

gen 2005 229

UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

Numéro d'ordre : 2288

THESE

pour obtenir le grade de
**DOCTEUR de L'UNIVERSITE DES
SCIENCES ET TECHNOLOGIES DE LILLE**

Discipline : Electronique
présentée et soutenue publiquement

par
Eric VERRIEST
Ingénieur ISEN
le 18 Juin 1998

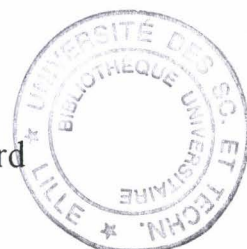
**CONCEPTION ET MISE EN OEUVRE EN TEMPS
REEL D'UN SYSTEME DE SOUSTRACTION DE
BRUIT ADAPTATIF POUR ANTENNE SONAR**

Directeur de thèse :
Jean-Noël DECARPIGNY

JURY :

Monsieur	G. SALMER	Président
Monsieur	J.N. DECARPIGNY	Directeur de thèse
Madame	M.A. GALLAND	Rapporteur
Monsieur	P. WAGSTAFF	Rapporteur
Madame	C. SERVIERE	Examinatrice
Monsieur	A. ROURE	Examineur
Monsieur	C. GIANGRECO	Examineur
Monsieur	J.M. CAPRON	Examineur

Thèse préparée à l'Institut d'Electronique et de Microélectronique du Nord



*A Marie,
pour son amour et sa patience
A Gérard, Ian, Jocelyne, Laurent et Pierre,
pour leur amitié
A mon frère, à ma famille*

REMERCIEMENTS

Ce travail a été réalisé au sein de l'Institut d'Electronique et de Microélectronique du Nord (IEMN) à Lille, en collaboration avec la Direction des Constructions Navales (DCN) de Toulon (Département Lutte Sous-Marine).

Je désire tout d'abord remercier Monsieur le Professeur G. SALMER, Directeur de l'Ecole Doctorale à l'Université des Sciences et Technologies de Lille (USTL), pour l'honneur qu'il me fait en acceptant de présider le jury.

Monsieur P. WAGSTAFF, Professeur à l'Université Technologique de Compiègne (UTC) et Madame M.A. GALLAND, Professeur à l'Ecole Centrale de Lyon, ont bien voulu s'intéresser à ce travail et l'examiner. Je tiens à leur exprimer toute ma gratitude. Merci également à Monsieur A. ROURE, Professeur au Laboratoire de Mécanique et d'Acoustique de Marseille (LMA), pour s'être penché sur ce travail.

Je remercie particulièrement Monsieur J.N. DECARPIGNY, Directeur de l'Institut Supérieur d'Electronique du Nord (ISEN), pour m'avoir accueilli au sein de son laboratoire et avoir accepté d'assurer la direction de ma thèse.

Je suis profondément reconnaissant à Monsieur J.M. CAPRON, Enseignant-Chercheur à l'ISEN, pour avoir encadré mon travail et s'être plongé avec moi dans ce domaine passionnant qu'est le temps réel. Sa disponibilité et son expérience furent indispensables pour mener à bien cette étude. Mes remerciements s'adressent également à Messieurs C. GRANGER et G. HAW, Enseignants-Chercheurs à l'ISEN, sans lesquels la validation expérimentale de la soustraction de bruit n'aurait pu avoir lieu.

Mes remerciements s'adressent aussi à Monsieur C. AUDOLY, Ingénieur à la DCN, qui a initié ce travail, et m'a fourni un support précieux concernant la modélisation du bruit propre. Merci également à Monsieur C. GIANGRECO, Ingénieur à la DCN, pour avoir supervisé ce travail, et à Monsieur B. DUBUS, Enseignant-Chercheur à l'ISEN, pour m'avoir aidé dans la compréhension de la théorie de la vibration des plaques.

Je souhaite adresser des remerciements très sincères à Madame C. SERVIERE, ainsi que Messieurs D. BAUDOIS et A. SILVENT, Professeurs au Centre d'Etude des Phénomènes Aléatoires et Géophysiques de Grenoble (CEPHAG), pour avoir accepté d'assurer un rôle de conseil dans le domaine de la soustraction de bruit adaptative.

Enfin, je tiens sincèrement à remercier tous les membres de l'ISEN. Leur accueil, leur compétence et leur amitié m'ont réellement soutenu et permis de mener à bien ce travail.

RESUME

L'objet de cette étude est la conception et la mise en oeuvre en temps réel d'un système de soustraction de bruit adaptatif, en vue d'une application à la soustraction du bruit propre d'origine mécanique qui se manifeste sur une antenne de sonar passif.

Dans une première partie, un modèle simple de plaque en vibration est présenté, permettant d'une part de mieux comprendre le phénomène de génération du bruit propre d'origine mécanique, d'autre part d'obtenir des signaux temporels simulés qui pourront servir de base pour les simulations des algorithmes de soustraction de bruit adaptatifs.

Dans une seconde partie, nous présentons les différentes méthodes de détermination des algorithmes de soustraction de bruit adaptatifs, tant dans le domaine temporel que dans le domaine fréquentiel. Nous opérons une première sélection parmi ces algorithmes sur la base de la faisabilité de leur mise en oeuvre en temps réel sur un processeur de traitement du signal standard.

Dans une troisième partie, ces algorithmes sont simulés sur des signaux modélisés ainsi que sur des signaux expérimentaux correspondant à des essais en lac ; les performances sont examinées. Nous présentons ensuite la mise en oeuvre en temps réel de ces algorithmes sur un processeur particulier de traitement du signal : le DSP TMS320C31 de Texas Instruments.

La dernière partie fait l'objet d'une validation de ces méthodes en temps réel, sur une structure mécanique représentative de la réalité physique. Nous vérifions les conclusions obtenues dans les chapitres antérieurs, et nous mettons en évidence les conditions de bon fonctionnement de ces algorithmes en temps réel.

Mots clés : bruit propre ; soustraction de bruit ; estimation adaptative ; référence bruit seul ; identification ; processeur de traitement du signal (DSP) ; temps réel

ABSTRACT

This thesis intends to present the design and implementation of a real-time adaptive noise cancellation system, in order to perform mechanical noise cancellation on a passive sonar antenna.

In the first part, we build up a simple model of a vibrating plate in order to have a better understanding of the mechanical noise generation in submarine environment. This model allows us to generate simulated signals in the time domain, which can be used for simulating the noise cancellation algorithms.

In the second part, we present the derivation of time-domain and frequency-domain adaptive noise cancellation techniques. We perform a first selection among those algorithms, based on the feasibility of their real-time implementation on a standard digital signal processor.

In the third part, the selected algorithms are simulated in a high-level language, both on simulated signals and on experimental underwater acoustic data. The performance of these algorithms is evaluated. Then we present their real-time implementation on a specific digital signal processor : the Texas Instruments TMS320C31.

In the last part, we validate the noise cancellation techniques in real time, on a mechanical structure close to the real submarine environment. We verify the foregoing conclusions and highlight the conditions under which the algorithms behave correctly in real time.

Keywords : mechanical noise ; noise cancellation ; adaptive estimation ; noise reference ; identification ; digital signal processor (DSP) ; real time

SOMMAIRE

PRINCIPALES ABBREVIATIONS UTILISEES	-1-
INTRODUCTION	-2-
I. MODELISATION DU BRUIT DE PLAQUE	-5-
I.1 Introduction ; notations utilisées	-6-
I.2 Position du problème	-7-
I.3 Résolution du modèle	-8-
I.3.1 Réponse de la plaque au bruiteur	-8-
I.3.1.1 Excitation rectiligne harmonique	-8-
I.3.1.2 Excitation rectiligne à spectre large bande	-16-
I.3.2 Réponse de la plaque au signal acoustique	-17-
I.3.2.1 Signal acoustique harmonique	-17-
I.3.2.2 Signal acoustique à spectre large bande	-19-
I.4 Résultats de la modélisation	-19-
I.4.1 Génération des stimuli	-19-
I.4.2 Influence de la distance bruiteur-capteurs	-21-
I.4.3 Influence du niveau spectral du bruiteur	-26-
I.5 Conclusion	-27-
I.6 Bibliographie	-28-
Annexe I-1 : Intégration numérique par la méthode de Gauss-Legendre	-28-
II. SOUSTRACTION DE BRUIT	-29-
II.1 Notations et abréviations utilisées	-30-
II.2 Position simplifiée du problème	-32-
II.3 Dérivation des algorithmes temporels : démarche générale	-33-
II.3.1 Choix du modèle pour le filtre adaptatif	-33-
II.3.2 Choix d'une approche de résolution du problème	-33-
II.3.2.1 Filtrage de Wiener	-33-
II.3.2.2 Méthode des Moindres Carrés	-35-
II.3.2.3 Filtrage de Kalman	-36-
II.4 Dérivation des algorithmes fréquentiels : démarche générale	-37-
II.5 Critères de sélection des algorithmes	-37-
II.6 Algorithme LMS transverse	-39-
II.6.1 Méthodes d'optimisation d'une fonction de M variables	-39-
II.6.2 Dérivation de l'algorithme LMS transverse	-42-
II.7 Algorithme Fast Transversal Filter (FTF)	-46-

II.7.1	Dérivation de l'algorithme RLS avec facteur d'oubli exponentiel	-46-
II.7.2	Cadre géométrique pour la dérivation du FTF	-49-
II.7.3	Dérivation de l'algorithme FTF	-51-
II.8	Algorithme du Spectrofiltre adaptatif	-55-
II.8.1	Filtrage adaptatif dans le domaine fréquentiel	-55-
II.8.2	Dérivation du Spectrofiltre adaptatif	-56-
II.8.3	Fonctionnalités supplémentaires	-58-
II.8.3.1	Contrôle de l'efficacité du traitement	-58-
II.8.3.1.1	Contrôle à priori, ou amont	-59-
II.8.3.1.2	Contrôle à postériori, ou aval	-60-
II.8.3.2	Fenêtrage des données	-60-
II.8.3.3	Contrainte du gradient	-62-
II.8.3.3.1	Equivalence du gradient fréquentiel et du spectrofiltre	-62-
II.8.3.3.2	Application de la contrainte au gradient fréquentiel	-63-
II.8.3.3.3	Application de la contrainte au spectrofiltre	-67-
II.8.4	Remarques importantes	-68-
II.9	Conclusion	-68-
II.10	Bibliographie	-69-
III.	SIMULATION DES ALGORITHMES ET MISE EN OEUVRE SUR DSP	-71-
III.1	Simulation des algorithmes	-72-
III.1.1	Simulation sur signaux expérimentaux	-72-
III.1.1.1	Conditions de l'expérimentation	-72-
III.1.1.2	Analyse des données	-72-
III.1.1.3	Méthode d'évaluation des algorithmes	-74-
III.1.1.4	Résultats obtenus	-75-
III.1.1.4.1	Algorithme FTF	-75-
III.1.1.4.2	Algorithme LMS temporel	-77-
III.1.1.4.3	Algorithme du Spectrofiltre adaptatif	-78-
III.1.2	Simulation sur les signaux extraits du modèle	-78-
III.1.2.1	Généralités	-78-
III.1.2.2	Résultats obtenus	-79-
III.1.3	Conclusion	-80-
III.2	Mise en oeuvre sur le DSP TMS320C31	-81-
III.2.1	Choix du système de développement	-81-
III.2.2	Présentation du DSP TMS320C31	-81-
III.2.2.1	Aspects mémoire	-82-
III.2.2.2	Architecture interne	-83-
III.2.2.3	Central Processing Unit	-83-

III.2.2.4 Direct Memory Access	-83-
III.2.2.5 Pipeline	-84-
III.2.2.6 Jeu d'instructions	-84-
III.2.2.7 Modes d'adressage	-85-
III.2.2.8 Outils de développement	-86-
III.2.3 Mise en oeuvre des méthodes fréquentielles	-86-
III.2.3.1 Mise en oeuvre des FFT/IFFT	-86-
III.2.3.2 Mise en oeuvre de la méthode par recouvrement de blocs	-88-
III.2.3.3 Aspects temps réel	-91-
III.2.3.4 Occupation mémoire des algorithmes	-92-
III.2.4 Mise en oeuvre des méthodes temporelles	-94-
III.2.4.1 Contrainte temps réel	-94-
III.2.4.2 Mise en oeuvre des algorithmes LMS	-95-
III.2.4.2.1 Généralités	-95-
III.2.4.2.2 Mise en oeuvre du LMS transverse	-96-
III.3 Conclusion	-97-
III.4 Bibliographie	-97-
Annexe III-1 : Validation du FTF sur signaux expérimentaux	-98-
Annexe III-2 : Validation du Spectrofiltre adaptatif sur signaux expérimentaux	-99-
Annexe III-3 : Validation du FTF sur signaux simulés	-100-
Annexe III-4 : Validation du Spectrofiltre adaptatif sur signaux simulés	-101-
Annexe III-5 : Mise en oeuvre du spectrofiltre adaptatif sur le TMS320C31	-102-
Annexe III-6 : Mise en oeuvre du LMS temporel sur le TMS320C31	-109-
IV. VALIDATION TEMPS REEL EN BASSIN D'ESSAIS ACOUSTIQUES	-113-
IV.1 Introduction	-114-
IV.2 Première validation en bassin d'essais	-114-
IV.2.1 Dispositif expérimental	-114-
IV.2.2 Instrumentation utilisée	-116-
IV.2.3 Analyse des signaux	-117-
IV.2.4 Résultats obtenus en soustraction de bruit	-120-
IV.2.4.1 Algorithme du spectrofiltre adaptatif	-120-
IV.2.4.2 Algorithme LMS temporel	-122-
IV.2.5 Interprétation des résultats obtenus	-122-
IV.2.5.1 Influence de la puissance du signal utile	-123-
IV.2.5.2 Modifications algorithmiques envisagées	-125-
IV.3 Seconde validation des algorithmes	-126-
IV.3.1 Validation sur signaux déterministes	-126-
IV.3.1.1 Généralités ; filtre à identifier	-126-

IV.3.1.2	Contrôle amont et fenêtrage des données	-127-
IV.3.1.2.1	Analyse des signaux	-127-
IV.3.1.2.2	Fenêtrage rectangulaire (naturel)	-128-
IV.3.1.2.3	Fenêtrage de Hamming	-133-
IV.3.1.3	Influence de la puissance du signal utile	-136-
IV.3.1.3.1	Analyse des signaux	-137-
IV.3.1.3.2	Résultats de la soustraction de bruit	-137-
IV.3.1.4	Influence du contrôle aval sur la soustraction de bruit	-140-
IV.3.1.5	Reconstitution du signal dans le domaine temporel	-141-
IV.3.2	Seconde validation en bassin d'essais acoustiques	-143-
IV.3.2.1	Dispositif expérimental	-143-
IV.3.2.2	Instrumentation utilisée	-143-
IV.3.2.3	Résultats obtenus en soustraction de bruit	-144-
IV.3.2.3.1	Analyse des signaux	-144-
IV.3.2.3.2	Algorithme du spectrofiltre adaptatif	-146-
IV.3.2.3.3	Algorithme du gradient fréquentiel	-149-
IV.3.2.3.4	Algorithme LMS temporel	-149-
IV.3.2.3.5	Influence de la puissance du signal utile	-150-
IV.4	Conclusion	-151-
IV.5	Bibliographie	-152-
	CONCLUSION	-153-
	REFERENCES BIBLIOGRAPHIQUES	-155-

PRINCIPALES ABBREVIATIONS UTILISEES

RIF	Réponse Impulsionnelle Finie
RII	Réponse Impulsionnelle Infinie
TFD	Transformée de Fourier Discrète
TFR (FFT)	Transformée de Fourier Rapide (Fast Fourier Transform)
IFFT	Inverse Fast Fourier Transform
RBS	Référence Bruit Seul
EQM	Erreur Quadratique Moyenne
EQMR	Erreur Quadratique Moyenne Résiduelle
LMS	Least Mean Square
RLS	Recursive Least Squares
LSL	Least Squares Lattice
FTF	Fast Transversal Filter
FAEST	Fast A Posteriori Error Sequential Technique
PARCOR	PARTial CORrelation
SDB	Soustraction De Bruit
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
ASIC	Application Specific Integrated Circuit
DSP	Digital Signal Processor
MIPS	Million Instructions Per Second
MFLOPS	Million FLoating-Point Operations Per Second
RAM	Random Access Memory
SRAM	Static Random Access Memory
DPRAM	Dual-Port Random Access Memory
DMA	Direct Memory Access
CPU	Central Processing Unit
ALU	Arithmetic and Logic Unit
ARAU	Auxiliary Register Arithmetic Unit
PC	Personal Computer
NOP	No OPERATION

INTRODUCTION

L'extraction de l'information contenue dans un signal bruité constitue un problème majeur en traitement du signal. Les premières études portant sur l'élimination de bruit ont été menées par Wiener [*Wiener 49*], elles ont été reprises et élargies notamment par Kalman, Bucy et Kailath [*Kalman 60*][*Kailath 74*]. D'une manière générale, la soustraction de bruit se fait par filtrage adaptatif, les filtres adaptatifs présentant l'avantage de pouvoir ajuster leurs propres paramètres automatiquement, si bien que leur mise en oeuvre ne nécessite quasiment aucune connaissance a priori des caractéristiques du signal et du bruit perturbateur. La soustraction de bruit repose sur l'utilisation d'un ou plusieurs signaux de référence représentatifs du bruit à éliminer. Ces signaux sont appelés références bruit ; ils sont mesurés par un ou plusieurs capteurs disposés dans le champ de bruit à des emplacements où le signal utile est extrêmement faible ou indétectable. Le lien statistique entre la référence bruit et le bruit perturbateur présent sur la voie signal se traduit par l'existence d'un filtrage entre les deux signaux. La soustraction de bruit peut dès lors se ramener au problème de l'identification d'un filtre permettant de reconstruire le bruit perturbateur présent sur la voie signal à partir de la référence bruit. Il suffit ensuite de soustraire au signal bruité l'estimation ainsi construite du bruit perturbateur.

Plus récemment, de nombreuses études portant sur l'élaboration et l'évaluation d'algorithmes performants pour la soustraction de bruit en présence de référence bruit ont été menées, dans le cas où la référence bruit est unique (ou scalaire) [*Baudois 81*][*Arques 82*][*Servière 89*], ainsi que dans le cas multiréférences-bruit [*Guerre-Chaley 90*].

L'une des premières mises en oeuvre de la soustraction de bruit adaptative a été réalisée par Howells et Applebaum entre 1957 et 1960 à la General Electric Company. Leurs travaux ont porté sur la conception et la réalisation d'une antenne avec annulation des lobes secondaires qui utilisait une référence bruit provenant d'une antenne auxiliaire et un filtre adaptatif à deux coefficients. Dans un tout autre domaine que le traitement d'antenne, Huhta et Webster ont mis au point à l'Université de Stanford en 1965 le premier système de soustraction de bruit adaptatif. Ce système visait à éliminer une interférence 60 Hz (fréquence du réseau électrique des Etats-Unis) sur un ensemble d'amplification et d'enregistrement d'électrocardiogrammes. Depuis 1965, la théorie de la soustraction de bruit adaptative a été appliquée avec succès à un nombre croissant d'applications, telles que l'élimination d'interférences périodiques en général, l'élimination d'échos sur lignes téléphoniques, et plus récemment la réduction de bruit en acoustique sous-marine (traitement d'une voie audio de sonar et élimination du bruit du bateau tracteur sur les sorties de capteurs d'une antenne linéaire remorquée [*Servière 89*][*Guerre-Chaley 90*]).

Notre étude s'inscrit dans l'application de la soustraction de bruit adaptative au traitement de signaux sonar. Les antennes de réception d'un sonar passif ont pour fonction de déterminer la direction d'une onde plane correspondant à un signal acoustique large bande de faible puissance, provenant d'une cible à détecter. Or les performances de ces antennes implantées sur un navire porteur -bâtiment de surface ou sous-marin- sont affectées par le bruit propre, qui vient s'ajouter au bruit ambiant du milieu marin. Ce bruit propre est provoqué par trois phénomènes principaux :

- le bruit du propulseur, rayonné par voie acoustique ou transmis par propagation le long de la coque du navire porteur ;
- le bruit hydrodynamique, provoqué par les fluctuations de pression liées à l'écoulement turbulent du fluide le long de la carène ;
- le bruit mécanique, provoqué par le rayonnement acoustique en champ proche des structures du navire, excitées par les machines.

