{\prod_{i=1}^K \rho_i} = \sqrt[K]{\prod_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}}.$$

Alors :

$$\begin{aligned} \frac{\partial \rho_i}{\partial \mathbf{X}_p} &= \frac{2\mathbf{A} \mathbf{X}_p \text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p) - 2\mathbf{B}_i \mathbf{X}_p \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)^2} = \frac{2\mathbf{A} \mathbf{X}_p - 2\mathbf{B}_i \mathbf{X}_p \rho_i}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} = \frac{2(\mathbf{A} - \rho_i \mathbf{B}_i) \mathbf{X}_p}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} \\ &= \frac{\partial \rho_i}{\partial \mathbf{X}_p} = \frac{2(\mathbf{A} - \rho_i \mathbf{B}_i) \mathbf{X}_p}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}, \end{aligned}$$

et donc

$$\frac{\partial \mathcal{L}}{\partial \mathbf{X}_p} = \mathbb{O} \quad \rightarrow \quad \frac{\partial}{\partial \mathbf{X}_p} \prod_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} + \frac{\partial \text{tr}(\boldsymbol{\Omega}(\mathbf{X}_p^T \mathbf{X}_p - \mathbb{I}_p))}{\partial \mathbf{X}_p} = \mathbb{O}$$

Donc

Calculs des dérivées des critères

$$2 \sum_{i=1}^K \frac{(A - \rho_i \mathbf{B}_i) \mathbf{X}_p}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} \prod_{\substack{j=1 \\ j \neq i}}^K \rho_j + 2 \mathbf{X}_p \boldsymbol{\Omega} = \mathbb{O}$$

Or

$$\prod_{\substack{j=1 \\ j \neq i}}^K \rho_j = \rho_G \frac{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} = \frac{\rho_G}{\rho_i}$$

On multiplie à gauche et à droite par $\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p) \neq 0$

$$2 \sum_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} \frac{\rho_G}{\rho_i} (A - \rho_i \mathbf{B}_i) \mathbf{X}_p + 2 \mathbf{X}_p \boldsymbol{\Omega} \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p) = \mathbb{O}$$

$$2 \sum_{i=1}^K \rho_G (A - \rho_i \mathbf{B}_i) \mathbf{X}_p + 2 \mathbf{X}_p \boldsymbol{\Omega} \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p) = \mathbb{O}$$

Ce qui conduit à

$$\left(\sum_{i=1}^K (A - \rho_i \mathbf{B}_i) \right) \mathbf{X}_p = - \frac{\mathbf{X}_p \boldsymbol{\Omega} \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\rho_G}$$

ou

$$\left(A - \frac{1}{K} \sum_{i=1}^K \rho_i \mathbf{B}_i \right) \mathbf{X}_p = \mathbf{X}_p \boldsymbol{\Omega}_0$$

où $\boldsymbol{\Omega}_0$ est symétrique réelle de dimension $p \times p$.

B - Critère arithmétique :

On pose :

$$\rho_A = \frac{1}{K} \sum_{i=1}^K \rho_i.$$

Alors

$$\begin{aligned} \frac{\partial \rho_A}{\partial \mathbf{X}_p} &= \frac{1}{K} \sum_{i=1}^K \frac{\partial \rho_i}{\partial \mathbf{X}_p} = \frac{2}{K} \sum_{i=1}^K \frac{(A - \rho_i \mathbf{B}_i) \mathbf{X}_p}{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} = \frac{2}{K \text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} \sum_{i=1}^K \rho_i (A - \rho_i \mathbf{B}_i) \mathbf{X}_p \\ &= \frac{2}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} \frac{1}{K} \sum_{i=1}^K (\rho_i \mathbf{A} \mathbf{X}_p - \rho_i^2 \mathbf{B}_i \mathbf{X}_p) = \frac{2}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} \left(\frac{1}{K} \sum_{i=1}^K \rho_i \mathbf{A} \mathbf{X}_p - \frac{1}{K} \sum_{i=1}^K \rho_i^2 \mathbf{B}_i \mathbf{X}_p \right) \end{aligned}$$

Soit

$$\frac{\partial \rho_A}{\partial \mathbf{X}_p} = \frac{2}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} \left(\rho_A \mathbf{A} - \frac{1}{K} \sum_{i=1}^K \rho_i^2 \mathbf{B}_i \right) \mathbf{X}_p$$

Calculs des dérivées des critères

En divisant par ρ on obtient :

$$\frac{\partial \mathcal{L}}{\partial \mathbf{X}_p} = \mathbb{0} \quad \rightarrow \quad \frac{2\rho_A}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)} \left(\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \frac{\rho_i^2}{\rho} \mathbf{B}_i \right) \mathbf{X}_p + 2\mathbf{X}_p \boldsymbol{\Omega} = \mathbb{0}$$

$$\left(\mathbf{A} - \frac{1}{K} \sum_{i=1}^K \frac{\rho_i^2}{\rho_A} \mathbf{B}_i \right) \mathbf{X}_p = \mathbf{X}_p \boldsymbol{\Omega}_0$$

C - Critère harmonique :

$$\rho_H = \frac{K}{\sum_{i=1}^K \frac{1}{\rho_i}} = \frac{K}{\sum_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}} = \frac{1}{\frac{1}{K} \sum_{i=1}^K \frac{\text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}}$$

donc

$$\rho_H = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\frac{1}{K} \sum_{i=1}^K \text{tr}(\mathbf{X}_p^T \mathbf{B}_i \mathbf{X}_p)} = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T (\frac{1}{K} \sum_{i=1}^K \mathbf{B}_i) \mathbf{X}_p)} = \frac{\text{tr}(\mathbf{X}_p^T \mathbf{A} \mathbf{X}_p)}{\text{tr}(\mathbf{X}_p^T \mathbf{B} \mathbf{X}_p)}$$

avec :

$$\mathbf{B} = \frac{1}{K} \sum_{i=1}^K \mathbf{B}_i.$$

On retrouve le cas où l'on a 2 processus avec une matrice de covariance de bruit moyenne :

$$(\mathbf{A} - \rho_H \mathbf{B}) \mathbf{X}_p = \mathbf{X}_p \boldsymbol{\Omega}_0.$$

Annexe 2 :

Réalisation du CSMF

Création de signaux stationnaires de test :

Afin de pouvoir tester le programme, nous avons créé 4 filtres avec l'outil « fdatool » de Matlab.

Par la suite, nous avons appliqué les filtres précédemment réalisés sur des signaux aléatoires.

Les signaux sont modélisés comme la sortie d'un filtre FIR dont l'entrée est un bruit blanc obtenu à partir de variance $v = 0.98$.

Le résultat obtenu est alors 4 signaux différents se situant partiellement dans la même bande fréquentielle.

On affiche leurs spectres.

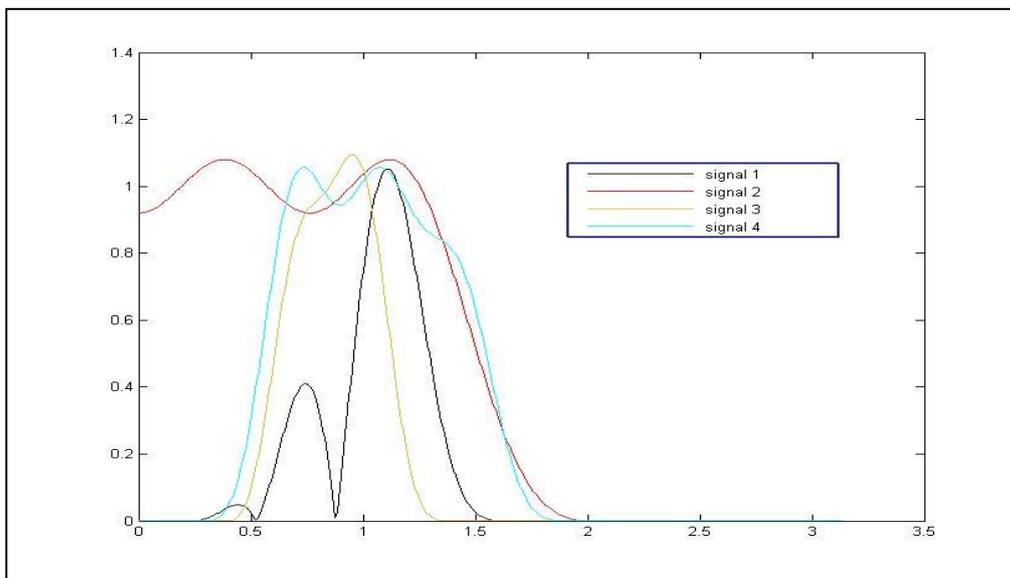


Figure 1 : Signaux de test

- Le signal 1 est la réponse fréquentielle du signal aléatoire sur lequel on a appliqué les coefficients d'un filtre passe bande dont les propriétés sont les suivantes :

$F_s = 48000$ Hz

$F_{stop1} = 2000$ Hz

$F_{pass1} = 8000$ Hz

Réalisation du CSMF

$F_{\text{pass}2} = 9000 \text{ Hz}$

$F_{\text{stop}2} = 12400 \text{ Hz}$

On sélectionne les 4096 points derniers points pour avoir s_1 .

- Le signal 2 est la réponse fréquentielle du signal aléatoire sur lequel on a appliqué les coefficients d'un filtre passe bas dont les propriétés sont les suivantes :

$F_s = 48000 \text{ Hz}$

$F_{\text{pass}} = 10000 \text{ Hz}$

$F_{\text{stop}} = 15000 \text{ Hz}$

On sélectionne les 4096 points derniers points pour avoir s_2 .

- Le signal 3 est la réponse fréquentielle du signal aléatoire sur lequel on a appliqué les coefficients d'un filtre passe bande dont les propriétés sont les suivantes :

$F_s = 48000 \text{ Hz}$

$F_{\text{stop}1} = 3200 \text{ Hz}$

$F_{\text{pass}1} = 5600 \text{ Hz}$

$F_{\text{pass}2} = 8000 \text{ Hz}$

$F_{\text{stop}2} = 10400 \text{ Hz}$

On sélectionne les 4096 points derniers points pour avoir s_3 .

- Le signal 4 est la réponse fréquentielle du signal aléatoire sur lequel on a appliqué les coefficients d'un filtre passe bande dont les propriétés sont les suivantes :

$F_s = 48000 \text{ Hz}$

$F_{\text{stop}1} = 2500 \text{ Hz}$

$F_{\text{pass}1} = 5000 \text{ Hz}$

$F_{\text{pass}2} = 7200 \text{ Hz}$

$F_{\text{stop}2} = 14400 \text{ Hz}$

On sélectionne les 4096 points derniers points pour avoir s_4 .

Création du signal test :

Le signal test construit à partir de ces différents signaux contient 11 zones :

Zone 1 : 200 points.

Zone 2 : 300 points.

Zone 3 : 100 points.

Zone 4 : 400 points.

Zone 5 : 100 points.

Réalisation du CSMF

Zone 6 : 450 points.

Zone 7 : 450 points.

Zone 8 : 500 points.

Zone 9 : 500 points.

Zone 10 : 500 points.

Zone 11 : 596 points.

Annexe 3 :

Calcul des vecteurs propres de $M(t)$

Le calcul des vecteurs propres de la matrice

$$\mathbf{M} = \mathbf{e}_A \mathbf{e}_A^T - \rho \mathbf{e}_B \mathbf{e}_B^T.$$

$\forall \mathbf{v}$, on peut écrire :

$$\mathbf{M}\mathbf{v} = \alpha_A \mathbf{e}_A - \rho \alpha_B \mathbf{e}_B$$

où $\alpha_A = \mathbf{e}_A^T \mathbf{v}$ et $\alpha_B = \mathbf{e}_B^T \mathbf{v}$.