Cette étude a pour but d'étudier la faisabilité d'un dispositif permettant de réduire en temps réel le bruit propre d'origine mécanique, qui se manifeste sur les capteurs d'antennes sonar (hydrophones) dans la bande fréquentielle 0-6KHz. En effet, des expériences récentes ont montré que dans certaines conditions d'implantation et pour certains porteurs, cette composante de bruit peut être prédominante. L'aspect temps réel de la réduction de bruit est très important, car dans le cas des systèmes sonar il faut détecter le plus tôt possible l'éventuel ennemi.

Une solution courante pour réduire le bruit propre consiste à utiliser un algorithme de soustraction de bruit au niveau du traitement d'antenne. Pour cela, il faut disposer de signaux représentatifs du bruit perturbateur à éliminer, ce qui peut être réalisé en utilisant par exemple des accéléromètres (capteurs de référence bruit) disposés sur la coque ou sur les machines les plus bruyantes. Cette méthode a cependant quelques inconvénients, car elle est presque entièrement empirique ; de plus, son efficacité est difficile à quantifier, et il n'existe pas actuellement de méthode fiable pour positionner les capteurs de référence bruit.

La solution développée dans cette étude consiste à utiliser des capteurs "intelligents" permettant une réduction locale du bruit propre, c'est-à-dire permettant d'épurer directement le signal mesuré par un hydrophone de sa composante de bruit propre. Le principe de base consiste à exploiter la corrélation entre la vibration de la structure mécanique supportant l'hydrophone (cette vibration étant mesurée par un accéléromètre placé sur la structure en vis-à-vis de l'hydrophone) et la pression acoustique rayonnée en champ proche, en vue de soustraire le bruit mécanique grâce à des algorithmes de filtrage adaptatif. Nous étudierons la faisabilité de cette solution, principalement sur le plan de la mise en oeuvre en temps réel sur un processeur spé-

cialisé de traitement du signal (Digital Signal Processor, ou DSP) à faible coût. Une validation en temps réel de cette solution sera menée en bassin d'essais acoustiques sur une structure mécanique proche de la réalité physique.

Le rapport de thèse est organisé comme suit :

- Le premier chapitre fait l'objet de la mise au point d'un modèle simple de plaque en vibration, afin de simuler les signaux mesurés par un hydrophone et un accéléromètre placés en vis-à vis sur cette plaque. En première approximation, la structure mécanique supportant l'antenne sonar est modélisée par une plaque élastique de dimensions infinies. Ceci permet de mieux comprendre la génération du bruit de plaque et en même temps de générer des signaux temporels qui serviront de base pour les simulations des algorithmes de soustraction de bruit.
- Dans le second chapitre, nous présentons les différentes méthodes de détermination des algorithmes de soustraction de bruit existant dans la littérature. Nous opérons une première sélection parmi ces algorithmes sur la base de la faisabilité de leur mise en oeuvre en temps réel sur un DSP standard. Nous dérivons ensuite les algorithmes ainsi sélectionnés, en présentant leurs avantages et inconvénients respectifs.
- Dans le troisième chapitre, les algorithmes les plus prometteurs compatibles avec la contrainte de mise en oeuvre en temps réel sont simulés dans un langage haut niveau (C) sur des signaux expérimentaux, ainsi que sur les signaux modélisés dans le premier chapitre ; les performances ainsi que les qualités de convergence et de stabilité à long terme sont examinées. Enfin, nous présentons la mise en oeuvre en temps réel de ces algorithmes sur le processeur de traitement du signal TMS320C31 de Texas Instruments.
- Le dernier chapitre fait l'objet d'une première validation en temps réel des algorithmes en bassin d'essais acoustiques, sur une structure représentative de la réalité physique. Les résultats sont présentés en termes de gain de traitement et de chute de cohérence entre la voie signal et la référence bruit seul. Les modifications algorithmiques à apporter en vue d'une amélioration des résultats sont présentées. Une seconde validation des algorithmes incorporant les modifications précédentes est menée en bassin d'essais acoustiques sur la même structure physique. Les améliorations par rapport à la première expérimentation sont mises en évidence.

Il est à noter que nous utilisons dans tout le texte le terme de bruiteur pour désigner la perturbation par rapport au signal utile (les sonaristes emploient usuellement ce terme pour désigner la cible à détecter).



I

MODELISATION DU BRUIT DE PLAQUE

I. MODELISATION DU BRUIT DE PLAQUE

I.1. INTRODUCTION ; NOTATIONS UTILISEES

Dans ce chapitre, nous allons modéliser la structure supportant une antenne de sonar passif sous la forme d'une plaque infinie immergée, soumise à une force vibratoire modélisant une source de bruit mécanique, et en même temps éclairée par un signal acoustique incident, représentant un signal utile à détecter. Le but de cette modélisation est d'une part de simuler les signaux temporels mesurés par un doublet de capteurs hydrophone (voie signal) - accéléromètre (référence bruit seul) placés en vis-à-vis sur cette plaque, d'autre part d'arriver à une meilleure compréhension du bruit de plaque. Cette modélisation a déjà été partiellement abordée dans le cas où l'excitation vibratoire est rectiligne sur la plaque [Feit et Liu 85] [Junger et Feit 86]. Les modélisations usuelles font l'approximation d'un bruiteur et d'un signal acoustique harmoniques ; nous étendons ici ces modélisations au cas d'un bruiteur et d'un signal acoustique à spectre large bande en fréquence. Nous travaillons sous la fréquence de coïncidence, de manière à pouvoir utiliser l'approximation des plaques minces [Junger et Feit 86].

Nous utiliserons dans ce chapitre les notations et valeurs numériques suivantes :

S_h	sensibilité de l'hydrophone en pression ($S_h = 10^{-4} \text{ V.Pa}^{-1}$)
S_{ha}	sensibilité de l'hydrophone en accélération ($S_{ha} = 10^{-5} \text{ V/m.s}^{-2}$)
S_{aa}	sensibilité de l'accéléromètre en accélération ($S_{aa} = 10^{-2} \text{ V/m.s}^{-2}$)
E	module d'Young de la plaque ($E = 2.1 \cdot 10^{11}$)
ν	coefficient de Poisson de la plaque ($\nu = 0.285$)
ρ_s	masse volumique de la plaque ($\rho_s = 7800 \text{ kg.m}^{-3}$)
h	épaisseur de la plaque ($h = 2.54 \text{ cm}$)
m	masse par unité de surface de la plaque ($m = \rho_s \cdot h$)
f_c	fréquence de coïncidence de la plaque ($f_c = 9031 \text{ Hz}$)
c_0	célérité du son dans le fluide ($c_0 = 1500 \text{ m.s}^{-1}$)
ρ_0	masse volumique du fluide ($\rho_0 = 1030 \text{ kg.m}^{-3}$)
ϵ	paramètre de charge du fluide ($\epsilon = \rho_0 \cdot c_0 / m \cdot \omega_c = 0.137$)
x_0	distance entre le bruiteur et les capteurs
θ	angle d'incidence du signal acoustique par rapport à la normale à la plaque
$U_{\text{hydro}}(x,z,t)$	tension électrique mesurée par l'hydrophone à l'instant t
$U_{\text{acc}}(x,z,t)$	tension électrique mesurée par l'accéléromètre à l'instant t
$w_{\text{vib}}(x,z,t)$	déplacement transverse de la plaque généré par le bruiteur
$\gamma_{\text{vib}}(x,z,t)$	accélération transverse de la plaque générée par le bruiteur
$P_{\text{vib}}(x,z,t)$	pression rayonnée en champ proche dans le fluide à cause du bruiteur
$w_{\text{ac}}(x,z,t)$	déplacement transverse de la plaque généré par le signal acoustique incident
$\gamma_{\text{ac}}(x,z,t)$	accélération transverse de la plaque générée par le signal acoustique incident

$P_{ac}(x,z,t)$ pression résultante en champ proche dans le fluide générée par le signal acoustique incident

$$\tilde{w}(k_x, z, t) = \int_{-\infty}^{\infty} w(x, z, t) \cdot \exp(-i \cdot k_x \cdot x) dx \quad \text{Transformée de Fourier spatiale en } x$$

$$w(x, z, t) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} \tilde{w}(k_x, z, t) \cdot \exp(i \cdot k_x \cdot x) dk_x \quad \text{Transformée de Fourier spatiale inverse en } k_x$$

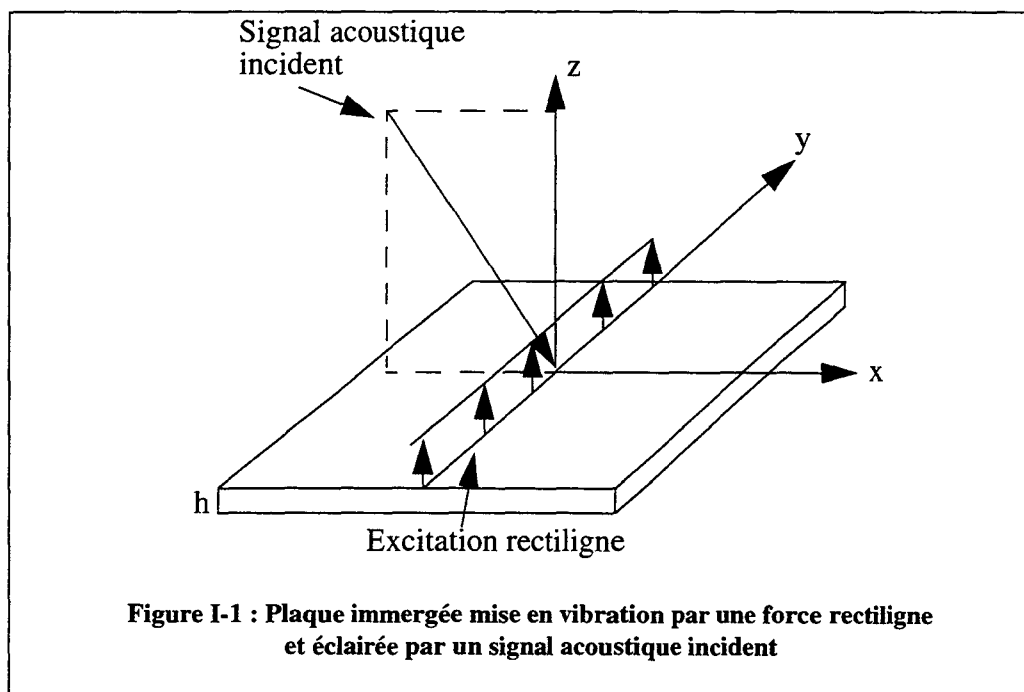
$$\widehat{w}(x, z, \omega) = \int_{-\infty}^{\infty} w(x, z, t) \cdot \exp(-i \cdot \omega \cdot t) dt \quad \text{Transformée de Fourier temporelle}$$

$$w(x, z, t) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} \widehat{w}(x, z, \omega) \cdot \exp(i \cdot \omega \cdot t) d\omega \quad \text{Transformée de Fourier inverse temporelle}$$

L'analyse fréquentielle est faite par Transformée de Fourier Rapide (TFR) sur 8192 points, avec une fréquence d'échantillonnage de 12 KHz pour satisfaire au théorème de Shannon appliqué à la modélisation de signaux dans la bande 0-6KHz.

I.2. POSITION DU PROBLEME

Nous allons considérer au cours de cette modélisation une plaque d'acier d'épaisseur h , de dimensions infinies dans le plan $z = 0$. Un fluide de masse volumique ρ_0 dans lequel le son se propage à la vitesse c_0 occupe le demi-plan $z > 0$, et de l'air remplit le demi-plan $z < 0$, comme représenté sur la Figure I-1.



Cette plaque est soumise à une force rectiligne large bande sur l'axe y , placée en $x = 0$. On prend également en compte un signal acoustique large bande incident, dans un plan y donné, ce qui rend le problème indépendant de la variable y . On place sur cette plaque, à une distance x_0 de l'origine, un doublet de capteurs hydrophone (voie signal) - accéléromètre (référence bruit)

seul) en vis-à vis l'un de l'autre. Le but de ce modèle est de simuler les tensions électriques mesurées par ces deux capteurs en fonction du temps.

En première approximation, on écrit ces tensions électriques comme des combinaisons linéaires des contributions suivantes (on néglige l'épaisseur de la plaque) : $\gamma_{vib}(x_0, 0, t)$, $P_{vib}(x_0, 0, t)$, $\gamma_{ac}(x_0, 0, t)$, $P_{ac}(x_0, 0, t)$.

$$U_{hydro}(x_0, 0, t) = S_h \cdot (P_{vib}(x_0, 0, t) + P_{ac}(x_0, 0, t)) + S_{ha} \cdot (\gamma_{vib}(x_0, 0, t) + \gamma_{ac}(x_0, 0, t)) \quad (\text{EQ I-1})$$

$$U_{acc}(x_0, 0, t) = S_{aa} \cdot (\gamma_{vib}(x_0, 0, t) + \gamma_{ac}(x_0, 0, t)) \quad (\text{EQ I-2})$$

L'objet des paragraphes suivants est le calcul des différents termes intervenant dans les équations (EQ I-1) et (EQ I-2).

I.3. RESOLUTION DU MODELE

La résolution du modèle peut se décomposer en deux parties ; en effet, nous pouvons superposer la réponse de la plaque au bruiteur et sa réponse au signal acoustique incident. Dans chaque partie, le cas harmonique sera considéré avant la généralisation au cas large bande en fréquence.

I.3.1. REPOSE DE LA PLAQUE AU BRUITEUR

Dans ce paragraphe, nous rappelons les résultats de Feit et Liu [Feit et Liu 85] obtenus pour les termes $w_{vib}(x, 0, t)$ et $P_{vib}(x, 0, t)$ dans le cadre d'une excitation rectiligne harmonique de la forme $F \cdot \delta(x) \cdot \exp(-i\omega t)$, puis nous généralisons ces résultats pour une excitation rectiligne $F(t) \cdot \delta(x)$ à spectre large bande.

I.3.1.1. EXCITATION RECTILIGNE HARMONIQUE

Compte tenu de l'indépendance du problème en y, l'équation des plaques régissant le déplacement transverse $w_{vib}(x, 0, t)$ s'écrit :

$$D \cdot \frac{\partial^4 w_{vib}}{\partial x^4} + m \cdot \frac{\partial^2 w_{vib}}{\partial t^2} = F \cdot \delta(x) \cdot \exp(-i\omega t) - P_{vib}(x, 0, t) \quad (\text{EQ I-3})$$

avec : $D = E \cdot h^3 / 12 \cdot (1 - \nu^2)$, E et ν représentant respectivement le module d'Young et le coefficient de Poisson de la plaque ; $m = \rho_s \cdot h$ représente sa masse par unité de surface. Par souci de concision, nous supprimons désormais dans ce paragraphe le facteur de dépendance temporelle harmonique $\exp(-i\omega t)$, commun à toutes les variables. L'équation d'Euler nous permet alors de lier la pression rayonnée en champ proche $P_{vib}(x, 0)$ au déplacement transverse de la plaque :

$$\left(\frac{\partial P_{vib}}{\partial z} \right)_{z=0} = -\rho_0 \cdot \frac{\partial^2 w_{vib}}{\partial t^2} = \rho_0 \cdot \omega^2 \cdot w_{vib}(x, 0) \quad (\text{EQ I-4})$$

Par ailleurs, la pression $P_{vib}(x, z)$ doit satisfaire à l'équation de Helmholtz :

$$\frac{\partial^2 P_{vib}}{\partial x^2} + \frac{\partial^2 P_{vib}}{\partial z^2} + k_0^2 \cdot P_{vib} = 0 \quad (\text{EQ I-5})$$

où $k_0 = \omega/c_0$ représente le nombre d'onde dans le fluide.

L'approche classique pour un tel problème est d'utiliser une transformée de Fourier spatiale des équations (EQ I-3), (EQ I-4) et (EQ I-5) dans l'espace des nombres d'onde k_x . La transformée de Fourier de l'équation (EQ I-3) donne l'équation (EQ I-3') : $D(k_x^4 - k_f^4) \cdot \tilde{w}_{vib}(k_x, 0) = F - \tilde{P}_{vib}(k_x, 0)$ avec : $k_f^4 = m \cdot \omega^2 / D$.

De même, la transformée de Fourier de l'équation (EQ I-4) donne l'équation (EQ I-4') :

$$\rho_0 \cdot \omega^2 \cdot \tilde{w}_{vib}(k_x, 0) = \left(\frac{\partial \tilde{P}_{vib}(k_x, z)}{\partial z} \right)_{z=0}$$

Cette dernière équation traduit le couplage fluide-structure. Enfin, la transformée de Fourier de l'équation (EQ I-5) donne l'équation (EQ I-5') :

$$\frac{\partial^2 \tilde{P}_{vib}(k_x, z)}{\partial z^2} + (k_0^2 - k_x^2) \cdot \tilde{P}_{vib}(k_x, z) = 0$$

L'équation caractéristique associée à l'équation (EQ I-5') admet des racines imaginaires pures ou réelles, suivant que l'inégalité $|k_x| > k_0$ est vérifiée ou non :

- si $|k_x| > k_0$: on a 2 racines réelles +/- $(k_x^2 - k_0^2)^{1/2}$ et la solution de l'équation (EQ I-5') s'écrit :

$$\tilde{P}_{vib}(k_x, z) = A_1 \exp\left(\left(k_x^2 - k_0^2\right)^{1/2} \cdot z\right) + B_1 \exp\left(-\left(k_x^2 - k_0^2\right)^{1/2} \cdot z\right)$$

- dans le cas contraire, les 2 racines +/- $i(k_0^2 - k_x^2)^{1/2}$ sont imaginaires pures et la solution s'écrit :

$$\tilde{P}_{vib}(k_x, z) = A_2 \exp\left(i\left(k_0^2 - k_x^2\right)^{1/2} \cdot z\right) + B_2 \exp\left(-i\left(k_0^2 - k_x^2\right)^{1/2} \cdot z\right)$$

où A_1, A_2, B_1 et B_2 sont des constantes d'intégration. Dans le premier cas, on choisit $A_1 = 0$ pour que la solution reste bornée quand z tend vers l'infini ; dans le second cas, on choisit $B_2 = 0$ car la pression se propage de la plaque vers le fluide, c'est-à-dire vers les z positifs. Enfin, la continuité de la pression en $k_x = k_0$ impose $B_1 = A_2$. D'une manière générale, on remarque que la solution de l'équation (EQ I-5') peut s'écrire sous la forme suivante, pour toute valeur de k_x :

$$\tilde{P}_{vib}(k_x, z) = K \cdot \exp\left(i\left(k_0^2 - k_x^2\right)^{1/2} \cdot z\right) = K \cdot \exp(i \cdot k_z \cdot z) \quad (\text{EQ I-6})$$

En effet, si $|k_x| > k_0$, k_z est imaginaire pur ; pour que la solution reste bornée quand z tend vers l'infini, on écrit :

$$i \cdot k_z = -\left(k_x^2 - k_0^2\right)^{1/2}$$

En injectant l'équation (EQ I-6) dans l'équation (EQ I-4'), on obtient :

$$K = \frac{-i \cdot \omega^2 \cdot \rho_0}{\sqrt{k_0^2 - k_x^2}} \cdot \tilde{w}_{vib}(k_x, 0) = \tilde{P}_{vib}(k_x, 0) \quad (\text{EQ I-7})$$

On peut alors remplacer $P_{vib}(k_x, 0)$ dans l'équation (EQ I-3') par sa valeur dans l'équation (EQ I-7), d'où il vient l'équation fondamentale :

$$\left(D(k_x^4 - k_f^4) - \frac{i \cdot \omega^2 \cdot \rho_0}{\sqrt{k_0^2 - k_x^2}} \right) \tilde{w}_{vib}(k_x, 0) = F \quad (\text{EQ I-8})$$

Feit et Liu suggèrent d'exprimer l'équation (EQ I-8) d'une façon plus générale, en faisant intervenir les opérateurs d'impédance acoustique spécifique du fluide $\tilde{Z}_a(k_x)$ et d'impédance de plaque $\tilde{Z}_p(k_x)$, définis par les relations suivantes :

$$\tilde{Z}_a(k_x) = \frac{\omega \cdot \rho_0}{\sqrt{k_0^2 - k_x^2}} \quad (\text{EQ I-9})$$

$$\tilde{Z}_p(k_x) = -i \cdot \omega \cdot m \left(1 - \frac{k_x^4}{k_f^4} \right) \quad (\text{EQ I-10})$$