Par ailleurs, un vecteur propre \mathbf{v}_p peut toujours s'écrire :

$$\mathbf{v}_p = \gamma_A \mathbf{e}_A - \gamma_B \mathbf{e}_B = \gamma_A \left(\mathbf{e}_A - \frac{\gamma_B}{\gamma_A} \mathbf{e}_B \right).$$

Si on note :

$$\alpha \equiv \frac{\gamma_B}{\gamma_A},$$

On aura

$$\begin{aligned} \mathbf{M}\mathbf{v}_p &= (\mathbf{e}_A \mathbf{e}_A^T - \rho \mathbf{e}_B \mathbf{e}_B^T)(\gamma_A \mathbf{e}_A - \gamma_B \mathbf{e}_B) \\ \mathbf{M}\mathbf{v}_p &= (\|\mathbf{e}_A\|^2 \gamma_A - \gamma_B \mathbf{e}_A^T \mathbf{e}_B) \mathbf{e}_A - \rho(\gamma_A \mathbf{e}_B^T \mathbf{e}_A - \gamma_B \|\mathbf{e}_B\|^2) \mathbf{e}_B \\ \mathbf{M}\mathbf{v}_p &= \gamma_A \left((\|\mathbf{e}_A\|^2 - \alpha \mathbf{e}_A^T \mathbf{e}_B) \mathbf{e}_A - \rho(\mathbf{e}_B^T \mathbf{e}_A - \alpha \|\mathbf{e}_B\|^2) \mathbf{e}_B \right) \end{aligned} \tag{A.2}$$

$\mathbf{v}_p = \mathbf{e}_A - \alpha \mathbf{e}_B$ est un vecteur propre de \mathbf{M} , avec la valeur propre λ si et seulement si

$$\begin{cases} \|\mathbf{e}_A\|^2 - \alpha \mathbf{e}_A^T \mathbf{e}_B = \lambda \\ \rho(\mathbf{e}_B^T \mathbf{e}_A - \alpha \|\mathbf{e}_B\|^2) = \lambda \alpha \end{cases}$$

i.e. si

$$\alpha = \frac{\rho(\mathbf{e}_B^T \mathbf{e}_A - \alpha \|\mathbf{e}_B\|^2)}{\|\mathbf{e}_A\|^2 - \alpha \mathbf{e}_A^T \mathbf{e}_B},$$

ou

$$\alpha^2(\mathbf{e}_B^T \mathbf{e}_A) - \alpha(\|\mathbf{e}_A\|^2 + \rho \|\mathbf{e}_B\|^2) + \rho(\mathbf{e}_A^T \mathbf{e}_B) = 0 \tag{A.2}$$

et

Calcul des vecteurs propres de $M(t)$

$$\lambda = \|\mathbf{e}_A\|^2 - \alpha \mathbf{e}_A^T \mathbf{e}_B = \mathbf{e}_A^T (\mathbf{e}_A - \alpha \mathbf{e}_B) = \mathbf{e}_A^T \mathbf{v}_p.$$

Cherchons la solution générale en résolvant l'équation quadratique (A.2).

Si $\mathbf{e}_A^T \mathbf{e}_B = 0$, alors, $\lambda = \|\mathbf{e}_A\|^2$ et $\alpha = 0$, donc $\mathbf{v}_p = \mathbf{e}_A$.

Si $\mathbf{e}_A^T \mathbf{e}_B \neq 0$, alors,

$$\Delta = [(\|\mathbf{e}_A\|^2 + \rho \|\mathbf{e}_B\|^2)^2 - 4\rho (\mathbf{e}_A^T \mathbf{e}_B)^2].$$

Et

$$\alpha = \frac{\|\mathbf{e}_A\|^2 + \rho \|\mathbf{e}_B\|^2 \pm \sqrt{\Delta}}{2(\mathbf{e}_A^T \mathbf{e}_B)}$$
$$\alpha = \frac{1}{2} \left\{ \frac{\|\mathbf{e}_A\|^2 + \rho \|\mathbf{e}_B\|^2}{\mathbf{e}_A^T \mathbf{e}_B} \pm \sqrt{\left[\frac{\|\mathbf{e}_A\|^2 + \rho \|\mathbf{e}_B\|^2}{\mathbf{e}_A^T \mathbf{e}_B} \right]^2 - 4\rho} \right\} \quad (\text{A.3})$$

Les deux solutions trouvées seront notées α^+ et α^- .

Si l'on note λ_m la plus grande valeur propre parmi les deux valeurs propres qu'on a trouvé, on peut écrire

$$\lambda_m = \|\mathbf{e}_A\|^2 - \mathbf{e}_A^T \mathbf{e}_B \alpha^-,$$

et

$$\mathbf{v}_p = \mathbf{e}_A - \alpha^- \mathbf{e}_B.$$

Annexe 4 :

Matrice de covariance et sous-espace principal

Soient $\mathbf{C}(t) \in M_{n \times n}$ une matrice de covariance (hermitienne définie positive) et $(\lambda_i(t), \mathbf{w}_i(t))$ ses éléments propres :

$$\mathbf{C}(t)\mathbf{w}_i(t) = \lambda_i(t)\mathbf{w}_i(t), \quad 1 \leq i \leq n.$$

On a $\lambda_i(t) \in \mathbb{R}^+$, et on peut donc les classer :

$$\lambda_1(t) \geq \lambda_2(t) \geq \dots \geq \lambda_n(t) \geq 0,$$

et les vecteurs propres sont naturellement orthogonaux et peuvent être normalisés :

$$\mathbf{w}_i^T(t)\mathbf{w}_j(t) = \delta_{ij}.$$

Si l'on note $\mathbf{W}_t(t) = [\mathbf{w}_1(t) \dots \mathbf{w}_n(t)]$ et $\Lambda_t(t) = [\lambda_i(t)]$, toutes deux de dimension $n \times n$, alors :

$$\begin{cases} \mathbf{C}(t)\mathbf{W}_t(t) = \mathbf{W}_t(t)\Lambda_t(t) \\ \mathbf{W}_t^T(t)\mathbf{W}_t(t) = \mathbb{I}_n \\ \mathbf{W}_t(t)\mathbf{W}_t^T(t) = \mathbb{I}_n \end{cases}$$

On a alors

$$\mathbf{C}(t) = \mathbf{W}_t(t)\Lambda_t(t)\mathbf{W}_t^T(t) \quad \text{ou} \quad \Lambda_t(t) = \mathbf{W}_t^T(t)\mathbf{C}(t)\mathbf{W}_t(t).$$

On peut aussi écrire que

$$\mathbf{C}(t) = \sum_{i=1}^n \lambda_i(t)\mathbf{w}_i(t)\mathbf{w}_i^T(t).$$

Soit $p < n$ et $\mathbf{W}(t) = [\mathbf{w}_1(t) \dots \mathbf{w}_p(t)] \in M_{n \times p}$; l'espace engendré par les p vecteurs colonne de \mathbf{W} constitue le sous-espace principal de dimension p , et il est noté

$$E^p(t) = \text{span}\{\mathbf{w}_1(t), \mathbf{w}_2(t), \dots, \mathbf{w}_p(t)\} = \text{span}\{\mathbf{W}(t)\}.$$

On note $q = n - p$ et $\mathbf{W}_\perp(t) = [\mathbf{w}_{p+1}(t) \dots \mathbf{w}_n(t)] \in M_{n \times q}$ qui engendre $E_\perp^p(t)$ tel que $E_\perp^p(t) \oplus E^p(t) = E^n$.

On notera aussi $\mathbf{C}(t)\mathbf{W}(t) = \mathbf{W}(t)\Lambda(t)$ avec, donc, $\Lambda(t) = [\lambda_i(t)]_{1 \leq i \leq p} \in M_{p \times p}$, et $\mathbf{C}(t)\mathbf{W}_\perp(t) = \mathbf{W}_\perp(t)\Lambda_\perp(t)$ où $\Lambda_\perp(t) = [\lambda_i(t)]_{1+p \leq i \leq n} \in M_{q \times q}$.

On les propriétés suivantes :

$$\begin{cases} \mathbf{W}^T(t)\mathbf{W}(t) = \mathbb{I}_p \\ \mathbf{W}_\perp^T(t)\mathbf{W}_\perp(t) = \mathbb{I}_q \\ \mathbf{W}^T(t)\mathbf{W}_\perp(t) = \mathbb{O}_{pq} \end{cases}$$

$\mathbf{W}(t)$ et $\mathbf{W}_\perp(t)$ sont des matrices orthogonales : $\mathbf{W}_T(t) = [\mathbf{W}(t) \quad \mathbf{W}_\perp(t)]$.

On peut aussi écrire que

$$\mathbf{\Pi}(t) \equiv \mathbf{W}(t)\mathbf{W}^T(t) = \sum_{i=1}^p \mathbf{w}_i(t)\mathbf{w}_i^T(t).$$

$\mathbf{\Pi}(t) \in M_{n \times n}$ est une projection orthogonale sur $E^p(t)$ parallèlement à $E_\perp^p(t)$:

$$\begin{cases} \forall \mathbf{w}_i(t) \in E^p, \mathbf{\Pi}\mathbf{w}_i(t) = \mathbf{w}_i(t) \\ \forall \mathbf{w}_i(t) \in E_\perp^p, \mathbf{\Pi}(t)\mathbf{w}_i(t) = \mathbf{0} \end{cases}$$

C'est-à-dire que $\mathbf{\Pi}(t)$ est une matrice de dimension $n \times n$ mais de rang p .

Note 1 : $\mathbf{\Pi}(t) = \mathbf{W}(t)\mathbf{W}^T(t)$ mais il existe une infinité d'autres matrices $\mathbf{U}(t) \in M_{n \times p}$ qui vérifient $\mathbf{\Pi}(t) = \mathbf{U}(t)\mathbf{U}^T(t)$!

En effet, si $\mathbf{U}(t) = \mathbf{W}(t)\mathbf{Q}$ avec \mathbf{Q} matrice unitaire de $M_{p \times p}$ (les vecteurs colonne de $\mathbf{U}(t)$ sont combinaisons linéaires de $\mathbf{w}_1(t) \dots \mathbf{w}_r(t)$), $\mathbf{U}(t)\mathbf{U}^T(t) = \mathbf{W}(t)\mathbf{Q}\mathbf{Q}^T\mathbf{W}^T(t) = \mathbf{W}(t)\mathbf{W}^T(t)$ (et $\mathbf{U}^T(t)\mathbf{U}(t) = \mathbf{Q}^T\mathbf{W}^T(t)\mathbf{W}(t)\mathbf{Q} = \mathbf{Q}^T\mathbb{I}_p\mathbf{Q} = \mathbf{Q}^T\mathbf{Q} = \mathbb{I}_p$).

Note 2 : tout élément \mathbf{x} de E^n peut s'écrire :

$$\mathbf{x} = \mathbf{x}_\parallel + \mathbf{x}_\perp$$

avec $\mathbf{x}_\parallel = \mathbf{\Pi}\mathbf{x} \in E^p$ et $\mathbf{x}_\perp = (\mathbb{I}_n - \mathbf{\Pi})\mathbf{x} \in E_\perp^p$ ($\mathbb{I}_n - \mathbf{\Pi}$ est également une projection).

On peut aussi définir \mathbf{u} de la façon suivante :

$$\mathbf{u} = \frac{\mathbf{x}_\perp}{\|\mathbf{x}_\perp\|}$$

on a donc $\mathbf{u} \in E_\perp^p$ et $\|\mathbf{u}\| = 1$.

Trouver $E^p(t)$, c'est donc trouver les p vecteurs propres principaux $\mathbf{W}(t)$ de $\mathbf{C}(t)$:

$$\mathbf{W}(t) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T\mathbf{H} = \mathbb{I}_p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{H}^T\mathbf{C}(t)\mathbf{H}) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T\mathbf{H} = \mathbb{I}_p}}{\operatorname{argmax}} \operatorname{tr}(\mathbf{C}(t)\mathbf{H}\mathbf{H}^T).$$

où $\mathbf{\Pi} = \mathbf{H}\mathbf{H}^T$ est un projecteur orthogonal de rang p .

On peut écrire également :

$$W(t) = \underset{\substack{\mathbf{H} \in M_{n \times p} \\ \mathbf{H}^T \mathbf{H} = \mathbb{I}_p}}{\operatorname{argmax}} \frac{\operatorname{tr}(\mathbf{H}^T \mathbf{C}(t) \mathbf{H})}{\operatorname{tr}(\mathbf{H}^T \mathbf{H})},$$

Trouver $E^p(t)$, c'est aussi trouver le projecteur orthogonal $\mathbf{\Pi}(t)$ qui maximise le critère suivant :

$$J(\mathbf{\Pi}(t)) = \operatorname{tr}(\mathbf{C}(t) \mathbf{\Pi}(t)) \quad \text{ou} \quad \frac{\operatorname{tr}(\mathbf{C}(t) \mathbf{\Pi}(t))}{\operatorname{tr}(\mathbf{\Pi}(t))}.$$