A l'aide de ces opérateurs, l'équation (EQ I-8) s'écrit de façon plus concise :

$$\tilde{w}_{vib}(k_x, 0) = \frac{i \cdot F}{\omega} (\tilde{Z}_a(k_x) + \tilde{Z}_p(k_x))^{-1} \quad (\text{EQ I-11})$$

En utilisant la relation de couplage fluide-structure définie par l'équation (EQ I-7), on obtient également la transformée de Fourier spatiale de la pression rayonnée en champ proche dans le fluide par l'excitation sur la plaque, évaluée en $z = 0$:

$$\tilde{P}_{vib}(k_x, 0) = \frac{F \cdot \tilde{Z}_a(k_x)}{\tilde{Z}_a(k_x) + \tilde{Z}_p(k_x)} \quad (\text{EQ I-12})$$

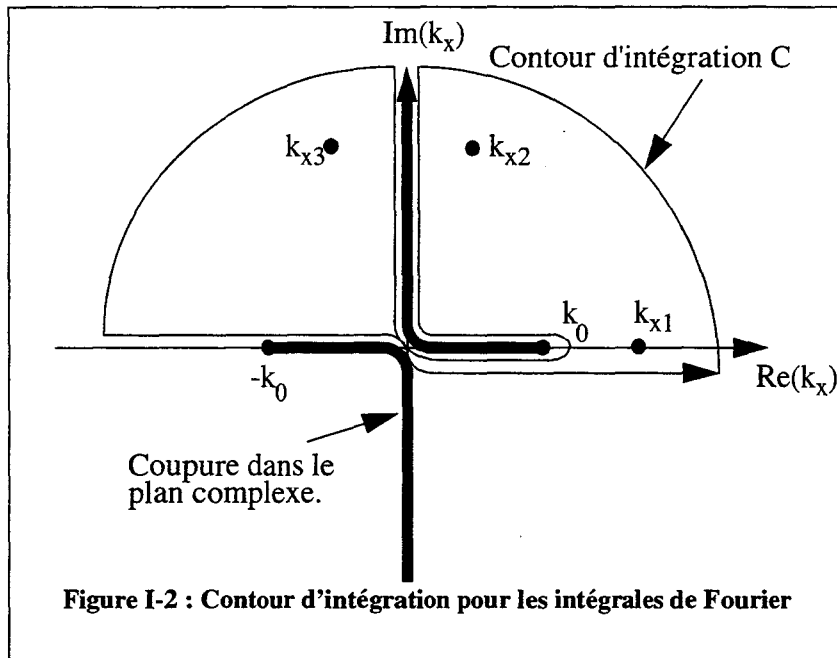
Il est intéressant à ce niveau de faire le lien avec l'impédance acoustique spécifique du fluide. Si $|k_x| > k_0$, on note que l'impédance acoustique est imaginaire pure, ce qui signifie que l'on n'a pas de dissipation d'énergie acoustique dans le fluide, c'est-à-dire pas de rayonnement en champ lointain : les solutions de l'équation caractéristique font intervenir des exponentielles réelles. Dans le cas contraire ($|k_x| < k_0$), l'impédance acoustique est réelle, ce qui correspond à un rayonnement d'énergie acoustique en champ lointain dans le fluide : les solutions de l'équation caractéristique font alors intervenir des exponentielles imaginaires.

Par transformée de Fourier spatiale inverse des équations (EQ I-11) et (EQ I-12), on obtient le déplacement transverse et la pression à la surface de la plaque en fonction de la distance x à l'origine (la dépendance temporelle reste bien sûr harmonique) :

$$w_{vib}(x, 0) = \frac{i \cdot F}{2\pi \cdot \omega} \int_{-\infty}^{\infty} \frac{\exp(i \cdot k_x \cdot x)}{\tilde{Z}_a(k_x) + \tilde{Z}_p(k_x)} dk_x \quad (\text{EQ I-13})$$

$$P_{vib}(x, 0) = \frac{F}{2\pi} \int_{-\infty}^{\infty} \frac{\tilde{Z}_a(k_x) \cdot \exp(i \cdot k_x \cdot x)}{\tilde{Z}_a(k_x) + \tilde{Z}_p(k_x)} dk_x \quad (\text{EQ I-14})$$

Les intégrations seront réalisées sur un contour (noté C) du plan complexe k_x en vue d'utiliser la méthode des résidus, comme l'indique la Figure I-2. On considère les fonctions à intégrer dans les équations (EQ I-11) et (EQ I-12) comme des fonctions complexes de la variable k_x . Le choix du demi-plan supérieur repose sur les conditions de décroissance des fonctions à intégrer quand x tend vers l'infini par valeurs positives. Par ailleurs, $k_z = (k_0^2 - k_x^2)^{1/2}$ est fonction de la variable complexe k_x ; on choisit pour k_z la détermination selon laquelle la partie réelle de k_z est strictement positive lorsque $k_0 > |k_x|$. Enfin, pour satisfaire aux conditions de décroissance de la pression à l'infini, il faut que la partie imaginaire de k_z soit strictement positive. Ceci justifie les coupures choisies dans le plan complexe, sur lesquelles la partie imaginaire de k_z est nulle, ce qui correspondrait à un champ de pression qui se propage à l'infini sans atténuation. Le contour d'intégration est choisi le long de l'axe réel, légèrement déplacé de manière à éviter les racines réelles de l'équation de dispersion (EQ I-15) définie ci-après. Il est fermé par un demi-cercle de rayon infini dans le demi-plan supérieur k_x , qui s'interdit de traverser les coupures définies précédemment.



Le dénominateur des intégrandes est une fonction irrationnelle de k_x , qui possède dix racines possibles une fois rendue rationnelle. La résolution de l'équation de dispersion :

$$\tilde{Z}_a(k_x) + \tilde{Z}_p(k_x) = 0 \quad (\text{EQ I-15})$$

revient par conséquent à chercher les racines d'un polynôme d'ordre 10 en k_x :

$$k_x^{10} - k_0^2 \cdot k_x^8 - 2k_f^4 \cdot k_x^6 + 2k_0^2 \cdot k_f^4 \cdot k_x^4 + k_f^8 \cdot k_x^2 - k_f^8 (k_0^2 + \rho_f^2 / m^2) = 0$$

On peut montrer que ce polynôme admet deux racines réelles +/- k_{x1} , avec $k_{x1} > k_0$ (voir Figure I-2). En effet, en posant $Z = k_x^2$ on obtient un polynôme $P(Z)$ de degré 5 :

$$Z^5 - k_0^2 \cdot Z^4 - 2k_f^4 \cdot Z^3 + 2k_0^2 \cdot k_f^4 \cdot Z^2 + k_f^8 \cdot Z - k_f^8 (k_0^2 + \rho_f^2 / m^2) = P(Z) = 0$$

Ce polynôme étant d'ordre impair, il admet au moins une racine réelle. De plus, $P(0) < 0$; cette racine est donc positive et fournit $\pm k_{x1}$; enfin, on montre facilement que $P(k_0) < 0$, ce qui impose l'inégalité $k_{x1} > k_0$. Junger et Feit [Junger et Feit 86] ont montré que l'équation de dispersion peut s'écrire plus simplement en utilisant la variable intermédiaire $\zeta = (k_x^2 - k_0^2)^{1/2}$. En effet, on aboutit alors à un polynôme de degré 5 qui s'écrit :

$$\zeta^5 + 2k_0^2 \cdot \zeta^3 + (k_0^4 - k_f^4) \zeta - \frac{\rho_f \cdot \omega^2}{D} = 0$$

Des cinq racines possibles de ce polynôme, on ne retient que les trois dont la partie réelle est strictement positive : ceci revient à imposer que la partie imaginaire de k_z soit strictement positive. On peut alors remonter aux trois pôles correspondants en k_x , racines de l'équation de dispersion (EQ I-15). Ces trois pôles (notés k_{x1} , k_{x2} et k_{x3}) sont contenus dans le contour d'intégration C (voir Figure I-2) : ils vont par conséquent contribuer à l'évaluation des intégrales définies par les équations (EQ I-13) et (EQ I-14). En plus des résidus associés à ces trois pôles, il faudra tenir compte des intégrales le long des coupures définies précédemment dans le plan complexe.

L'équation (EQ I-14) généralisée dans le plan complexe s'écrit :

$$P_{vib}(x, 0) = \frac{F}{2\pi} \int_C \frac{\exp(i \cdot k_x \cdot x)}{1 + \frac{m}{\rho_0} \left(1 - \frac{k_x^4}{k_f^4} \right) \sqrt{k_x^2 - k_0^2}} dk_x$$

Il faut définir précisément la racine carrée intervenant au dénominateur en fonction du chemin d'intégration suivi (voir Figure I-3). Pour cela, on décompose l'expression $k_x^2 - k_0^2$ sous la forme $(k_x - k_0)(k_x + k_0)$ et on étudie l'évolution de l'argument de la racine carrée en suivant le chemin d'intégration défini sur la Figure I-3 : a-b-c-d-e-f-g-h-i-j-k-l-m-n.

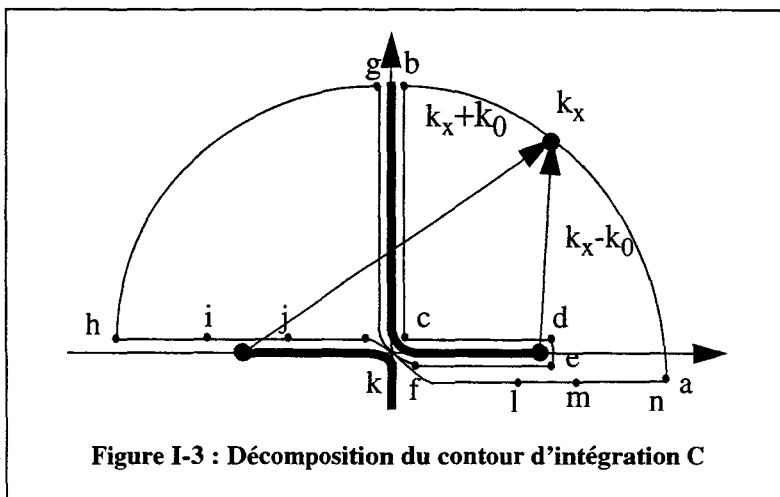


Figure I-3 : Décomposition du contour d'intégration C

TABLEAU 1. Evolution de l'argument de la racine carrée le long du chemin d'intégration C

	a	b	c	d	e	f	g	h	i	j	k	l	m	n
$k_x - k_0$	0	$\pi/2$	π	π	$-\pi$	$-\pi$	$-3\pi/2$	$-\pi$	$-\pi$	$-\pi$	$-\pi$	$-\pi$	0	0
$k_x + k_0$	0	$\pi/2$	0	0	0	0	$\pi/2$	π	π	0	0	0	0	0
produit	0	π	π	π	$-\pi$	$-\pi$	$-\pi$	0	0	$-\pi$	$-\pi$	$-\pi$	0	0
racine	0	$\pi/2$	$\pi/2$	$\pi/2$	$-\pi/2$	$-\pi/2$	$-\pi/2$	0	0	$-\pi/2$	$-\pi/2$	$-\pi/2$	0	0

La dernière ligne de ce tableau nous donne l'argument de la racine carrée en fonction de la branche parcourue sur le contour (C). Il reste simplement à paramétrer les intégrales sur les morceaux [bc], [cd], [ef] et [fg] ; en effet, on montre facilement en utilisant les lemmes de Jordan que les contributions sur les arcs de cercle (ab), (de) et (gh) tendent vers 0 quand on fait tendre le rayon vers l'infini (respectivement : vers 0 pour l'arc (de)).

- Sur [bc] le paramétrage s'écrit (1) =
$$\int_{\infty}^0 \frac{e^{ix \cdot iB}}{1 + \frac{i \cdot m}{\rho_0} \left(1 - \frac{i^4 \cdot B^4}{k_f^4}\right) \sqrt{B^2 + k_0^2}} i \cdot dB$$

- Sur [cd] le paramétrage s'écrit (2) =
$$\int_0^{k_0} \frac{e^{i \cdot x \cdot A}}{1 + \frac{i \cdot m}{\rho_0} \left(1 - \frac{A^4}{k_f^4}\right) \sqrt{k_0^2 - A^2}} dA$$

- Sur [ef] le paramétrage s'écrit (3) =
$$-\int_0^{k_0} \frac{e^{i \cdot x \cdot A}}{1 - \frac{i \cdot m}{\rho_0} \left(1 - \frac{A^4}{k_f^4}\right) \sqrt{k_0^2 - A^2}} dA$$

- Sur [fg] le paramétrage s'écrit (4) =
$$-\int_{\infty}^0 \frac{e^{ix \cdot iB}}{1 - \frac{i \cdot m}{\rho_0} \left(1 - \frac{i^4 \cdot B^4}{k_f^4}\right) \sqrt{B^2 + k_0^2}} i \cdot dB$$

Recombinons maintenant les contributions (1) et (4), puis les contributions (2) et (3) :

$$(1) + (4) = \int_0^{\infty} \frac{-2 \frac{m}{\rho_0} \left(1 - \frac{B^4}{k_f^4}\right) \sqrt{B^2 + k_0^2} e^{-x \cdot B}}{1 + \frac{m^2}{\rho_0^2} \left(1 - \frac{B^4}{k_f^4}\right)^2 (B^2 + k_0^2)} dB$$

$$(2) + (3) = \int_0^{k_0} \frac{-2i \frac{m}{\rho_0} \left(1 - \frac{A^4}{k_f^4}\right) \sqrt{k_0^2 - A^2} e^{i \cdot x \cdot A}}{1 + \frac{m^2}{\rho_0^2} \left(1 - \frac{A^4}{k_f^4}\right)^2 (k_0^2 - A^2)} dA$$

Il reste à effectuer le changement de variables $u = B/k_f$ (respectivement : $u = A/k_f$) et à prendre en compte les contributions des résidus aux pôles inclus dans le contour. On opère de la même manière pour l'équation (EQ I-13). Tous calculs faits, on retrouve les résultats donnés par Feit et Liu [Feit et Liu 85]. Le déplacement transverse et la pression rayonnée en champ proche dans le fluide peuvent s'écrire de la manière suivante (X représentant ici la distance normalisée par rapport au nombre d'onde k_f , c'est-à-dire $X = x \cdot k_f$) :

$$w_{vib}(X, 0) = \frac{i \cdot F \cdot k_f}{4m \cdot \omega^2} \left\{ 4[R_{1w} + R_{2w} + R_{3w}] \right. \quad (\text{EQ I-16})$$

$$- \frac{4\varepsilon}{\pi \cdot \alpha} \cdot i \int_0^\infty \frac{(\alpha^2 + u^2)^{1/2} \exp(-u \cdot X)}{(u^4 - 1)^2 (\alpha^2 + u^2) + \varepsilon^2 / \alpha^2} du$$

$$+ \left. \frac{4\varepsilon}{\pi \cdot \alpha} \int_0^\alpha \frac{(\alpha^2 - u^2)^{1/2} \exp(i \cdot u \cdot X)}{(u^4 - 1)^2 (\alpha^2 - u^2) + \varepsilon^2 / \alpha^2} du \right\}$$

$$P_{vib}(X, 0) = \frac{F \cdot \rho_0}{4m} \left\{ 4 \cdot [R_{1p} + R_{2p} + R_{3p}] \right. \quad (\text{EQ I-17})$$

$$- \left(\frac{4}{\pi} \right) \int_0^\infty \frac{(u^4 - 1)(\alpha^2 + u^2)^{1/2} \exp(-u \cdot X)}{(u^4 - 1)^2 (\alpha^2 + u^2) + \varepsilon^2 / \alpha^2} du$$

$$- \left. \frac{4 \cdot i}{\pi} \int_0^\alpha \frac{(u^4 - 1)(\alpha^2 - u^2)^{1/2} \exp(i \cdot u \cdot X)}{(u^4 - 1)^2 (\alpha^2 - u^2) + \varepsilon^2 / \alpha^2} du \right\}$$

avec : $\varepsilon = \rho_0 \cdot c_0 / m \cdot \omega_c$ (paramètre de charge), $\alpha = (\omega / \omega_c)^{1/2}$ et $\omega_c = c_0^2 \cdot (m/D)^{1/2}$ (pulsation de coïncidence de la plaque).

Les intégrales figurant dans les équations (EQ I-16) et (EQ I-17) sont évaluées numériquement par une méthode de Gauss-Legendre (voir Annexe I-1). Par ailleurs, les résidus s'écrivent :

$$R_{1w} = \exp\left(i \cdot \overline{k_{x1}} \cdot X\right) \left[4 \overline{k_{x1}}^3 + (\varepsilon/\alpha) \overline{k_{x1}} (\overline{k_{x1}}^2 - \alpha^2)^{-3/2} \right]^{-1}$$

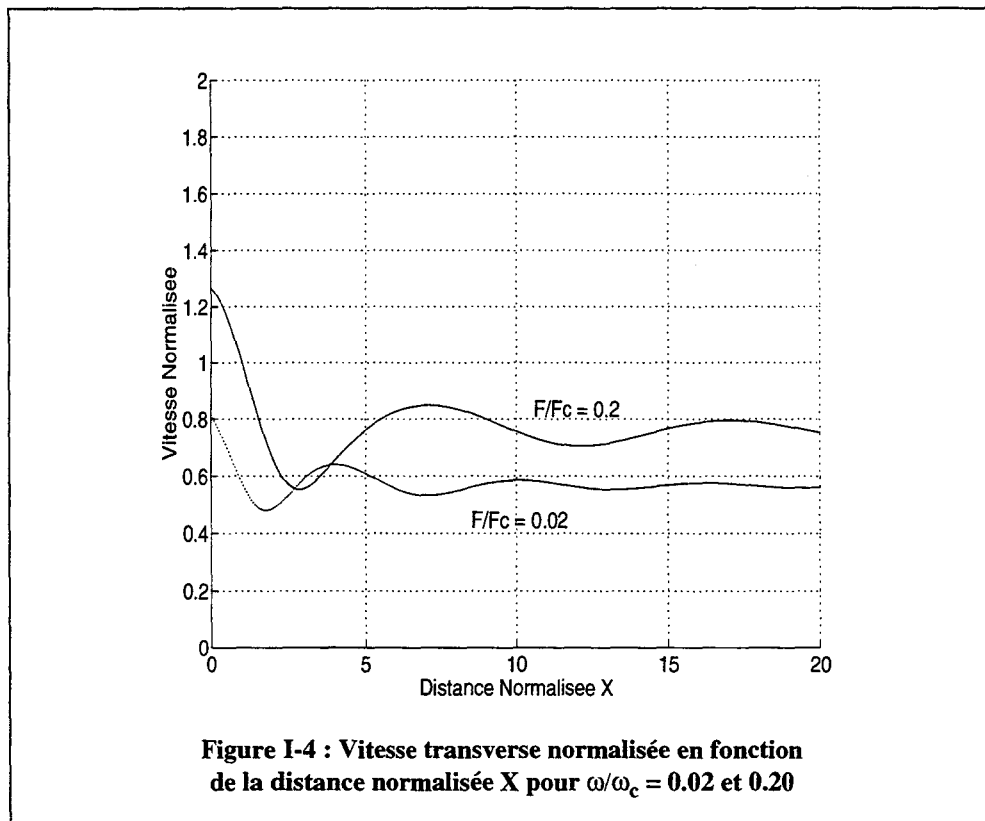
$$R_{2,3w} = \exp\left(i \cdot \overline{k_{x2,3}} \cdot X\right) \left[4 \overline{k_{x2,3}}^3 - (i\varepsilon/\alpha) \overline{k_{x2,3}} (\alpha^2 - \overline{k_{x2,3}}^2)^{-3/2} \right]^{-1}$$

$$R_{1p} = -i \left(\bar{k}_{x1}^2 - \alpha^2 \right)^{-1/2} \cdot R_{1w}$$

$$R_{2,3p} = \left(\alpha^2 - \bar{k}_{x2,3}^2 \right)^{-1/2} \cdot R_{2,3w}$$

où \bar{k}_{x1} , \bar{k}_{x2} , \bar{k}_{x3} sont les racines de l'équation de dispersion, normalisées par rapport à k_f . Pour évaluer ces résidus, il faut connaître les pôles k_{x1} , k_{x2} , k_{x3} et donc résoudre l'équation de dispersion (EQ I-15). Ceci est fait numériquement par une méthode itérative de Newton-Raphson. On remarque lors de la résolution que les parties réelles de k_{x2} et k_{x3} ont une valeur très faible par rapport aux parties imaginaires correspondantes ; la contribution de ces pôles au déplacement transverse et à la pression rayonnée dans le fluide sera donc négligeable dès qu'on se place suffisamment loin de l'excitation.