D'après la note précédente, si $\mathbf{\Pi}(t)$ est unique, un algorithme de maximisation de J consiste à trouver $\mathbf{W}(t)$ ou tout autre matrice $\mathbf{U}(t) \in M_{n \times p}$ telle que $\mathbf{\Pi}(t) = \mathbf{W}(t) \mathbf{W}^T(t) = \mathbf{U}(t) \mathbf{U}^T(t)$. En effet si $\mathbf{U}(t) = \mathbf{W}(t) \mathbf{Q}$ avec $\mathbf{Q} \mathbf{Q}^T = \mathbb{I}_p$, alors :

$$\begin{aligned} \frac{\operatorname{tr}(\mathbf{U}^T(t) \mathbf{C}(t) \mathbf{U}(t))}{\operatorname{tr}(\mathbf{U}^T(t) \mathbf{U}(t))} &= \frac{\operatorname{tr}(\mathbf{Q}^T \mathbf{W}^T(t) \mathbf{C}(t) \mathbf{W}(t) \mathbf{Q})}{\operatorname{tr}(\mathbf{Q}^T \mathbf{W}^T(t) \mathbf{W}(t) \mathbf{Q})} = \frac{\operatorname{tr}(\mathbf{W}^T(t) \mathbf{C}(t) \mathbf{W}(t) \mathbf{Q} \mathbf{Q}^T)}{\operatorname{tr}(\mathbf{W}^T(t) \mathbf{W}(t) \mathbf{Q} \mathbf{Q}^T)} \\ &= \frac{\operatorname{tr}(\mathbf{W}^T(t) \mathbf{C}(t) \mathbf{W}(t))}{\operatorname{tr}(\mathbf{W}^T(t) \mathbf{W}(t))}. \end{aligned}$$

Annexe 5 :

Applications numériques

Construction des réalisations à partir de valeurs/vecteurs propres fixés :

Afin de retrouver les matrices de covariance correspondantes aux valeurs et vecteurs propres fixés, on utilise l'algorithme suivant :

Génération des réalisations

Pour $t = 1$ à « nombre de zones »

Pour $i = 1$ jusqu'au nombre de réalisations souhaité dans chaque zone

$x_i = \text{zeros}(n, 1)$

Pour $j = 1$ à n (dimension de x_i)

$x_i = x_i + a_j v_j$

où a_i est une variable aléatoire gaussienne centrée de variance λ_j la valeur propre associée au vecteur propre v_j de C .

Fin pour

Fin pour

Fin pour

Exemples dans le cas stationnaire

Scénario 1 :

$$[\lambda_{i,1}^S] = [20, 19, 17, 1, 1, 1, 1, 1]^T$$

$$[\lambda_{i,1}^B] = [1, 1, 1, 1, 1, 1, 1, 1]^T, \text{ donc bruit blanc.}$$

Une seule zone : 35000 réalisations

Les vecteurs propres ne changent pas.

Applications numériques

Scénario 2 :

$$[\lambda_{i,1}^S] = [20,18,16,1,1,1,1,1]^T$$

$$[\lambda_{i,1}^B] = [2,3,4,1,2,3,4,5]^T \text{ donc bruit non blanc.}$$

Une seule zone : 35000 réalisations

Les valeurs propres du signal V^S et du bruit V^B sont égales.

$$V^B = V^S$$

Les vecteurs propres ne changent pas.

Exemples dans le cas non stationnaire

Scénario 3 :

$$[\lambda_{i,j}^S] = \begin{bmatrix} 20 & 20 & 20 & 20 \\ 19 & 19 & 19 & 19 \\ 17 & 17 & 17 & 17 \\ 2 & 2 & 2 & 2 \\ 1.7 & 1.7 & 1.7 & 1.7 \\ 1.5 & 1.5 & 1.5 & 1.5 \\ 1.2 & 1.2 & 1.2 & 1.2 \\ 0.9 & 0.9 & 0.9 & 0.9 \end{bmatrix}, \quad [\lambda_{i,j}^B] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

On crée 4 zones stationnaires en permutant 2 vecteurs propres dans chaque zone.

Le bruit est blanc.

Dans la première zone, on garde les mêmes V^S que le premier scénario. On permute uniquement les vecteurs propres du signal comme suit :

1 ^{ère} zone : 5000 réalisations	pas de permutation
2 ^{ème} zone : 15000 réalisations	2 avec 4
3 ^{ème} zone : 10000 réalisations	1 avec 5
4 ^{ème} zone : 5000 réalisations	3 avec 6

Applications numériques

Scenario 4 :

$$[\lambda_{i,j}^S] = \begin{bmatrix} 20 & 11 & 11 & 11 \\ 19 & 19 & 10 & 10 \\ 17 & 17 & 17 & 9 \\ 2 & 2 & 2 & 2 \\ 1.7 & 1.7 & 1.7 & 1.7 \\ 1.5 & 1.5 & 1.5 & 1.5 \\ 1.2 & 1.2 & 1.2 & 1.2 \\ 0.9 & 0.9 & 0.9 & 0.9 \end{bmatrix}, [\lambda_{i,j}^B] = \begin{bmatrix} 1 & 2 & 1 & 6 \\ 1 & 3 & 1 & 8 \\ 1 & 4 & 1 & 5 \\ 1 & 1 & 1 & 4 \\ 1 & 2 & 1 & 5 \\ 1 & 3 & 1 & 7 \\ 1 & 4 & 1 & 7 \\ 1 & 1 & 1 & 4 \end{bmatrix}$$

On crée 4 zones stationnaires en permutant 2 vecteurs propres dans chaque zone.

Le bruit est blanc dans les zones 1 et 3.

On permute uniquement les vecteurs propres du signal comme suit :

1^{ère} zone : 5000 réalisations pas de permutation

2^{ème} zone : 15000 réalisations 2 avec 4

3^{ème} zone : 10000 réalisations 1 avec 5

4^{ème} zone : 15000 réalisations 3 avec 6

Scenario 5 :

Les mêmes $[\lambda_{i,j}^S]$ que le scénario 4.

Les mêmes $[\lambda_{i,j}^B]$ que le scénario 2 (dans toutes les zones).

Les mêmes V^S que le scénario 4.

Le bruit est non blanc.

Scenario 6 :

La même chose que le scénario 5.

B est fixée.

$x_A(t)$: signal.

Scenario 7 :

La même chose que le scénario 6.

B est fixée.

$x_A(t)$: signal + bruit

Bibliographie

- [1] H.L. Van Trees, *Radar-Sonar Signal Processing and Gaussian Signals in Noise*, dans *Detection, Estimation, and Modulation Theory*, Wiley-Blackwell, Part 3, 2002.
- [2] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, 1994.
- [3] T. Kailath, H. V. Poor, *Detection of Stochastic Processes*, dans *IEEE Transactions on Information Theory*, vol. 44, n° 6, p. 30, 1998.
- [4] J.G. Proakis, *Digital Communications: 3rd (third) Edition*, McGraw-Hill, 1995.
- [5] B. Borloz, *Estimation, détection, classification par maximisation du rapport signal-à-bruit : le filtre adapté stochastique sous contrainte*, thèse, Université de Toulon, 2005.
- [6] D. O. North, *An Analysis of the factors which determine signal/noise discrimination in pulsed-carrier systems*, *Proceedings of the IEEE*, vol. 51, n° 7, p. 1016-1027, 1963.
- [7] F. Chavand, *Des données à l'information : De l'invention de l'écriture à l'explosion numérique*. ISTE Editions, p. 412, 2017.
- [8] Z. Zhao, A. Zhao, J. Hui, B. Hou, R. Sotudeh and F.Niu, *A Frequency-Domain Adaptive Matched Filter for Active Sonar Detection Acoustic Sensing and Ultrasonic Drug Delivery*, *Sensors*, vol 17, 2017.
- [9] S. Verdu, H.V. Poor, *Signal Selection for Robust Matched Filtering*, dans *IEEE Trans. Communications*, Vol. COM-31, no. 5, p.667-670, 1983.
- [10] S. Verdu, H.V. Poor, *Minimax Robust Discrete-time Matched Filters*, *IEEE Trans. Communications*, Vol. COM-31, no. 2, p.208-215, 1983.
- [11] J-F. Cavassilas. *Stochastic matched filter* dans *Proceedings of the Institute of Acoustics (International Conference on Sonar Signal Processing)*, vol. 13, Part 9, 1991.
- [12] J-F. Cavassilas, B. Xerri, *Extension de la notion de filtre adapté. Contribution à la détection de signaux courts en présence de termes perturbateurs*, dans *la Revue Traitement du Signal* vol. 10 no 3, 1992.
- [13] J. Bonnal, P. Danes, et M. Renaud, *Detection of acoustic patterns by stochastic Matched Filtering*, in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, 2010, p. 1970-1975.
- [14] B. Xerri, J-F. Cavassillas, et B. Borloz, *Le filtre adapté stochastique. Contribution à la détection de textures bidimensionnelles*, dans *Seizième Colloque GRETSI 1997, Grenoble*, p.4, 1997.
- [15] P. Courmontagne, *Le filtrage adapté stochastique adaptatif appliqué aux images bruitées*, dans *Signal Processing, 1999*, vol. 16, no 4, 1991.
- [16] P. Courmontagne et F. Chaillan, *The Adaptive Stochastic Matched Filter for SAS Images denoising*, dans *OCEANS 2006, USA*, p.1-6, 2006.

Bibliographie

- [17] P. Courmontagne, G. Julien, et M. E. Bouhier, *An improvement to the pulse compression scheme*, dans *OCEANS'10 IEEE SYDNEY*, Sydney, Australia, p. 1-10, 2010.
- [18] J.-L. Mori, P. Gounon, *The use of stochastic matched filter in active sonar*, dans *10th European Signal Processing Conference*, 2000.
- [19] C. Frascini, *Filtre adapté stochastique pour les signaux continus avec application au sonar actif*, thèse, Université de Toulon, 2009.
- [20] F. Bénard, H. Glotin, et P. Giraudet, *Highly Defined Whale Group Tracking by Passive Acoustic Stochastic Matched Filter*, *Advances in Sound Localization*, Dr. Pawel Strumillo (Ed.), Toulon, 2011.
- [21] P. Giraudet et H. Glotin, *Echo-Robust and Real-Time 3D Tracking of Marine Mammals Using Their Transient Calls Recorded by Hydrophones Array*, dans *IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 4, 2006.
- [22] F. Caudal et H. Glotin, *Stochastic Matched Filter outperforms Teager-Kaiser-Mallat for tracking a plurality of sperm whales*, in *2008 New Trends for Environmental Monitoring Using Passive Systems*, Hyères, France, p. 1-9, 2008.
- [23] L. Bouffaut, R. Dreo, V. Labat, A.-O. Boudraa, et G. Barruol, *Antarctic Blue Whale calls detection based on an improved version of the stochastic matched filter*, p. 2319-2323, 2017.
- [24] L. Bouffaut, R. Dreo, V. Labat, A.-O. Boudraa, et G. Barruol, *Passive stochastic matched filter for Antarctic blue whale call detection* dans *The Journal of the Acoustical Society of America*, vol 144, 2018.
- [25] G. Julien, *Filtrage Stochastique et amélioration des performances des systèmes de positionnement d'engins sous-marins en milieu bruyant*, thèse, Université de Toulon, 2012.
- [26] L. Bouffaut, R. Dreo, V. Labat, et A.-O. Boudraa, *Filtrage Adapté Stochastique passif pour la détection de plongeurs*, dans *24ème Colloque GRETSI 2017*, Juan-Les-Pins, 2017.
- [27] L. Bouffaut et A.-O. Boudraa, *Passive stochastic matched filter: application to scuba divers detection*, dans *The 42nd IEEE International Conference on Acoustics, Speech And Signal Processing - ICASSP2017*, New Orleans - LA (USA), 2017.
- [28] B. Borloz et B. Xerri, *Detection by SNR Maximization: Application to the Blind Source Separation Problem* dans *Independent Component Analysis and Blind Signal Separation dans Fifth International Conference, ICA 2004, Granada, Spain 2004, Proceedings. Springer Science & Business Media*, p.602-609, 2004.
- [29] B. Borloz et B. Xerri, *Subspace signal-to-noise ratio maximization: the constrained stochastic matched filter*, dans *Proceedings of the Eighth International Symposium on Signal Processing and Its Applications*, 2005, Sydney, Australia, vol. 2, p. 735-738, 2005.
- [30] B. Borloz et B. Xerri, *Subspace SNR Maximization: The Constrained Stochastic Matched Filter*, dans *IEEE Transactions on Signal Processing*, vol. 59, n° 4, p. 1346-1355, 2011.
- [31] N. Juennard, B. Xerri, et B. Borloz, *Detection by the Stochastic Classification Matched Filter*, dans *the 5th WSEAS Int. Conf. on Signal Processing, Robotics and Automation*, Madrid, Spain, p. 341-344, 2006.