Avant de généraliser ces résultats pour une excitation large bande en fréquence, nous pouvons les valider en les confrontant avec les résultats obtenus par Feit et Liu [Feit et Liu 85]. Un bon accord est obtenu, notamment sur l'amplitude de la vitesse transverse normalisée (dérivée par rapport au temps du déplacement transverse $w_{vib}(X,0)$, normalisée par rapport à la vitesse transverse dans le vide) en fonction de la distance normalisée X pour différentes fréquences sous la coïncidence : $\omega/\omega_c = 0.02$ et 0.20 (voir Figure I-4).



Pour une fréquence d'excitation donnée inférieure à la fréquence de coïncidence de la plaque, l'amplitude de la vitesse transverse normalisée dans la plaque oscille autour d'une valeur constante. En effet, les résidus associés aux pôles k_{x2} et k_{x3} deviennent très rapidement négli-

geables dès qu'on s'éloigne de l'excitation. Par contre, la contribution associée au pôle réel k_{x1} correspond à une onde de flexion dans la plaque, dont l'amplitude reste indépendante de la distance x . Par ailleurs, on peut montrer par un raisonnement asymptotique [Junger et Feit 86] que l'amplitude des contributions associées aux coupures dans le plan complexe décroît en fonction de la distance x en $(k_f x)^{-3/2}$. Pour conclure, on constate que le fluide a pour effet d'atténuer la vitesse transverse de la plaque pour les fréquences d'excitation inférieures à la coïncidence, et ce d'autant plus que la fréquence d'excitation est faible. Les résultats pour une excitation harmonique étant validés, nous allons maintenant généraliser le raisonnement pour une excitation à spectre large bande en fréquence.

1.3.1.2. EXCITATION RECTILIGNE A SPECTRE LARGE BANDE

Nous allons maintenant considérer une excitation rectiligne à spectre large bande $F(t) \cdot \delta(x)$. Le but est d'obtenir les signaux temporels résultants pour l'accélération transverse et la pression rayonnée dans le fluide en champ proche. Pour cela, nous reprenons la méthode de résolution précédente en appliquant une transformée de Fourier en ω aux équations (EQ I-13) et (EQ I-14). On obtient ainsi les spectres fréquentiels du déplacement transverse de la plaque et de la pression rayonnée en champ proche dans le fluide, évalués en $x = x_0$:

$$\widehat{w}_{vib}(x_0, 0, \omega) = \frac{i \cdot \widehat{F}(\omega)}{2\pi \cdot \omega} \int_C \frac{\exp(i \cdot k_x \cdot x_0)}{\widetilde{Z}_a(k_x) + \widetilde{Z}_p(k_x)} dk_x \quad (\text{EQ I-18})$$

$$\widehat{P}_{vib}(x_0, 0, \omega) = \frac{\widehat{F}(\omega)}{2\pi} \int_C \frac{\widetilde{Z}_a(k_x) \exp(i \cdot k_x \cdot x_0)}{\widetilde{Z}_a(k_x) + \widetilde{Z}_p(k_x)} dk_x \quad (\text{EQ I-19})$$

Par transformée de Fourier inverse en ω , on obtient les signaux dans le domaine temporel (l'accélération transverse est la dérivée seconde par rapport au temps du déplacement transverse) :

$$P_{vib}(x_0, 0, t) = \frac{1}{2\pi} \int_0^{\omega_c} \widehat{P}_{vib}(x_0, 0, \omega) \cdot \exp(i \cdot \omega \cdot t) d\omega \quad (\text{EQ I-20})$$

$$\gamma_{vib}(x_0, 0, t) = \frac{1}{2\pi} \int_0^{\omega_c} -\omega^2 \cdot \widehat{w}_{vib}(x_0, 0, \omega) \cdot \exp(i \cdot \omega \cdot t) d\omega \quad (\text{EQ I-21})$$

L'utilisation de la Transformée de Fourier implique que les signaux soient déterministes. En toute rigueur, ce type d'approche se prête donc mal à la modélisation de signaux aléatoires stationnaires. Seule une description fréquentielle par la densité spectrale de puissance a un sens pour de tels signaux, mais elle ne permet pas de remonter aux signaux temporels. On dépasse cette contradiction en supposant que l'on travaille sur de grandes durées d'observation et que les signaux sont quasi-stationnaires.

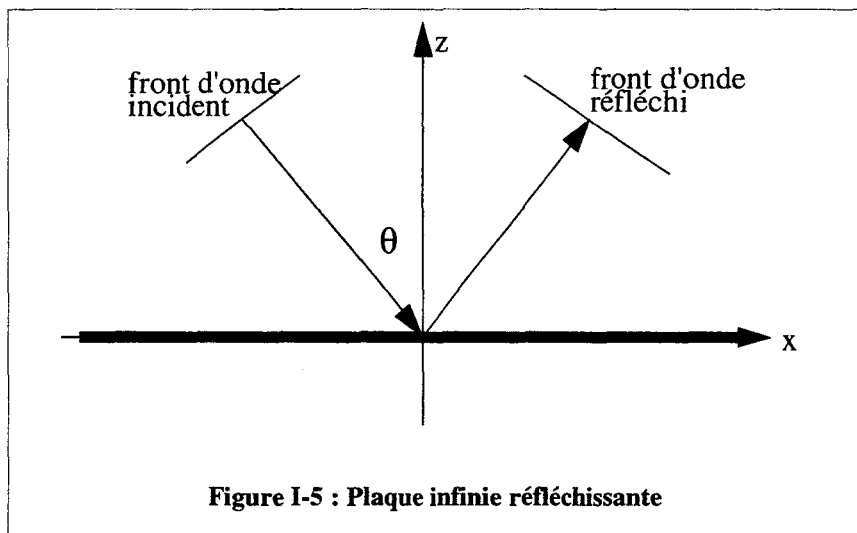
Les équations (EQ I-20) et (EQ I-21) sont résolues par Transformée de Fourier Rapide. Pour chaque fréquence discrétisée, les équations (EQ I-18) et (EQ I-19) sont évaluées par intégration dans le plan complexe (voir paragraphe I.3.1.1). Pour une plaque d'acier d'épaisseur 1 pouce, la fréquence de coïncidence f_c vaut 9030 Hz. Or l'approximation des plaques minces reste

valable pour les fréquences vérifiant $f/f_c < 0.7$. La fréquence maximale de notre bande d'analyse se limite donc à 6 KHz, ce qui correspond aux spécifications imposées pour la soustraction de bruit.

Pour obtenir le spectre en fréquence de l'excitation $F(t)$, nous simulons un processus temporel aléatoire gaussien à moyenne nulle et variance unitaire, dont l'amplitude correspond au niveau spectral isotrope moyen du bruit d'agitation de mer 3 en eau profonde [Thomson ASM 90].

I.3.2. REPONSE DE LA PLAQUE AU SIGNAL ACOUSTIQUE

On prend maintenant en compte sur la plaque un signal acoustique à spectre large bande dans un plan y constant, comme indiqué sur la Figure I-5.



Comme précédemment, nous allons nous inspirer de la méthode utilisée pour un signal harmonique, puis la généraliser pour un signal à spectre large bande en fréquence.

I.3.2.1. SIGNAL ACOUSTIQUE HARMONIQUE

Le problème de la réflexion du son par une plaque élastique infinie a été abordé par Junger et Feit [Junger et Feit 86] pour une onde incidente harmonique de la forme :

$$P_{inc}(x, z, t) = P_i \cdot \exp(i \cdot k_0 \cdot (x \cdot \sin\theta - z \cdot \cos\theta)) \cdot \exp(-i \cdot \omega \cdot t)$$

Junger et Feit ont montré qu'on peut décomposer la pression acoustique totale résultante dans le fluide en trois composantes principales :

- la pression incidente, notée $P_{inc}(x, z, t)$;
- la pression réfléchie en supposant la plaque rigide, qui peut s'écrire :

$$P_{refl}(x, z, t) = P_i \cdot \exp(i \cdot k_0 \cdot (x \cdot \sin\theta + z \cdot \cos\theta)) \cdot \exp(-i \cdot \omega \cdot t)$$

- la pression rayonnée par la vibration de la plaque élastique, notée $P_{ray}(x, z, t)$.

On peut écrire la pression d'excitation à la surface de la plaque comme la somme de la pression incidente et de la pression réfléchie évaluées en $z = 0$:

$$P_{inc}(x, z = 0) + P_{refl}(x, z = 0) = 2 \cdot P_i \cdot \exp(i \cdot k_0 \cdot x \cdot \sin\theta) \quad (\text{EQ I-22})$$

où le facteur de dépendance temporelle est supprimé par souci de concision. On constate que l'amplitude de la pression résultante sur la plaque est double de l'amplitude de la pression incidente. Tout comme l'excitation, la réponse dynamique de la plaque doit être indépendante de y . On aura alors des ondes se propageant dans la plaque suivant l'axe x avec le même nombre d'onde que la pression d'excitation à la surface de la plaque, c'est-à-dire : $k_0 \cdot \sin\theta$. L'accélération transverse résultante à la surface de la plaque peut alors s'écrire :

$$\gamma_{ac}(x) = \Gamma_{ac} \cdot \exp(i \cdot k_0 \cdot x \cdot \sin\theta)$$

Contrairement au cas précédent, l'accélération n'a pas un spectre continu en nombre d'onde, mais elle est constituée d'un Dirac en $k_0 \cdot \sin\theta$. En ce sens, on peut dire que la distribution utilisée pour l'accélération est sinusoïdale. La vitesse de phase des ondes dans la plaque vaut $c_0 / \sin\theta$; par conséquent, elle est toujours supérieure à la vitesse du son dans le fluide.

Pour résoudre l'équation des plaques, il faut connaître la pression résultante totale en $z=0$. On en connaît maintenant une composante, qui intervient comme un terme d'excitation forcée, indépendant de la réponse de la plaque ; mais il faut également prendre en compte la pression rayonnée en champ proche par la vibration de la plaque élastique engendrée par l'onde incidente. Ce terme de pression rayonnée est dû en fait à la distribution sinusoïdale d'accélération $\gamma_{ac}(x, z = 0)$; on peut montrer [Junger et Feit 86] qu'il peut s'écrire de la manière suivante :

$$P_{ray}(x, z) = P \cdot R(z) \cdot \exp(i \cdot k_0 \cdot x \cdot \sin\theta) \quad (\text{EQ I-23})$$

où $R(z)$ exprime la dépendance en z de la pression rayonnée. Cette pression doit satisfaire à l'équation bidimensionnelle de Helmholtz :

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k_0^2 \right) P_{ray}(x, z) = 0 \quad (\text{EQ I-24})$$

Par ailleurs, la condition aux limites sur la plaque s'écrit :

$$\left(\frac{\partial P_{ray}}{\partial z} \right)_{z=0} = -\rho_0 \cdot \gamma_{ac} \quad (\text{EQ I-25})$$

D'où le champ de pression rayonnée :

$$P_{ray}(x, z) = \frac{i \cdot \rho_0}{k_0 \cdot \cos\theta} \Gamma_{ac} \exp(i \cdot k_0 \cdot (x \cdot \sin\theta + z \cdot \cos\theta)) \quad (\text{EQ I-26})$$

On est en présence d'ondes acoustiques se propageant vers les z positifs : en effet, le nombre d'onde dans le fluide est toujours supérieur au nombre d'onde dans la plaque. En évaluant cette expression en $z = 0$, on obtient la troisième composante de pression agissant sur la plaque. D'où l'équation des plaques à l'emplacement des capteurs :

$$\frac{D}{\omega} \left((k_0 \cdot \sin \theta)^4 - k_f^4 \right) \Gamma_{ac} \exp(i \cdot k_0 \cdot x_0 \cdot \sin \theta) = \left[2P_i + \frac{i \cdot \rho_0}{k_0 \cdot \cos \theta} \Gamma_{ac} \right] \exp(i \cdot k_0 \cdot x_0 \cdot \sin \theta) \quad (\text{EQ I-27})$$

On en déduit la réponse totale en accélération, pour une onde incidente harmonique :

$$\gamma_{ac}(x_0, 0) = \frac{-2P_i \cdot \cos \theta \cdot \exp(i \cdot k_0 \cdot x_0 \cdot \sin \theta)}{m \left\{ \frac{i \cdot \rho_0}{m \cdot k_0} + \left[1 - \left(\frac{\omega}{\omega_c} \right)^2 (\sin \theta)^4 \right] \cos \theta \right\}} \quad (\text{EQ I-28})$$

I.3.2.2. SIGNAL ACOUSTIQUE A SPECTRE LARGE BANDE

Pour traiter le cas du signal acoustique à spectre large bande en fréquence, il suffit d'appliquer une transformée de Fourier en ω à l'équation (EQ I-28), et on obtient directement le spectre en fréquence de l'accélération transverse résultante :

$$\widehat{\gamma}_{ac}(x_0, 0, \omega) = \frac{-2\widehat{P}_i(\omega) \cdot \cos \theta \cdot \exp(i \cdot k_0 \cdot x_0 \cdot \sin \theta)}{m \left\{ \frac{i \cdot \rho_0}{m \cdot k_0} + \left[1 - \left(\frac{\omega}{\omega_c} \right)^2 (\sin \theta)^4 \right] \cos \theta \right\}} \quad (\text{EQ I-29})$$

De même pour le spectre de la pression acoustique totale à la surface de la plaque :

$$\widehat{P}_{ac}(x_0, 0, \omega) = 2\widehat{P}_i(\omega) \cdot \exp(i \cdot k_0 \cdot x_0 \cdot \sin \theta) + \frac{i \cdot \rho_0}{k_0 \cos \theta} \cdot \widehat{\gamma}_{ac}(x_0, 0, \omega) \quad (\text{EQ I-30})$$

Les signaux temporels sont alors obtenus par transformée de Fourier inverse. En pratique, on utilise une TFR dans les mêmes conditions que pour l'excitation rectiligne. Nous nous sommes placés dans un cas de figure où le signal incident est du même ordre de grandeur que l'excitation sur la plaque. Nous avons donc généré son spectre fréquentiel de la même manière, en prenant garde toutefois de décorrélérer complètement les deux processus.

I.4. RESULTATS DE LA MODELISATION

Pour valider ce modèle, nous allons étudier l'influence des paramètres suivants sur les différentes densités spectrales de puissance :

- la distance entre l'excitation (ou bruiteur) et les capteurs ;
- le niveau spectral du bruiteur.

Nous allons également ajouter dans le spectre du bruiteur des raies arbitraires, stables en amplitude et en fréquence, afin de prendre en compte les phénomènes de résonances, qui n'ont pas pu être modélisés avec une plaque infinie.

I.4.1. GENERATION DES STIMULI

Dans un premier temps, nous avons fixé les niveaux spectraux de la force d'excitation et du signal acoustique à -60 dB (référence 1Pa), ce qui correspond au niveau spectral isotrope moyen du bruit d'agitation de mer 3 en eau profonde [Thomson ASM 90]. Dans ces conditions, les caractéristiques des processus aléatoires utilisés pour générer les spectres du bruiteur et du

signal acoustique incident sont les suivantes (voir Figures I-6 et I-7).

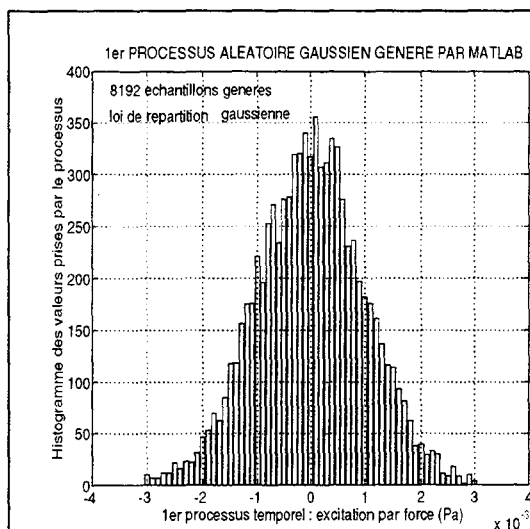


Figure I-6 : Histogramme des valeurs du processus temporel générateur du spectre du bruiteur (unité = $N.m^{-1}$)

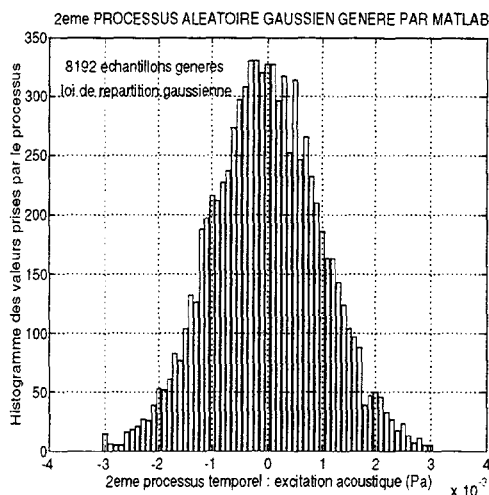


Figure I-7 : Histogramme des valeurs du processus temporel générateur du spectre du signal acoustique incident (unité = Pa)

Le premier processus concerne le bruiteur et le second le signal acoustique incident ; les histogrammes des valeurs (voir Figures I-6 et I-7) montrent qu'il s'agit de processus aléatoires gaussiens, centrés. Les densités spectrales de puissance de ces processus (voir Figures I-8 et I-9), obtenues par périodogramme moyenné de Welch [Oppenheim et Schaffer 75], garantissent que l'excitation sur la plaque et le signal acoustique incident sont des signaux large bande.

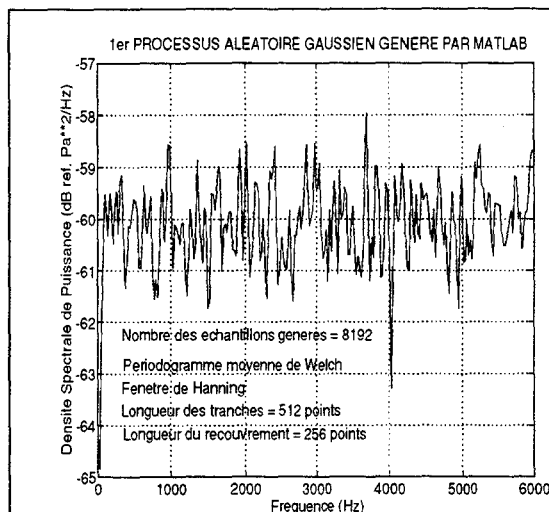


Figure I-8 : Densité spectrale de puissance du processus temporel générateur du spectre du bruiteur

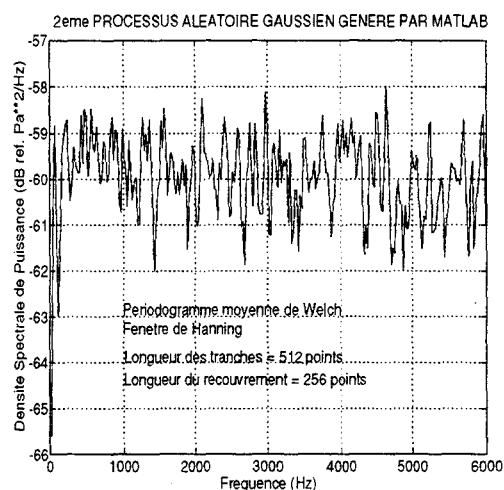
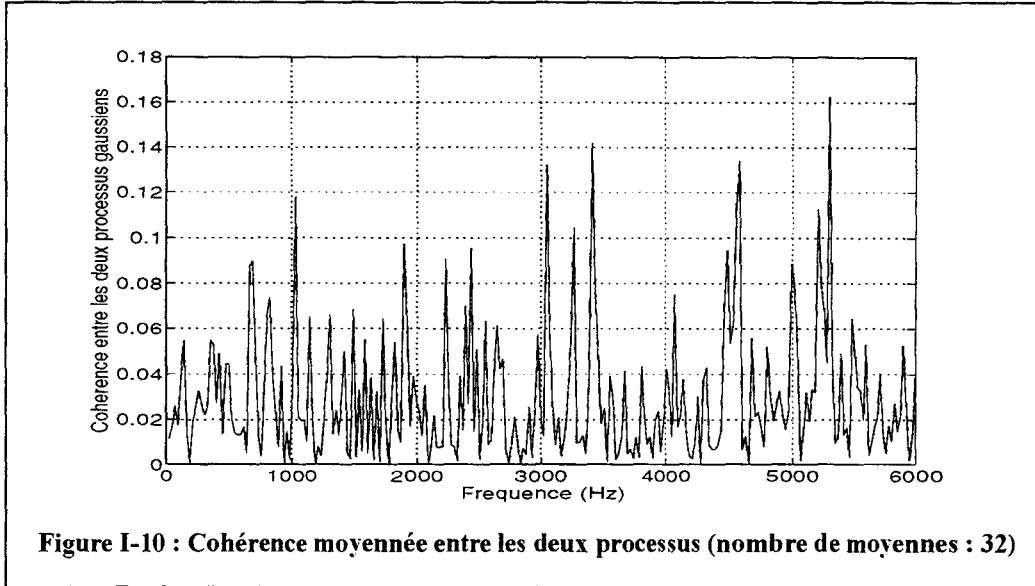


Figure I-9 : Densité spectrale de puissance du processus temporel générateur du spectre du signal acoustique incident

On peut considérer qu'il s'agit de bruits "blancs" dans la bande de fréquences utile [0-6KHz]. Enfin, ces processus sont parfaitement décorrélés : la cohérence moyennée (voir Figure I-10) reste très faible sur la bande de fréquences étudiée. Ceci traduit l'indépendance du signal acous-

tique incident par rapport à la force d'excitation appliquée sur la plaque.

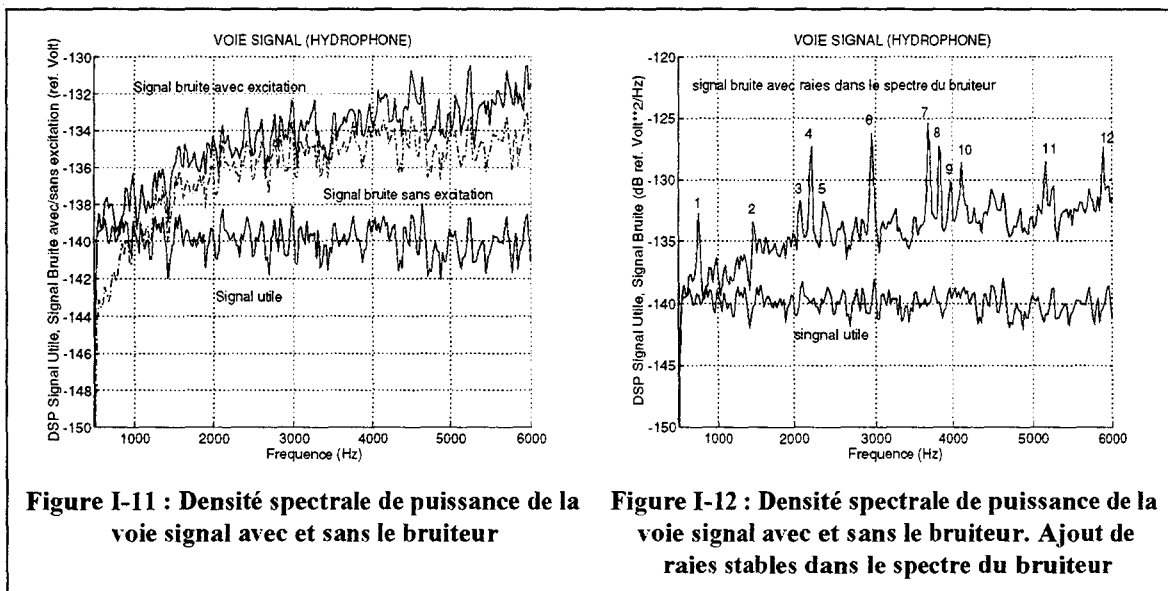


I.4.2. INFLUENCE DE LA DISTANCE BRUITEUR-CAPTEURS

Nous avons fixé les paramètres de la simulation aux valeurs suivantes :

- distance entre le bruiteur et les capteurs (x_0) = 1 m. ;
- angle d'incidence du signal acoustique par rapport à la normale à la plaque = $\pi/4$;
- niveau spectral du bruiteur = -60 dB (référence 1Pa) ;
- niveau spectral du signal acoustique incident = -60 dB (référence 1Pa).

Nous allons en même temps étudier l'influence de la présence de raies stables en amplitude et en fréquence dans le spectre du bruiteur sur les densités spectrales de puissance calculées par le modèle.



Nous considérons que le signal utile (voir Figure I-11) est constitué uniquement de la

pression acoustique incidente, ce qui sous-estime le rapport signal à bruit sur la voie signal ; en effet, on pourrait considérer que la pression réfléchiée par la plaque contribue elle aussi au signal utile. Sous cette hypothèse pessimiste, le rapport signal à bruit sur la voie signal vaut -3.6 dB. Par ailleurs, on mesure approximativement un écart de 6 dB entre la densité spectrale de puissance du signal utile mesurée par l'hydrophone et la densité spectrale de puissance du signal qu'il mesurerait sans le bruiteur, c'est-à-dire si la plaque était soumise uniquement au signal acoustique incident. Ceci peut se justifier par le fait que l'amplitude de la pression résultante à la surface de la plaque est double de l'amplitude de la pression incidente (voir l'équation EQ I-30).

Le niveau du signal utile mesuré par l'hydrophone se trouve à -140 dB (référence 1 Volt) ; en effet, le niveau spectral du signal acoustique incident a été ajusté à -60 dB (référence 1 Pa.) et on prend en compte un hydrophone de sensibilité 10^{-4} V.Pa⁻¹. On note que l'hydrophone réagit assez faiblement au bruiteur (voir Figure I-11). D'une part, le niveau spectral de la pression rayonnée en champ proche dans le fluide à cause de l'excitation acoustique (voir Figure I-19) est supérieur de 5 dB au niveau spectral de la pression rayonnée en champ proche dans le fluide à cause du bruiteur. D'autre part, bien que le niveau spectral de l'accélération transverse générée par le bruiteur soit supérieur de 7 dB au niveau spectral de l'accélération transverse générée par le signal acoustique (voir Figure I-17), nous savons qu'un hydrophone est nettement plus sensible aux effets acoustiques qu'aux effets accélérométriques directs [Audoly et Giangreco 92].

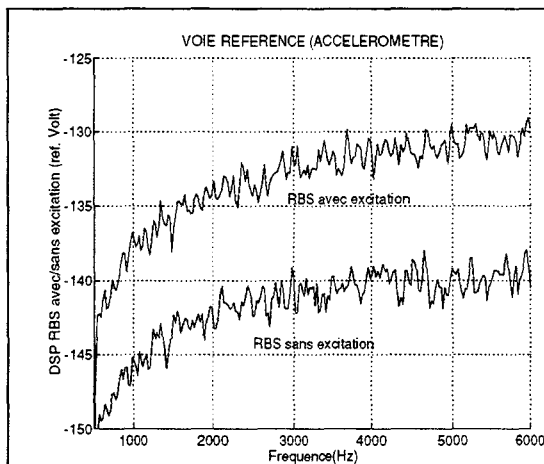


Figure I-13 : Densité spectrale de puissance de la voie référence avec et sans le bruiteur

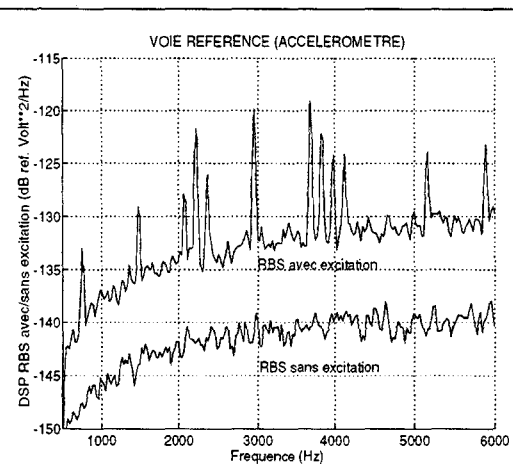


Figure I-14 : Densité spectrale de puissance de la voie référence avec et sans le bruiteur. Ajout de raies stables dans le spectre du bruiteur

L'accéléromètre (voir Figure I-13) mesure un niveau spectral moyen voisin de -140 dB (référence 1 Volt) lorsque la plaque est soumise uniquement au signal acoustique incident. En effet, le niveau spectral de l'accélération transverse générée dans la plaque par le signal acoustique se trouve approximativement à -100 dB (voir Figure I-17), et on prend en compte un accéléromètre de sensibilité 10^{-2} V/m.s⁻². Lorsqu'on prend en compte le bruiteur (Figure I-13),

l'accéléromètre répond beaucoup mieux que l'hydrophone par effet accélérométrique direct. Enfin, on met en évidence (voir Figures I-15 et I-16) un point fondamental : l'absence de cohérence entre les deux capteurs lorsqu'on n'ajoute pas de raies dans le spectre du bruiteur. Ces raies permettent en fait de se rapprocher de la réalité physique, puisque le spectre résultant du bruit de machinerie dans un bâtiment se compose d'un spectre continu de niveau faible, auquel se superposent des raies de fort niveau, essentiellement en basses fréquences.

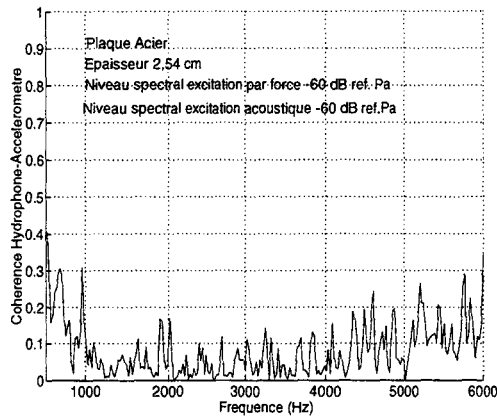


Figure I-15 : Cohérence entre voie signal et voie référence en présence du bruiteur

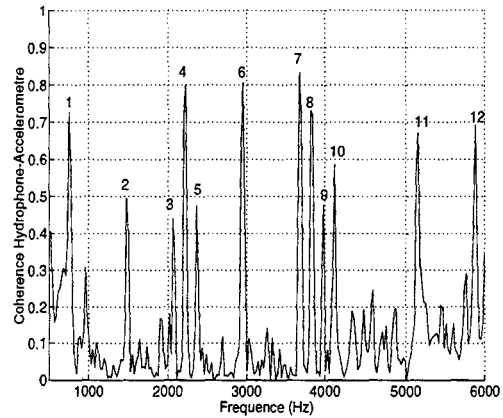


Figure I-16 : Cohérence entre voie signal et voie référence en présence du bruiteur ; Ajout de raies stables dans le spectre du bruiteur

Bien entendu, on retrouve ces raies dans les densités spectrales de puissance de la voie signal (voir Figure I-12), de la référence bruit (voir Figure I-14), de la pression rayonnée dans le fluide P_{vib} (voir Figure I-20) et de l'accélération transverse γ_{vib} (voir Figure I-18).

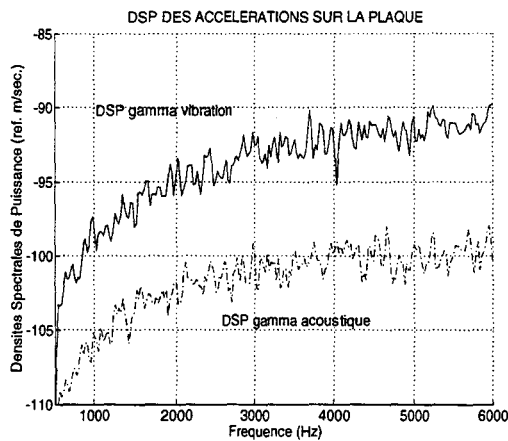


Figure I-17 : Densités spectrales de puissance des accélérations transverse sur la plaque en présence du bruiteur

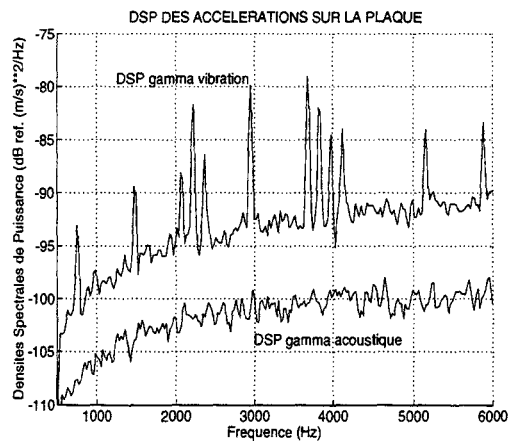
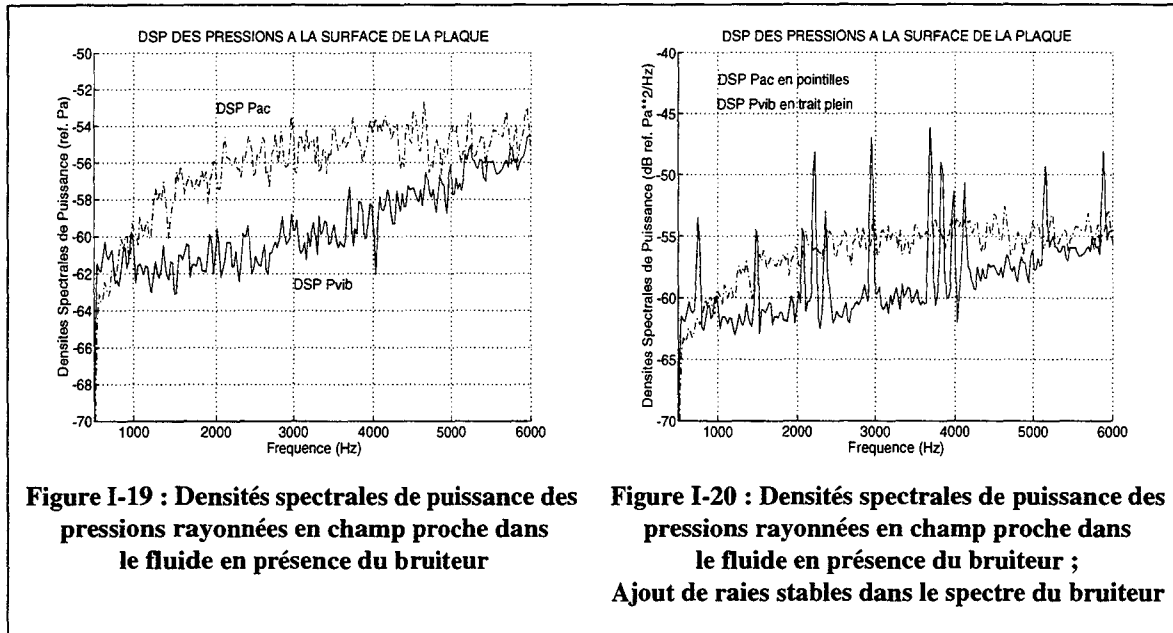


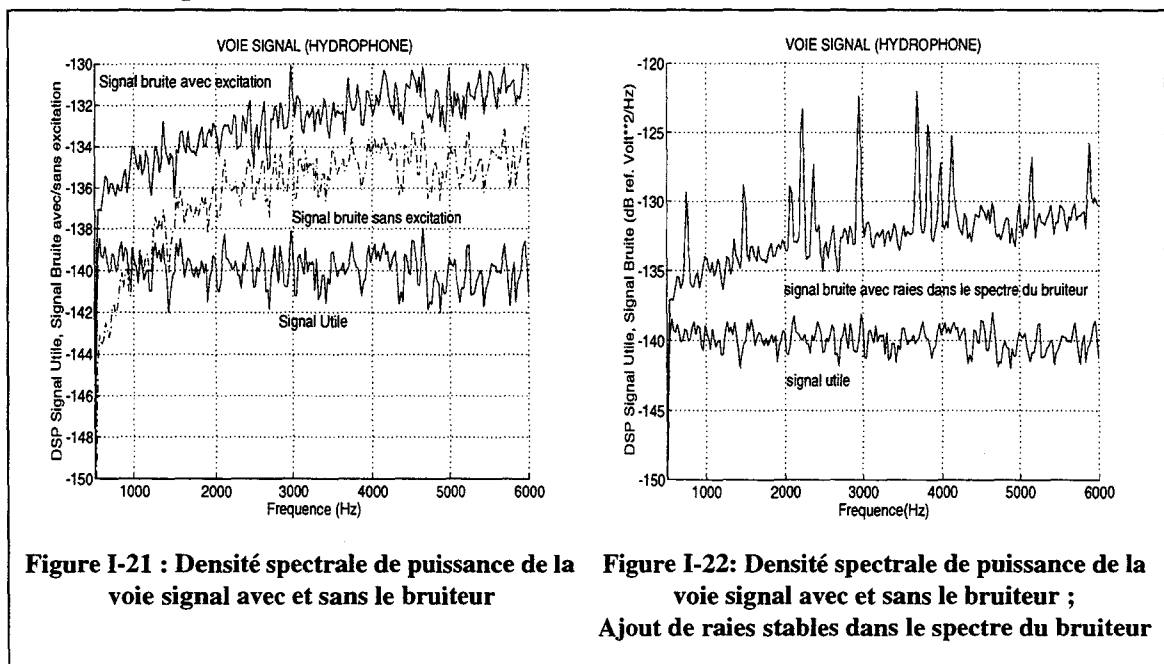
Figure I-18 : Densités spectrales de puissance des accélérations transverse sur la plaque en présence du bruiteur ; Ajout de raies stables dans le spectre du bruiteur

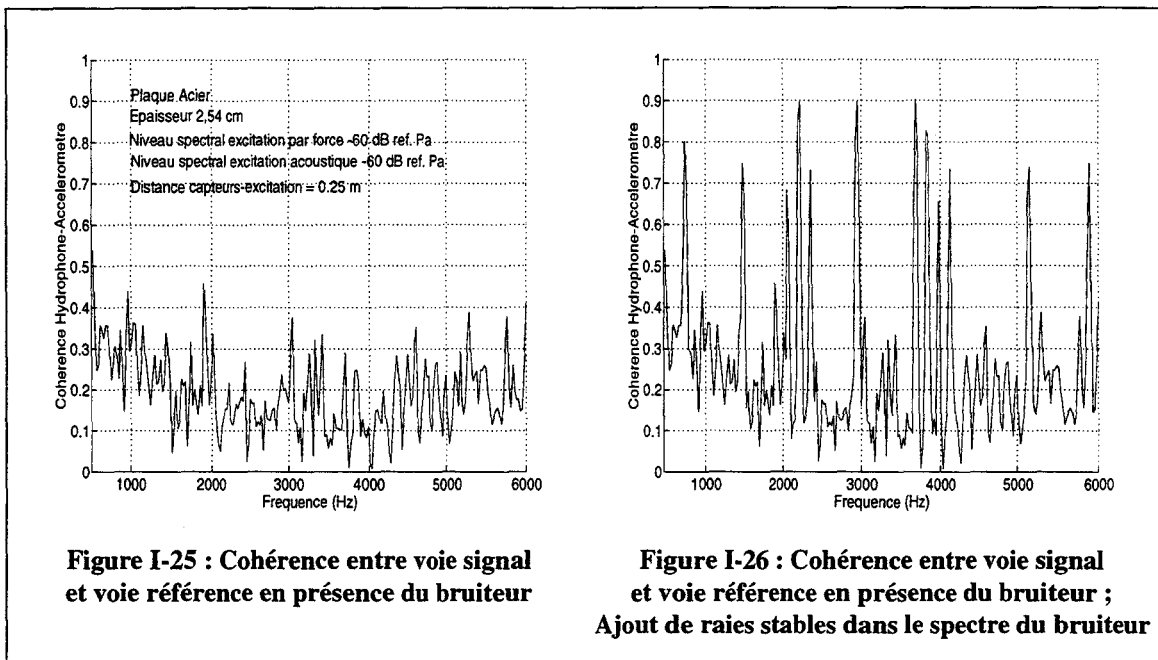
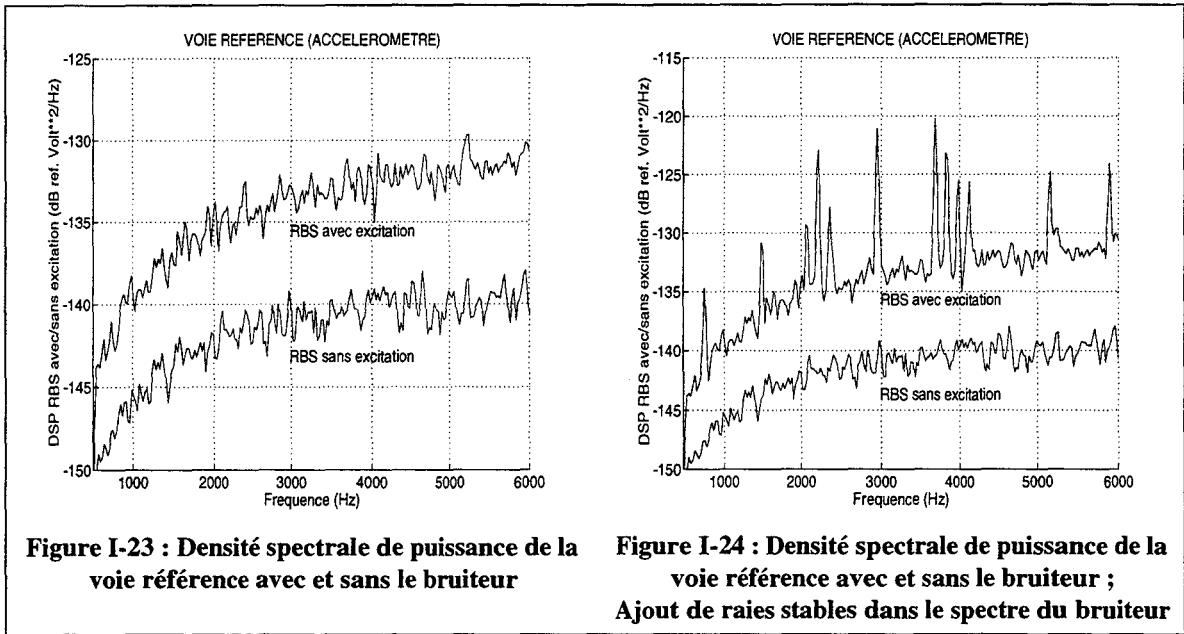
Le résultat fondamental de cette première simulation est d'avoir montré que la cohérence entre les deux capteurs hydrophone et accéléromètre ne peut devenir significative qu'au voisinage de certaines raies (voir Figure I-16). La cohérence reste faible entre les deux capteurs

sur des signaux qui ne présentent pas une structure de raies fines, ce qui confirme les acquisitions expérimentales réalisées récemment par la Direction des Constructions Navales.