Bibliographie

- [32] N. Juennard, *Détection et localisation de particules de très hautes énergies en acoustique sous-marine*, thèse, Université de Toulon, 2007
- [33] N. Juennard, C. Jauffret, et B. Xerri, *Very-High-Energy Neutrinos: Detection and Estimation in the Mediterranean Sea*, dans *IEEE Journal of Oceanic Engineering*, vol. 35, n° 4, p. 917-935, 2010.
- [34] F. Cismondi, B. Xerri, C. Jauffret, J. Schlosser, N. Vignal, et A. Durocher, *Analysis of SATIR test for the qualification of high heat flux components: defect detection and classification by signal-to-noise ratio maximization*, *Phys. Scr.*, vol. 2007, n° T128, p. 213, 2007.
- [35] F. Cismondi, *Contrôle non destructif par thermographie infrarouge des composants face au plasma des machines de fusion contrôlée : modélisation et traitement statistique du signal associé*, thèse, Université de Toulon, 2007.
- [36] M. Chagmani, B. Xerri, et B. Borloz, *Application of the Constrained Stochastic Matched Filter Subspace Tracking to Submarine Acoustic Signals*, in *Proceedings of the 2nd Mediterranean Conference on Pattern Recognition and Artificial Intelligence*, p. 33-37, 2018.
- [37] M. Chagmani, B. Xerri, B. Borloz, et C. Jauffret, *The constrained stochastic matched filter subspace tracking*, dans *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, p. 205-209, 2017.
- [38] *La recherche sur les cétacés, SOS Grand Bleu, protection des dauphins et des mammifères marins.*
- [39] H. Delacour, *La Courbe ROC (receiver operating characteristic): principes et principales applications en biologie clinique*, p. 145-54, 2005.
- [40] *GREC Groupe de recherche sur les cétacés, Le monde du silence et les cétacés, GREC*, disponible en ligne sur <http://www.cetaces.org>, 2009.
- [41] SABIOD Scaled Acoustic, *BIODiversity platform*, (http://sabiiod.univ-tln.fr/data_samples.html)
- [42] G.H. Golub, C.F. Van Loan, *Matrix Computations*, 3^{ème} Ed. Johns Hopkins University Press, Baltimore, MD, USA, 1996.
- [43] G.H. Golub, C.F. Van Loan, *Some modified matrix eigenvalue problems*, *SIAM Review*, 15, p. 318-334, 1973.
- [44] Schreiber, *Implementation of adaptive array algorithms*, *IEEE Transactions ASSP*, p. 1038-1045, 1986.
- [45] Karasalo, *Estimating the covariance matrix by signal subspace averaging*, *IEEE Transactions ASSP-34*, p. 8-12, 1986.
- [46] R. DeGroat, *Non-iterative subspace tracking*, *IEEE Trans. Signal. Processing*, SP-40, p. 571-577, 1992.
- [47] S. Marcos, J. Sanchez, M. Benidir, P. Comon, *Algorithmes adaptatifs pour la poursuite du sous-espace source*, Chapitre 14 dans *Les Méthodes à Hautes Résolution. Traitement d'antenne et analyse spectrale*, Ouvrage collectif sous la direction de Sylvie Marcos, Editions HERMES, 1998.
- [48] S. Marcos, L. Atallah, *Algorithme adaptatif à haute résolution et faible complexité pour la*

Bibliographie

- poursuite des directions d'arrivée de sources mobiles* Actes du Colloque GRETSI, Vannes, 1999.
- [49] S. Marcos, J. Sanchez, M. Benidir, *An adaptative algorithm for direction finding and array shape estimation in a non-stationary environment*, *Journal of VLSI Signal Processing Systems, Special issues on Array Optimization and Adaptive Tracking Algorithms*, vol 14(1), 1996.
- [50] H. Belkacimi, S. Marcos, *Fast iterative subspace algorithms for airborne space time adaptive processing (STAP) radar*, *EURASIP Journal on Advances in Signal Processing, Special issue on Multisensor Processing for Signal Extraction and Applications*, 2006.
- [51] S. Beau, S. Marcos, *Range dependant clutter rejection using range recursive space time adaptive processing (STAP)*, *Elsevierignal Processing*, vol 90(1),2010.
- [52] P. Strobach, *Bi-iteration SVD subspace tracking algorithms*, dans *IEEE Transactions on Signal Processing*, vol. 45, no 5, p. 1222-1240, 1997.
- [53] R. Badeau, G. Richard, et B. David, *Sliding Window Adaptive SVD Algorithms*, *IEEE Transactions on Signal Processing*, vol. 52, no 1, p. 1-10, 2004.
- [54] Shan Ouyang et Yingbo Hua, *Bi-iterative least-square method for subspace tracking*, dans *IEEE Transactions on Signal Processing*, vol. 53, no 8, p. 2984-2996, 2005.
- [55] P. Strobach, *Sliding window adaptive SVD using the unsymmetric householder partial compressor*, *Signal Processing*, vol. 90, no 1, p. 352-362, 2010.
- [56] B. Yang, *Projection approximation subspace tracking*, dans *IEEE Transactions on Signal Processing*, 1995.
- [57] K. Abed-Meraim, A. Chkeif, and Y. Hua, *Fast orthonormal PAST algorithm*, dans *IEEE Signal Processing*, vol. 3, no 2, p. 60 – 62, 2000.
- [58] S. Douglas, *Numerically-robust adaptive subspace tracking using householder transformations*, *Sensor Array and Multichannel Signal Processing Workshop.2000*. Proceedings de *the 2000 IEEE*, p. 499–503, 2000.
- [59] K. Abed-Meraim, A. Chkeif, Y. Hua, S. Attallah, *On a class of orthonormal algorithms for principal and minor subspace tracking*, *J. VLSI Signal Process. Syst. Signal Image Video Technol*, vol. 31, no. 1, p. 57–70, 2002.
- [60] A. Valizadeh, M. Karimi, *Fast subspace tracking algorithm based on the constrained projection approximation*, *EURASIPJ. Adv. Signal Process*, p. 1-16, 2009.
- [61] T. F. Chan, *Rank revealing QR factorizations*, *Linear Algebra and its Applications*, vol. 88-89, p. 67-82, 1987.
- [62] E. Oja, *Simplified neuron model as a principal component analyzer*, dans *Journal of Mathematical Biology*, p. 267-273, 1982.
- [63] P. Strobach, *Low-rank adaptive filters*, *IEEE Transactions on Signal Processing*, vol. 44, no. 12, p. 2932-2947, 1996.