Dans une deuxième simulation, nous avons ramené la distance entre le bruitier et les capteurs à 0.25m., afin d'étudier l'influence de cette distance sur les différentes densités spectrales de puissance. On constate que le fait de se rapprocher du bruitier ne modifie pas le niveau du signal utile mesuré par l'hydrophone, puisque le signal acoustique est uniformément réparti sur la plaque. Par contre, le bruit perturbateur est plus présent sur la voie signal (Figure I-21), ce qui se traduit par une dégradation du rapport signal à bruit, qui passe à -5.8 dB. Le niveau spectral de la référence bruit n'est pas modifié (voir Figure I-23). Enfin, la cohérence entre les deux capteurs (voir Figure I-25) est plus élevée que dans le cas précédent, mais reste faible sur la bande de fréquences utile.





L'absence de cohérence entre l'hydrophone et l'accéléromètre sur des signaux ne présentant pas une structure de raies fines (voir Figure I-25) se confirme ; le fait d'ajouter des raies stables dans le spectre du bruiteur renforce de nouveau nettement cette cohérence, ce qui rend envisageable un traitement de soustraction de bruit efficace sur ces raies (voir Figure I-26).

Lorsqu'on s'éloigne significativement du bruiteur, les densités spectrales mesurées par les deux capteurs varient très peu. On retrouve une cohérence très faible entre l'hydrophone et l'accéléromètre sur des signaux ne présentant pas une structure de raies fines, ainsi que la présence de pics de cohérence sur les raies ajoutées dans le spectre du bruiteur. Par ailleurs, les densités spectrales des différentes composantes des signaux mesurés par l'hydrophone et

l'accéléromètre sont semblables à celles calculées pour une distance bruiteur-captteurs de 1m. Il n'est pas étonnant de voir le rapport signal à bruit se stabiliser en fonction de la distance à l'excitation ; en effet, l'étude de la plaque infinie vibrant en flexion montre l'existence d'une onde se propageant sans atténuation pour les fréquences d'excitation inférieures à la coïncidence. La pression acoustique incidente étant répartie uniformément sur la plaque, le rapport signal à bruit doit tendre vers une valeur asymptotique lorsqu'on s'éloigne du bruiteur.

I.4.3. INFLUENCE DU NIVEAU SPECTRAL DU BRUITEUR

On peut se placer dans un cas où la réception est moins bonne sur la voie signal. Pour cela, on fixe le niveau spectral du bruiteur à -54 dB (référence 1Pa.), ce qui correspond au niveau spectral isotrope moyen du bruit d'agitation de mer 5 en eau profonde [Thomson ASM 90]. Dans cette simulation, tous les autres paramètres sont inchangés. Les processus temporels utilisés pour générer les spectres en fréquence du bruiteur et du signal acoustique incident sont toujours gaussiens, centrés et décorrélés.

On constate (voir Figures I-27 et I-28) que l'hydrophone et l'accéléromètre sont sensibles à cette augmentation du bruit perturbateur : le rapport signal à bruit sur la voie signal passe à -7.8 dB. Par ailleurs, on observe une remontée de la cohérence (voir Figure I-29), qui toutefois reste inférieure à 0.5 sur la majeure partie de la bande de fréquences étudiée. Enfin, l'augmentation du niveau du bruit perturbateur se traduit directement par une augmentation des densités spectrales de puissance des termes P_{vib} et γ_{vib} . (voir Figures I-30 et I-31).

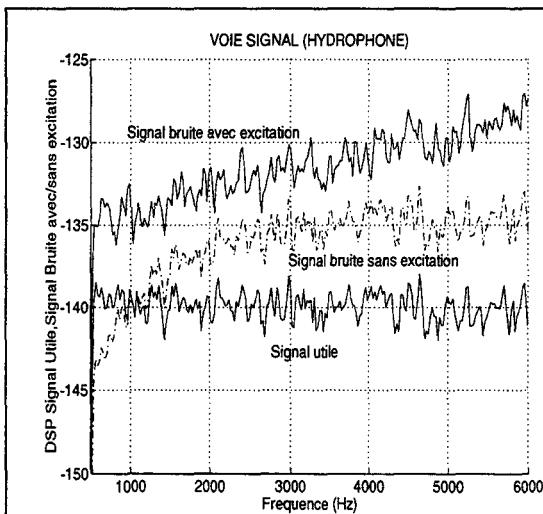


Figure I-27 : Densité spectrale de puissance de la voie signal avec et sans bruiteur

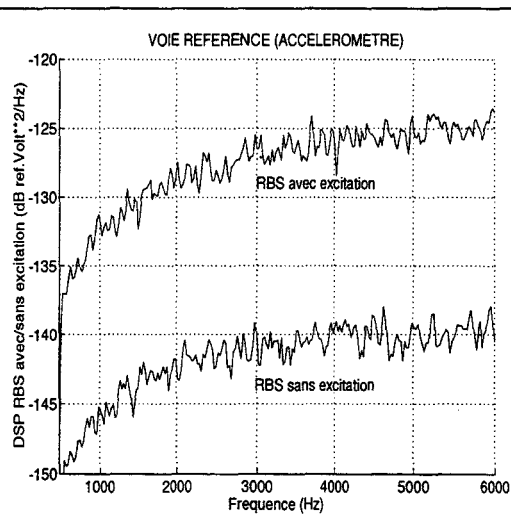
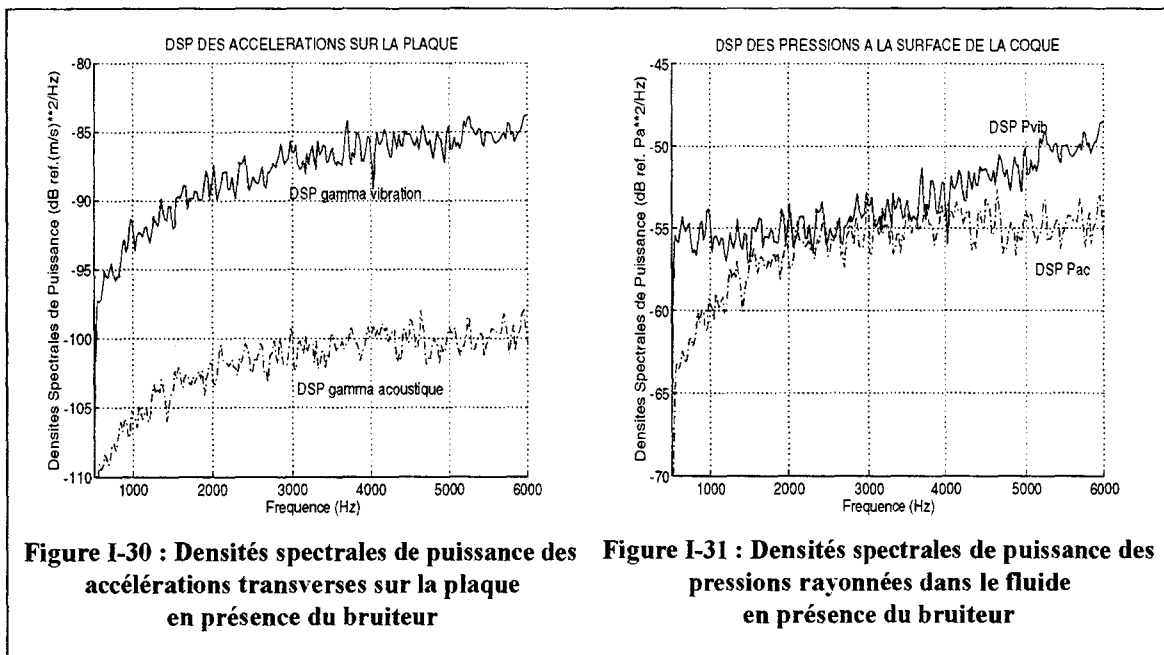
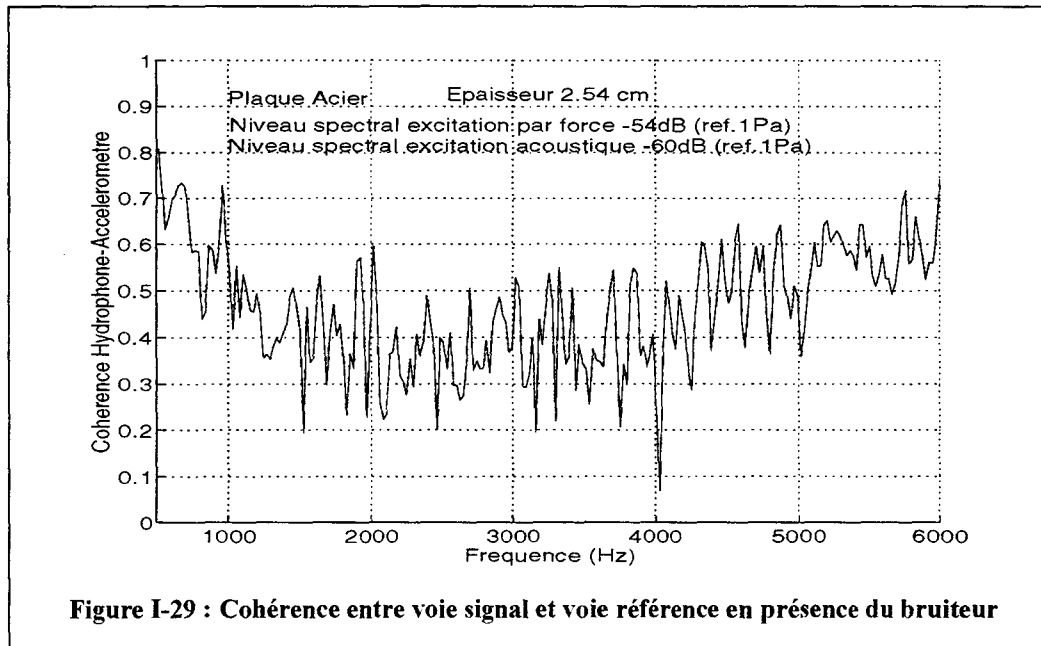


Figure I-28 : Densité spectrale de puissance de la voie référence avec et sans bruiteur



I.5. CONCLUSION

La mise au point de ce modèle nous a permis de simuler les signaux temporels mesurés par un hydrophone et un accéléromètre en vis-à-vis sur une structure vibrante plane de dimensions infinies. Nous avons montré que la cohérence entre ces deux voies est faible sauf dans le cas où des raies sont rajoutées au bruit pour simuler les effets des harmoniques des machines tournantes et des résonances d'une plaque de dimensions finies. On peut donc à priori envisager un traitement de soustraction de bruit efficace sur ces raies.

I.6. BIBLIOGRAPHIE

- [Audoly et Giangreco 92] AUDOLY C., GIANGRECO C., A study of the output of a hydrophone fixed on a vibrating structure, Journal de Physique IV, Colloque C1, supplément au Journal de Physique III, Vol.2, Avril 1992.
- [Feit et Liu 85] FEIT D., LIU Y.N., The nearfield response of a line-driven fluid-loaded plate, J. Acoustical Society of America, August 1985.
- [Junger et Feit 86] JUNGER M.C., FEIT D., Sound, Structures and their Interaction, 2nd Edition, 1986.
- [Oppenheim et Schaffer 75] OPPENHEIM A.V., SCHAFER R.W., Digital Signal Processing, p.556, Prentice-Hall, 1975.
- [Thomson ASM 90] Thomson Sintra ASM, Sonar et Acoustique Sous-Marine, Tome I (Le Milieu Marin), 1ère édition, Décembre 1990.

Annexe I-1 : Intégration numérique par la méthode de Gauss-Legendre

Cette méthode, très précise, utilise des points non régulièrement espacés et convenablement choisis. Elle peut donc s'appliquer lorsque l'intégrande est connu analytiquement, ou lorsqu'il est tabulé numériquement en ces points précis. Elle ne peut pas s'appliquer lorsque l'intégrande est donné par les valeurs qu'il prend en un certain nombre de points différents de ces points privilégiés. Nous ne donnerons pas les détails du calcul, mais seulement l'idée directrice et la manière d'appliquer la méthode.

D'une manière générale, en développant un intégrande $f(x)$ sur une base de polynômes, on peut écrire l'intégrale comme une combinaison linéaire des valeurs que prend la fonction en différents points :

$$\int_a^b f(x) dx = C \cdot [w_1 \cdot f(x_1) + w_2 \cdot f(x_2) + \dots + w_n \cdot f(x_n)]$$

où C est proportionnel à $(b-a)$, et où les facteurs de pondération w_i dépendent de la fonction par laquelle on approche $f(x)$ (segments de droites pour la méthode des trapèzes, arcs de paraboles pour la méthode de Simpson). Dans la méthode de Gauss, on développe $f(x)$ dans une base de polynômes orthogonaux. Les x_i sont alors racines de ces polynômes, et sont donc irrégulièrement espacés. On peut utiliser des polynômes de Jacobi, Tchebychev, Laguerre, Hermite..., mais les plus couramment utilisés sont ceux de Legendre. Ceux-ci étant définis dans l'intervalle $[-1 ; +1]$, il faut d'abord faire sur x un changement de variable linéaire transformant x de l'intervalle $[a ; b]$ en x^* appartenant à l'intervalle $[-1 ; +1]$. On choisit ensuite la valeur de n , qui donne le nombre de points où la fonction $f(x)$ doit être évaluée. On lit ensuite dans les tables les valeurs de x^* , racines du polynôme de Legendre d'ordre n . Par changement de variable linéaire, on remonte aux valeurs correspondantes de x , et on applique l'équation précédente avec $C = (b-a)/2$.

II

SOUSTRACTION DE BRUIT

II. SOUSTRACTION DE BRUIT

II.1. NOTATIONS ET ABBREVIATIONS UTILISEES

Dans ce chapitre, nous utiliserons les notations suivantes pour les algorithmes temporels :

$x(n)$	signal mesuré à l'instant n par l'hydrophone (voie signal)
$s(n)$	signal utile
$v(n)$	bruit additif sur la voie signal
$u(n)$	signal mesuré à l'instant n par l'accéléromètre (référence bruit seul)
$y(n)$	sortie du filtre adaptatif
$e(n)$	signal estimé à l'instant n (signal débruité)
$H(z)$	fonction de transfert en z du filtre adaptatif
M	ordre du filtre adaptatif RIF
R	matrice d'autocorrélation statistique d'ordre M de la référence bruit (M,M)
p	vecteur d'intercorrélacion statistique d'ordre M entre voie signal et voie référence ($M,1$)
R_n	matrice d'autocorrélation stochastique d'ordre M de la référence bruit (M,M)
p_n	vecteur d'intercorrélacion stochastique d'ordre M entre voie signal et voie référence ($M,1$)
$\underline{\theta}_M$	paramètre vectoriel ($M,1$)
$\underline{\theta}_M^n$	paramètre vectoriel ($M,1$) évalué à l'instant n
$J(\underline{\theta})$	fonction de coût statistique du paramètre vectoriel $\underline{\theta}$
$L^2(\Omega)$	espace de Hilbert des variables aléatoires de carré intégrable
$\underline{U}(n)$	vecteur de régression d'ordre M de la référence bruit seul à l'instant n ($M,1$) $\underline{U}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^T$
U_n	sous-espace de $L^2(\Omega)$ engendré par les M échantillons temporels les plus récents de la référence bruit seul
\underline{w}^n	vecteur des M coefficients du filtre adaptatif RIF calculé à l'instant n dans le domaine temporel ($M,1$) : $\underline{w}^n = [w^n(0), w^n(1), \dots, w^n(M-1)]^T$
$(\underline{X}, \underline{Y})$	produit scalaire entre les deux vecteurs \underline{X} et \underline{Y}
\underline{x}_n	vecteur d'observation de la voie signal dans R^T (T représentant la taille maximale des observations) : $\underline{x}_n = [x(n), x(n-1), \dots, x(1), 0, \dots, 0]^T$ ($T,1$)
$X_{k,j}(n)$	matrice des observations de la voie signal entre les instants k et j ($T, j-k+1$) : $X_{k,j}(n) = [\underline{x}_{n-k}, \underline{x}_{n-k-1}, \dots, \underline{x}_{n-j}]$
$\underline{\sigma}$	vecteur d'annihilation temporelle ($T,1$) : $\underline{\sigma} = [1, 0, \dots, 0]^T$
$P_{1,M}(n)$	projecteur orthogonal (T,T) sur le sous-espace de R^T défini par la matrice des observations $U_{1,M}(n)$
$\overline{P}_{1,M}(n)$	projecteur orthogonal (T,T) sur le sous-espace de R^T orthogonal au sous-espace défini par la matrice des observations $U_{1,M}(n)$

$\underline{f}_{M,n}$	vecteur d'erreur de prédiction avant (T,1) sur le vecteur d'observation \underline{u}_n
$f_{M(n)}$	erreur de prédiction avant instantanée sur le vecteur d'observation \underline{u}_n : c'est la première composante du vecteur $\underline{f}_{M,n}$
$\underline{b}_{M,n}$	vecteur d'erreur de prédiction arrière (T,1) sur le vecteur \underline{u}_{n-M}
$b_{M(n)}$	erreur de prédiction arrière instantanée sur le vecteur \underline{u}_{n-M} : c'est la première composante du vecteur $\underline{b}_{M,n}$
\underline{g}^n	vecteur gain de Kalman (M,1) d'ordre M calculé à l'instant n
\underline{a}^n	filtre prédictif direct (ou avant) d'ordre M (M,1) de la référence bruit calculé à l'instant n
\underline{b}^n	filtre prédictif rétrograde (ou arrière) d'ordre M (M,1) de la référence bruit calculé à l'instant n
$\alpha(n)$	puissance d'erreur de prédiction avant calculée à l'instant n
$\beta(n)$	puissance d'erreur de prédiction arrière calculée à l'instant n
$\gamma(n)$	paramètre angulaire à l'ordre M
$\gamma_{M+1}(n)$	paramètre angulaire à l'ordre M+1
Pour les <u>algorithmes fréquentiels</u> , les notations suivantes seront employées :	
N	nombre d'échantillons par bloc de données
D	nombre de points de recouvrement entre deux blocs consécutifs
$w^j(v)$	coefficient complexe représentant le gain du filtre adaptatif \underline{w} calculé au canal fréquentiel v après acquisition de la tranche de signal de rang j
$X_j(v)$	TFD de la tranche de rang j contenant N échantillons du signal $x(n)$ pour le canal fréquentiel v
$X_j^*(v)$	nombre complexe conjugué de $X_j(v)$
\underline{u}_{jL}	bloc de rang j du signal $u(n)$ (L,1) : $\underline{u}_{jL} = [u(jL), u(jL+1), \dots, u(jL+L-1)]^T$
\underline{u}_{jL}^a	$\underline{u}_{jL}^a = [u(jL-D), u(jL-D+1), \dots, u(jL-1), u(jL), u(jL+1), \dots, u(jL+L-1)]^T$ (L+D,1) : bloc de rang j du signal $u(n)$ dans la méthode par recouvrement de blocs.

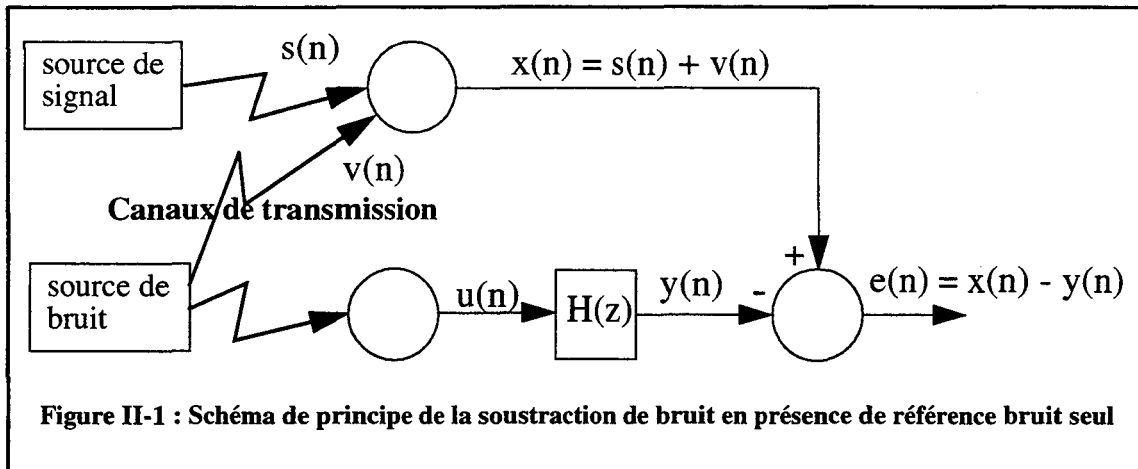
Par ailleurs, nous utiliserons souvent les abréviations suivantes :

RBS	Référence Bruit Seul
EQM	Erreur Quadratique Moyenne
EQMR	Erreur Quadratique Moyenne Résiduelle
RIF	Réponse Impulsionnelle Finie
RII	Réponse Impulsionnelle Infinie
LMS	Least Mean Square
RLS	Recursive Least Squares
LSL	Least Squares Lattice
FTF	Fast Transversal Filter
(I)FFT	(Inverse) Fast Fourier Transform

II.2. POSITION SIMPLIFIEE DU PROBLEME

L'objectif de ce chapitre n'est pas de faire une présentation exhaustive de tout ce qui existe en matière de soustraction de bruit, mais plutôt de rappeler les principes de base d'extraction de l'information contenue dans un signal bruité par des techniques de filtrage adaptatif avec référence bruit seul. Dans la présentation des techniques existantes, nous garderons à l'esprit la nécessité d'une mise en oeuvre finale en temps réel sur un processeur de traitement du signal (DSP) à faible coût. Le coût de calculs des algorithmes sera donc un critère de sélection majeur pour notre application.

Le but est de débruiter le signal mesuré par un hydrophone appartenant à une antenne de sonar passif de sa composante de bruit propre d'origine mécanique. Pour cela, on se munit d'un accéléromètre, qui constitue le capteur de référence bruit seul (voir Figure II-1).



Le signal $x(n)$ mesuré à l'instant n par l'hydrophone est la somme d'un signal utile $s(n)$ et d'un bruit perturbateur $v(n)$ corrélé à la référence bruit seul accélérométrique $u(n)$. Le but de la soustraction de bruit est d'identifier un filtre $H(z)$ qui permette de reconstruire $v(n)$ à partir de $u(n)$ en temps réel. La qualité de la séparation dans $x(n)$ entre la partie utile $s(n)$ et le bruit $v(n)$ dépendra du degré de dépendance linéaire entre le bruit $v(n)$ et la référence bruit seul $u(n)$, ainsi que de la décorrélation entre le signal utile $s(n)$ et le signal mesuré par la référence bruit $u(n)$.

Les caractéristiques des canaux par lesquels le bruit est transmis aux capteurs sont inconnues et peuvent évoluer au cours du temps, d'où la nécessité d'un filtrage adaptatif sur les données. La réponse impulsionnelle du filtre adaptatif $H(z)$ pourra être mise à jour à l'aide d'un algorithme de minimisation d'un critère d'erreur. Dans un système de soustraction de bruit, le signal d'erreur est directement la puissance du signal de sortie $e(n)$. En effet, supposons que les signaux $s(n)$, $u(n)$, $v(n)$ et $y(n)$ soient stationnaires et à moyenne nulle. En utilisant la relation :

$$e(n) = s(n) + v(n) - y(n) \quad (\text{EQ II-1})$$

on peut exprimer la puissance du signal de sortie :

$$E[e(n)^2] = E[s(n)^2] + E[(v(n) - y(n))^2] + 2 \cdot E[s(n) \cdot v(n)] - 2 \cdot E[s(n) \cdot y(n)] \quad (\text{EQ II-2})$$

En utilisant l'hypothèse de décorrélation entre $s(n)$ et $u(n)$, on peut simplifier cette expression :

$$E[e(n)^2] = E[s(n)^2] + E[(v(n) - y(n))^2] \quad (\text{EQ II-3})$$

Si $s(n)$ est décorrélé avec $u(n)$, il est également décorrélé avec $y(n)$; on peut donc considérer que le traitement ne modifie pas le signal utile $s(n)$. Dans ces conditions, minimiser la puissance totale du signal de sortie revient à minimiser la puissance de l'erreur d'estimation sur le bruit perturbateur $v(n)$ [Widrow 75]. On voit que, sous l'hypothèse fondamentale de décorrélation entre signal utile et référence bruit seul, minimiser la puissance du signal estimé équivaut à minimiser la puissance du bruit résiduel.

II.3. DERIVATION DES ALGORITHMES TEMPORELS

Ce paragraphe a pour but de présenter une méthodologie générale de dérivation pour les algorithmes de soustraction de bruit adaptatifs dans le domaine temporel. D'une manière tout à fait générale, on utilise une démarche en deux étapes [Michaut 92] [Haykin 96] :

- choix d'un modèle pour le filtre adaptatif : Réponse Impulsionnelle Finie (RIF) ou Réponse Impulsionnelle Infinie (RII), en fonction du système à identifier ;
- choix d'une méthode de résolution du problème : filtrage de Wiener, méthode des Moindres Carrés ou filtrage de Kalman.

II.3.1. CHOIX DU MODELE POUR LE FILTRE ADAPTATIF

Il faut faire le choix d'un modèle RIF ou RII pour la fonction de transfert $H(z)$. Le choix d'un modèle RII, bien qu'intéressant pour représenter des systèmes à réponse impulsionnelle longue, ne se justifiera que pour atteindre un objectif d'optimisation qui serait inaccessible à un filtre RIF d'ordre raisonnable. En effet, le choix d'un modèle récursif pose de nombreux problèmes de mise en oeuvre par rapport à un modèle transverse classique. En pratique, nous choisirons un modèle RIF d'ordre fini, noté désormais M .

II.3.2. CHOIX D'UNE APPROCHE DE RESOLUTION DU PROBLEME

II.3.2.1. FILTRAGE DE WIENER

On part d'une fonction de coût statistique (c'est-à-dire faisant intervenir l'opérateur d'espérance mathématique) à minimiser par rapport aux coefficients du filtre adaptatif : c'est l'Erreur Quadratique Moyenne (EQM). La solution de la minimisation est alors donnée par l'équation normale statistique, qui fait intervenir l'inverse de la matrice d'autocorrélation statistique d'ordre M (notée R) de la référence bruit [Haykin 96].

A ce niveau, on peut résoudre le problème de deux manières équivalentes mathématiquement, mais qui débouchent sur des représentations différentes :

- **Représentation transversale directe** : le paramètre à optimiser est le vecteur des M coefficients du filtre adaptatif RIF. On décompose l'estimation du bruit perturbateur sur les M échantillons d'observation les plus récents de la référence bruit seul, puis on adapte directement les coefficients de cette combinaison linéaire. Le filtre transversal optimal est calculé itérativement par une méthode newtonienne d'optimisation de la fonction de coût statistique. On a alors le choix entre une méthode de type gradient, c'est-à dire sans gain matriciel, et une méthode de type quasi-Newton, avec un gain matriciel. Dans les deux cas, le passage de l'algorithme statistique (c'est-à dire faisant intervenir l'opérateur d'espérance mathématique) à un algorithme stochastique (calculable directement sur les données) se fait par approximation stochastique directe du vecteur gradient, c'est-dire que l'on supprime directement l'opérateur d'espérance mathématique. La méthode du gradient conduit à la famille des algorithmes LMS-transverse, dont le coût de calculs est en $o(M)$ opérations par itération. Ces algorithmes présentent l'inconvénient d'être sous-optimaux et d'avoir une vitesse de convergence dépendante du conditionnement de la matrice R . La méthode quasi-Newton conduit à l'algorithme RLS, optimal mais très coûteux en calculs : $o(M^2)$ opérations par itération. Ainsi, la représentation transversale directe dans l'approche de Wiener conduit à choisir entre une solution exacte (RLS) au coût de calculs prohibitif et une solution approchée (LMS) à faible coût de calculs.
- **Représentation en treillis** : la base de décomposition est obtenue par orthogonalisation de Gram-Schmidt des échantillons d'observation de la référence bruit seul. La nouvelle base ainsi obtenue est constituée des erreurs de prédiction rétrograde ; dans cette base, la matrice d'autocorrélation est diagonale, ce qui simplifie son inversion. Mathématiquement, ce procédé revient à opérer une factorisation de Cholesky sur la matrice d'autocorrélation R de la référence bruit. Cette approche fait intervenir les notions de prédiction directe et rétrograde. Le problème d'optimisation est plus complexe puisqu'il se décompose en deux étapes ; néanmoins, la méthode quasi-Newton utilisée dans cette représentation conduit à un algorithme moins coûteux que le RLS : $o(M)$ opérations par itération. En effet, la mise en oeuvre d'un gain matriciel dans la représentation en treillis conduit à une simple normalisation des pas de calculs. Ce gain fondamental est lié à la diagonalisation du problème d'optimisation. Des algorithmes sous-optimaux et moins coûteux en calculs, de type LMS-treillis peuvent également être obtenus. Ils sont moins sensibles au conditionnement du problème que les algorithmes de type LMS-transverse, à condition de choisir un pas de calcul adapté par étage, ce qui n'est pas évident sans une connaissance à priori de la statistique des données.

Ces deux approches partent de la matrice d'autocorrélation statistique et utilisent ses propriétés : symétrie sur signaux réels et structure Toeplitz sur signaux stationnaires au second ordre. Elles sont donc adaptées à un contexte stationnaire. Suivant la représentation choisie (transversale directe ou treillis) et la méthode d'optimisation (gradient ou quasi-Newton), on ob-

tient différents algorithmes. Ces algorithmes peuvent être modifiés pour fonctionner en contexte faiblement non-stationnaire par le choix du pas d'adaptation intervenant dans la méthode d'optimisation.

II.3.2.2. METHODE DES MOINDRES CARRES

Cette méthode présente une formulation stochastique dès le départ. On part d'une fonction de coût stochastique qui converge, sous l'hypothèse d'ergodisme des signaux, vers la fonction de coût statistique utilisée dans le filtrage de Wiener. On évite ainsi l'approximation stochastique directe, puisqu'on travaille directement sur les données. On a le choix entre plusieurs fonctions de coût, qui définissent un fenêtrage différent sur les données et débouchent sur des algorithmes différents. D'une manière générale, la solution est donnée par l'équation normale stochastique, qui fait intervenir une matrice R_n stochastique, aux propriétés beaucoup moins intéressantes que la matrice R statistique du filtrage de Wiener (en particulier, on perd la structure Toeplitz).

Comme pour le filtrage de Wiener, on est amené au problème de l'inversion de R_n . A nouveau, on peut résoudre ce problème de plusieurs manières, qui débouchent sur des représentations différentes :

- **Représentation transversale directe** : on peut facilement retrouver les algorithmes de type RLS obtenus dans l'approche de Wiener. Il est également possible de développer une théorie géométrique simple à base de projecteurs orthogonaux permettant d'obtenir des algorithmes plus rapides [Cioffi et Kailath 84]. Cette théorie géométrique fait intervenir l'orthogonalisation de l'espace d'observation et donc les concepts de prédiction directe et rétrograde. Les algorithmes qui en dérivent sont classiquement appelés algorithmes RLS-rapides (Fast-Kalman, FAEST, FTF...) et ont une complexité en $o(M)$ opérations par itération ; en revanche, ils n'ont pas la propriété de stabilité exponentielle pour la propagation des erreurs [Slock et Kailath 88]. En particulier, le choix d'un facteur d'oubli strictement inférieur à 1 donne lieu à des risques de divergence numérique rapide, d'où la nécessité pour ces méthodes de mettre en oeuvre des techniques de stabilisation numérique afin de garantir leur fonctionnement en horizon temporel infini [Benallal et Gilloire 88].
- **Représentation en treillis** : le problème d'inversion matricielle est résolu par factorisation de Cholesky de la matrice R_n stochastique, et repose sur une orthogonalisation de Gram-Schmidt de l'espace d'observation. On débouche sur des algorithmes des moindres carrés en treillis (Least Squares Lattice) ; leur complexité est également en $o(M)$ opérations par itération, et ils présentent une supériorité numérique évidente sur leurs homologues transversaux, avec notamment la propriété de stabilité exponentielle [Friedlander 82].
- **Représentation en tableau systolique** : le problème d'inversion matricielle est résolu par décomposition Q-R. Cette décomposition transforme directement la matrice du signal d'entrée en une matrice qui a toutes ses valeurs nulles, à l'exception d'une sous-matrice

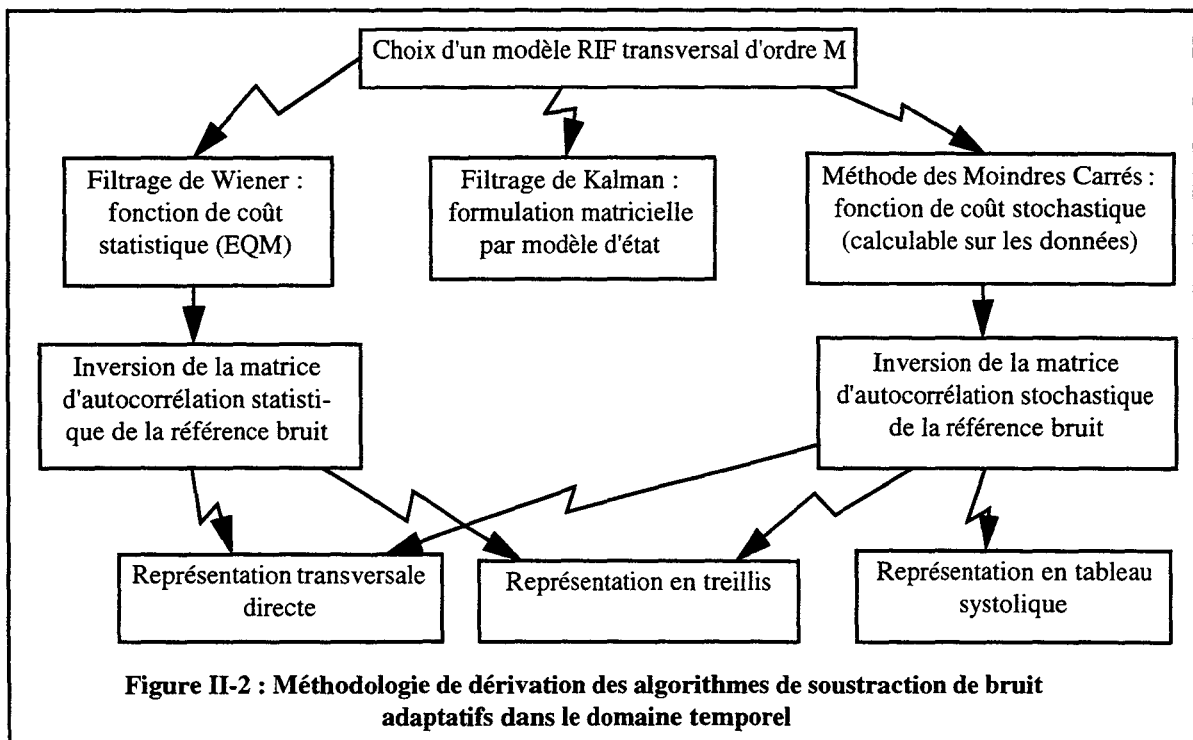
triangulaire. La triangularisation se fait par des opérations de rotation [Givens 58], qui sont très résistantes vis-à-vis des effets de réalisation en précision finie. Le filtre optimal est alors calculé par résolution d'un système triangulaire. Le processus de triangularisation peut être mis en oeuvre très efficacement dans une structure en tableau systolique [Kung et Leiserson 78], mais cette technique donne lieu à des algorithmes beaucoup plus coûteux en calculs que les deux représentations précédentes.

L'intérêt de la méthode des Moindres Carrés est double par rapport à l'approche de Wiener : la fonction de coût n'étant pas unique, on peut l'adapter au problème à résoudre ; de plus, une approche purement géométrique en découle, qui permet de dériver des algorithmes rapides. Enfin, la non-stationnarité des signaux est prise en compte dès la fonction de coût et traitée mathématiquement, ce qui n'est pas le cas dans l'approche de Wiener.

II.3.2.3. FILTRAGE DE KALMAN

Par souci d'exactitude, nous devons mentionner une troisième approche possible de résolution du problème : le filtrage de Kalman. Cette méthode part d'une représentation d'état d'un système bruité ; c'est un compromis entre le filtrage de Wiener et la méthode des Moindres Carrés car le problème initial est formulé de manière statistique, et la non-stationnarité des signaux peut être prise en compte dès le départ. L'inconvénient majeur de cette approche est d'utiliser une formulation matricielle par modèle d'état. Elle est donc particulièrement coûteuse en calculs, et de ce fait mal adaptée à une mise en oeuvre en temps réel.

La figure II-2 résume les différentes approches possibles de résolution du problème.