Bibliographie

- [64] R. Badeau, G. Richard, B. David, and K. Abed-Meraim, *Approximated power iterations for fast subspace tracking*, dans *Seventh International Symposium on Signal Processing and Its Applications*, vol. 2, 2003.
- [65] R. Badeau, B. David, and G. Richard, *Fast approximated power iteration subspace tracking*, *IEEE Transactions on Signal Processing*, vol. 53, no. 8, p. 2931 – 2941, 2005.
- [66] C. E. Davila, *Efficient, high performance, subspace tracking for time-domain data*, dans *IEEE Transactions on Signal Processing*, vol. 48, no. 12, p. 3307–3315, 2000.
- [67] M. Arjomandi-Lari et M. Karimi, *Generalized YAST algorithm for signal subspace tracking*, *Signal Processing*, vol. 117, p. 82–95, 2015.
- [68] Y.Miao, Y.Hua, *Fast subspace tracking and neural network learning by a novel information criterion*, *IEEE Transactions, Signal Processing*, vol. 46, no. 7, p. 1967 – 1979, 1998.
- [69] M.Rupp, C.Reyes, *Robust versions of the past algorithm*, *Signal Processing* dans *Proceedings of the 20th European Signal Processing Conference*, p.2313–2317.
- [70] T.D.Nguyen, I.Yamada, *A unified convergence analysis of normalized PAST algorithm for estimating principal and minor components*, *Signal Processing*, vol.93, p.176–184, 2013.
- [71] Y. Hua, Y. Xiang, T. Chen, K. Abed-Meraim, and Y. Miao, *A new look at the power method for fast subspace tracking*, dans *Digital Signal Process*, vol. 9, no. 4, p. 297–314, 1999.
- [72] S. Marcos, A. Marsal, and M. Benidir, *The Propagator method for source bearing estimation*, dans *Signal Processing*, vol. 42, p. 121–138, 1995.
- [73] R. Landqvist, A. Mohammed, *Effect of normalization of eigenvectors on the PAST and RP algorithms for PCA*, dans *Proceedings of the EUSIPCO*, 2005.
- [74] R. Badeau, G. Richard, et B. David, *Suivi d'espace dominant par la méthode des puissances itérées*, *Actes du Colloque GRETSI*, 2003, Paris, France, p.137–140, 2003.
- [75] X.Doukopoulos, G.Moustakides, *Fast and stable subspace tracking*, *IEEE Transactions, Signal Processing*, vol. 56, p.1452–1465, 2008.
- [76] R.Wang, M.Yao, D.Zhang, H.Zou, *A novel orthonormalization matrix based fast and stable dpm algorithm for principal and minor subspace tracking*, *IEEE Transactions, Signal Processing*, vol. 60, p. 466–472, 2012.
- [77] P.Strobach, *The fast recursive row-householder subspace tracking algorithm*, *Signal Processing* vol. 89, p.2514–2528, 2009.
- [78] M.Thameri, K.Abed-Meraim, A.Belouchrani, *Minor subspace tracking using mns technique*, dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 2433–2436, 2012.
- [79] R.Badeau, B.David, G.Richard, *Yet another subspace tracker* dans *IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP'05)*, vol. 4, p. 329–332, 2005.

Bibliographie

- [80] R.Badeau, B.David, G.Richard, *Yast algorithm for minor subspace tracking*, dans *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2006 Proceedings*, vol.3, 2006.
- [81] R. A. Horn and C. R. Johnson, *Matrix Analysis*.Cambridge University Press, 1990. p.185, ISBN: 0521386322.
- [82] R. A. Horn and C. R. Johnson, *Matrix Analysis*.Cambridge University Press, 1990. p.190, ISBN: 0521386322.
- [83] G.H. Golub, C.F.van Loan, *Matrix Computations fourth edition*, Chapter 2, p.82, 2013. ISBN: 1421407949.

Maissa Chagmani

IM2NP, Université de Toulon

Filtre adapté stochastique : poursuite de sous-espace et filtrage multicritère. Application à l'acoustique sous-marine

Résumé en français

Dans ce manuscrit, on s'intéresse à la détection et la classification de signaux aléatoires en utilisant le filtrage adapté stochastique sous contraintes (CSMF) qui maximise le rapport signal à bruit RSB après projection dans un sous-espace dit « optimal » de dimension fixée *a priori*.

Le CSMF peut être appliqué dans les 2 cas suivants : « Détection » quand on a un signal d'intérêt et un bruit et « classification » quand on est en présence de plusieurs perturbations.

Dans ce deuxième cas, on peut envisager plusieurs critères à optimiser : ainsi on présente le filtrage multicritère qui combine plusieurs critères pour améliorer les performances.

Enfin, nous proposons un algorithme adaptatif : le « *Constrained Stochastic Matched Filter Subspace Tracking* » (CSMFST) en s'inspirant des méthodes de poursuite du sous-espace principal, notamment, les algorithmes de type *Subspace Projection* (SP). Le but étant de faire la poursuite du sous-espace optimal en temps réel, dans le cadre de la détection de signaux dont les propriétés statistiques varient dans le temps.

Mot clés : filtrage, tracking, rapport signal-à-bruit, détection, projection, sous-espace, algorithme, multicritère.

Stochastic matched filter: Subspace Tracking and multicriteria filtering. Application to submarine acoustic signals.

Résumé en anglais

In this manuscript, we are interested in the detection and classification of random signals using the Constrained Stochastic Adapted Filter (CSMF), which maximizes the signal-to-noise ratio SNR after projection into an "optimal" subspace of a fixed dimension.

The CSMF can be applied in the following two cases: "Detection" when we have a signal of interest and a noise and "classification" when we work in the presence of several disturbances.

In this second case, we can consider several criteria to optimize: then, we introduce a multicriteria filtering that combines many criteria to improve the performances.

Finally, we propose an adaptive algorithm: the "Constrained Stochastic Matched Filter Subspace Tracking" (CSMFST) taking inspiration from the main subspace tracking methods, in particular, the *Subspace Projection* (SP) type algorithms. The goal is to track the optimal subspace in real time, when signals with statistical properties that vary in time.

Keywords: filtering, tracking, signal-to-noise ratio, detection, projection, subspace, algorithm, multicriteria.