II.4. DERIVATION DES ALGORITHMES FREQUENTIELS

Les méthodes fréquentielles fournissent une alternative très intéressante aux méthodes temporelles. En effet, elles permettent de résoudre l'approche de Wiener de manière exacte avec un coût de calculs autorisant une mise en oeuvre en temps réel sur DSP, grâce notamment aux algorithmes de Transformée de Fourier Rapide (TFR, ou FFT). La FFT peut être considérée comme une technique d'orthogonalisation des signaux d'entrée, dans la mesure où les différents canaux fréquentiels sont approximativement décorrélés deux à deux [Widrow et Stearns 85] [Servièrre 89]. Sous cette hypothèse, on peut mettre à jour les canaux fréquentiels du gain du filtre adaptatif indépendamment les uns des autres. Par conséquent, on peut considérer un filtre adaptatif dans le domaine fréquentiel comme la mise en parallèle de M filtres élémentaires à un coefficient complexe. Néanmoins, la FFT transforme l'espace d'observation temporel en un espace fréquentiel dual ; l'interprétation de la solution obtenue est donc différente par rapport à l'approche temporelle. Il est à noter que les méthodes fréquentielles montrent des performances supérieures aux méthodes temporelles pour le traitement de signaux à spectres de raies [Servièrre 89]. Nous nous intéresserons donc tout particulièrement à ces méthodes.

II.5. CRITERES DE SELECTION DES ALGORITHMES

La contrainte principale pour notre application est de fonctionner en temps réel sur un DSP avec une fréquence d'échantillonnage voisine de 20 KHz, afin de pouvoir effectuer une soustraction de bruit efficace dans la bande [0-10 KHz]. Nous devons respecter cette contrainte, tout en choisissant tant que possible un DSP à faible coût. Ceci implique pour les algorithmes les contraintes suivantes :

- bonnes performances en soustraction de bruit propre ;
- simplicité de mise en oeuvre ;
- robustesse par rapport aux instabilités numériques liées à la mise en oeuvre en précision finie, afin de garantir un fonctionnement en horizon temporel infini ;
- possibilité d'influer sur la robustesse de l'algorithme par rapport à la présence de signal utile dans la référence bruit. En effet, la dérivation des algorithmes adaptatifs repose sur l'hypothèse fondamentale de décorrélation entre le signal utile et le signal mesuré sur la voie référence ; or cette hypothèse risque d'être invalidée dans notre application (le couplage fluide-structure se traduisant par une pollution de la référence bruit accélérométrique par le signal utile acoustique : voir Chapitre I).

Afin de fixer les idées par rapport à la contrainte de fonctionnement en temps réel, une estimation de la complexité des principaux algorithmes temporels mentionnés précédemment est donnée dans le tableau suivant [Cioffi et Kailath 84]. Ces algorithmes sont donnés dans leur version avec fenêtrage exponentiel, ce fenêtrage permettant de prendre en compte d'éventuelles

non-stationnarités des signaux avec une complexité relativement faible [Cioffi et Kailath 84]. En ce qui concerne les algorithmes du domaine fréquentiel, leur complexité a été évaluée dans [Servièrre 89]. La complexité des algorithmes est ici exprimée en nombre de multiplications par échantillon de sortie, en fonction de l'ordre du filtre (M) pour les algorithmes temporels, ou de la taille des tranches (N) et du taux de recouvrement (α) pour les algorithmes fréquentiels ; en effet, tout processeur de traitement du signal est théoriquement capable d'effectuer une multiplication en parallèle avec une addition (MAC) ou avec une soustraction. Figure également pour chaque algorithme le nombre de divisions et racines carrées par échantillon de sortie, car ces opérations ne sont pas immédiates sur un DSP.

Un rapide calcul va nous permettre d'effectuer une première sélection parmi ces algorithmes ; en effet, les temps de cycle usuels des DSP courants à faible coût (Texas Instruments, Motorola, Analog Devices...) sont voisins de 50 nsec. Connaissant la fréquence d'échantillonnage des signaux que nous devons traiter (20 KHz), nous pouvons donner une approximation du nombre de cycles machine disponibles pour traiter chaque couple d'échantillons voie signal-voie référence : $(20 \text{ KHz} \times 50 \text{ nsec/cycle})^{-1} = 1000$ cycles disponibles par itération temporelle. Or, les simulations des algorithmes effectuées au chapitre III aboutissent à un ordre optimal du filtre (M) proche de 100 ; ceci limite donc a priori le choix des algorithmes temporels à ceux dont la complexité n'excède pas 10M opérations par itération. Par conséquent, nous pouvons déjà éliminer les algorithmes Fast Kalman, LSL, FTF normalisé et quasi-normalisé, ainsi que le LSL normalisé (voir Tableau 2). Il faut également considérer le fait que l'estimation de complexité donnée ci-dessous ne prend en compte que les opérations arithmétiques des algorithmes ; en plus de ces opérations, nous aurons notamment des lectures/écritures à effectuer en mémoire, ainsi que des sauvegardes de contexte si l'algorithme de soustraction de bruit est placé par exemple dans une routine d'interruption synchronisée sur l'échantillonnage des données (comme ce sera forcément le cas pour les algorithmes du domaine temporel). Cela contribue à augmenter la complexité des algorithmes dans une proportion que nous ignorons a priori, et à restreindre encore plus leur choix dans la perspective d'une mise en oeuvre en temps réel sur un DSP à faible coût. En ce qui concerne les algorithmes fréquentiels (Spectrofiltre adaptatif et Gradient fréquentiel), leur capacité à traiter les données par blocs les rend beaucoup moins dépendants de la contrainte de fonctionnement en temps réel.

Table 2: Estimation de complexité arithmétique par échantillon de sortie pour les principaux algorithmes temporels/fréquentiels

Algorithme	Complexité arithmétique	Auteur/Date
FTF	7M (2 divisions)	Cioffi et Kailath/1984
FAEST	7M (2 divisions)	Carayannis et al./1983
Fast Kalman	11M (2 divisions)	Morf et al./1978
LSL	14M (3M divisions)	Friedlander et al./1982

Table 2: Estimation de complexité arithmétique par échantillon de sortie pour les principaux algorithmes temporels/fréquentiels

Algorithme	Complexité arithmétique	Auteur/Date
FTF normalisé	11M (2 div. + 2 racines)	Cioffi et Kailath/1984
FTF quasi-normalisé	9M (2 div. + 2 racines)	Cioffi et Kailath/1984
LSL normalisé	29M (4M div. + 5M rac.)	Friedlander et al./1981
LMS	2M-3M (0-1 division)	Widrow et Hoff/1960
Spectrofiltre adaptatif (algorithme fréquentiel)	$(1.5\log_2(N)+5)/(1-\alpha)$ (1/1- α division)	Wiener
LMS fréquence (gradient fréquentiel)	$(2\log_2(N)+3)/(1-\alpha)$	Widrow

Ces considérations, en conjonction avec les simulations menées au chapitre III, nous permettent d'extraire trois algorithmes particulièrement intéressants pour notre application : les algorithmes LMS transverse (Least Mean Square) et FTF (Fast Transversal Filter) dans le domaine temporel, et l'algorithme du Spectrofiltre adaptatif dans le domaine fréquentiel. Le paragraphe suivant fait l'objet de la dérivation de ces trois algorithmes.

II.6. ALGORITHME LMS TRANSVERSE

II.6.1. OPTIMISATION D'UNE FONCTION DE M VARIABLES

Nous allons tout d'abord faire un bref rappel sur les méthodes newtoniennes d'optimisation d'une fonction de M variables, car elles interviennent directement dans la dérivation de l'algorithme LMS transverse. Le filtrage de Wiener définit une fonction de coût statistique quadratique d'un paramètre vectoriel, qui est pour nous le vecteur des M coefficients du filtre adaptatif RIF dans la représentation transversale directe. D'une manière tout à fait générale, notons $\underline{\theta}_M$ le paramètre vectoriel à optimiser ($\underline{\theta}_M = [\theta_1, \theta_2, \dots, \theta_M]^T$). On travaille désormais dans l'espace R^M muni de la norme euclidienne standard.

La fonction de coût statistique s'écrit :

$$J(\underline{\theta}_M) = E\left[\left(x(n) - \underline{\theta}_M^T \cdot \underline{U}(n)\right)^2\right] \quad (\text{EQ II-4})$$

où $\underline{U}(n)$ est le vecteur de régression de la référence bruit à l'instant n : $\underline{U}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^T$. Ce vecteur contient à chaque instant n les M échantillons les plus récents de la voie référence. La fonction de coût définit une surface d'erreur dans l'espace des M coefficients du filtre adaptatif. Le caractère quadratique de cette fonction simplifie grandement le problème d'optimisation, car les méthodes newtoniennes reposent sur l'approximation locale de la fonction de coût par une fonction quadratique. De plus, une fonction quadratique ne présente pas d'extrêma locaux parasites.

La fonction de coût définie par l'équation (EQ II-4) peut se développer de la manière suivante :

$$J(\underline{\theta}_M) = \underline{\theta}_M^T \cdot A \cdot \underline{\theta}_M + \underline{b}^T \cdot \underline{\theta}_M + c \quad (\text{EQ II-5})$$

où A est une matrice (M,M) réelle symétrique, \underline{b} un vecteur (M,1) réel et c un scalaire réel. La fonction de coût admet donc un unique point stationnaire $\underline{\theta}^{\text{opt}}$ tel que $J(\underline{\theta}^{\text{opt}}) = J_{\text{min}}$. Ce point est défini par les relations suivantes :

$$\nabla J(\underline{\theta}^{\text{opt}}) = \underline{0} \quad (\text{EQ II-6})$$

$$\Delta \underline{\theta}^T \cdot \nabla J(\underline{\theta}^{\text{opt}})^2 \cdot \Delta \underline{\theta} > 0 \quad (\text{EQ II-7})$$

pour tout déplacement $\Delta \underline{\theta}$ sur la surface d'erreur.

$\nabla J(\underline{\theta}^{\text{opt}})^2$ est la matrice du Hessian de la fonction J ; nous noterons cette matrice H. Notons que cette matrice est constante pour une fonction J quadratique.

Les méthodes newtoniennes ont en commun de se déplacer par itérations sur la surface d'erreur en suivant la direction de plus grande pente (modifiée ou non par un gain matriciel, qui permet de prendre en compte la géométrie de la surface en un point donné), afin de se rapprocher du minimum absolu de la surface :

- choix d'un point de départ $\underline{\theta}_M^0$ (arbitraire ou non) ;
- calcul itératif sur l'instant n :

$$\underline{\theta}_M^{n+1} = \underline{\theta}_M^n - \mu^n \cdot S^n \cdot \nabla J(\underline{\theta}_M^n) \quad (\text{EQ II-8})$$

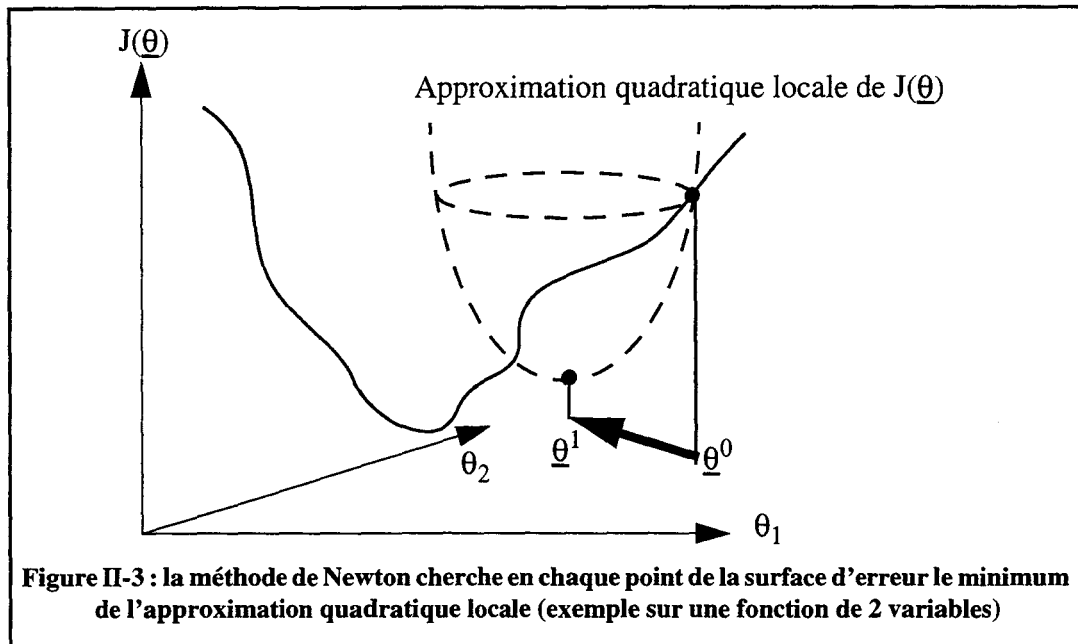
où (S^n) est une suite de matrices définies positives représentant les gains matriciels successifs de l'algorithme. μ^n définit le pas d'adaptation de l'algorithme à l'itération n. Ceci est la formulation la plus générale des méthodes newtoniennes d'optimisation d'une fonction de M variables.

A ce niveau, on fait la distinction entre les méthodes de type Gradient (c'est-à-dire sans gain matriciel : $S^n = I_M$ pour tout n) et les méthodes de type Quasi-Newton, pour lesquelles un gain matriciel permet de prendre en compte la géométrie exacte de la surface.

Avant d'introduire les méthodes Quasi-Newton d'optimisation, nous devons aborder la méthode de Newton. Cette méthode suppose que J est effectivement quadratique par rapport au paramètre à optimiser (voir Figure II-3). Elle possède la propriété intéressante de terminaison quadratique : appliquée à une fonction quadratique de M variables, elle trouve le minimum en un nombre fini d'itérations inférieur ou égal à M. En l'occurrence, la méthode de Newton trouve le minimum en une seule itération : à partir d'un point initial quelconque $\underline{\theta}_M^0$, le point $\underline{\theta}^{\text{opt}}$ définissant le minimum est obtenu en une seule itération par :

$$\underline{\theta}^{opt} = \underline{\theta}_M^0 - H^{-1} \cdot \nabla J(\underline{\theta}_M^0) \quad (\text{EQ II-9})$$

Néanmoins, cette méthode présente deux inconvénients majeurs : son coût de calculs est élevé à cause de l'inversion matricielle du Hessien ; de plus, elle présente un risque de divergence si la fonction n'est pas quadratique (voir Figure II-3). En effet, dans ce cas, le Hessien peut être non-défini positif en un point, auquel cas la direction de déplacement calculée sur la surface d'erreur n'est plus une direction de descente.



L'idée de base de la méthode Quasi-Newton est de construire une suite de matrices définies positives (S^n) convergeant vers $H(\underline{\theta}^{opt})^{-1}$. La contrainte fixée sur les matrices (S^n) permet d'assurer que chaque direction de déplacement est une direction de descente. La propriété de convergence globale est dès lors assurée, à condition de choisir un pas d'adaptation positif suffisamment petit. Cette propriété est essentielle pour une mise en oeuvre automatique, car elle permet d'éviter tout risque de divergence lorsque n tend vers l'infini. On peut même obtenir une convergence superlinéaire [Michaut 92] si la surface d'erreur est quadratique (ou si l'on est au voisinage du minimum de la surface d'erreur), à condition de choisir un pas d'adaptation optimal. Enfin, si on veut la propriété de terminaison quadratique, il faut en plus construire la suite (S^n) de manière à ce que les directions de déplacement successives correspondantes soient "conjuguées" entre elles [Huang 70].

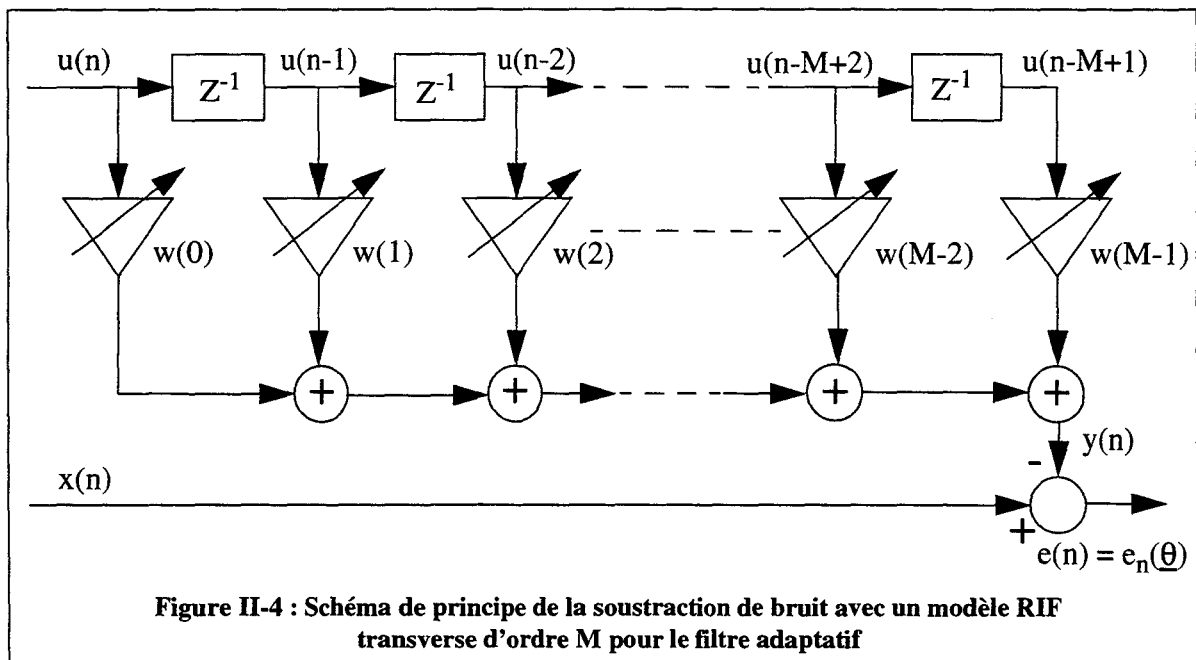
La méthode Quasi-Newton permet finalement de pallier les deux inconvénients principaux de la méthode de Newton. Si la propriété de terminaison quadratique est très difficile à obtenir, on peut malgré tout obtenir une convergence asymptotique superlinéaire en optimisant le pas d'adaptation. Enfin, la contrainte de convergence de la suite (S^n) garantit l'indépendance de la vitesse de convergence asymptotique par rapport au conditionnement du Hessien. Ceci revient à prendre en compte la géométrie de la surface de coût.

La méthode du Gradient repose sur l'approximation $S^n = I_M$ (matrice identité d'ordre M). I_M est toujours définie positive, donc la propriété de convergence globale est assurée à condition bien sûr de choisir un pas d'adaptation strictement positif assez petit. La descente est alors garantie, même sur une surface d'erreur non-quadratique. Par contre, le taux de convergence quadratique est seulement linéaire et dépend du conditionnement du Hessien (le taux de conditionnement du Hessien étant défini par le rapport de la valeur propre la plus faible à la valeur propre la plus forte). En particulier, si les valeurs propres sont très dispersées, le taux de convergence du gradient peut devenir très lent (cette situation correspond à des lignes de niveau de la surface d'erreur très excentriques, c'est-à-dire éloignées du cas hypersphérique dans un espace à M dimensions).

Le point commun des méthodes newtoniennes d'optimisation est de reposer sur l'approximation locale de la fonction de coût par une fonction quadratique à l'aide de la formule de Taylor. Nous verrons que le choix d'une structure de filtrage RIF conduit à une fonction de coût quadratique, donc ces méthodes sont indiquées pour la résolution du problème de la soustraction de bruit.

II.6.2. DERIVATION DE L'ALGORITHME LMS TRANSVERSE

Pour obtenir cet algorithme, on fait le choix d'un modèle RIF transverse d'ordre M pour le filtre adaptatif (voir Figure II-4). On utilise ensuite la représentation transversale directe dans l'approche de Wiener. L'approche de Wiener nécessite de travailler dans l'espace de Hilbert des variables aléatoires de carré intégrable, noté $L^2(\Omega)$.



On suppose toujours que les signaux sont réels, stationnaires au second ordre et à moyenne nulle. Notons $U_n = \text{Vect}\{u(n), u(n-1), \dots, u(n-M+1)\}$ le sous-espace de $L^2(\Omega)$ engendré par les M échantillons temporels les plus récents de la référence bruit seul.

Le paramètre à optimiser est le vecteur des M coefficients du filtre adaptatif : $\underline{w} = [w(0), w(1), \dots, w(M-1)]^T$. Le filtre doit fournir une estimation du bruit perturbateur $v(n)$ à partir de U_n :

$$y(n) = \hat{v}(n|U_n) = \sum_{k=0}^{M-1} w(k) \cdot u(n-k) \quad (\text{EQ II-10})$$

On a vu que sous l'hypothèse fondamentale de décorrélation entre le signal utile $s(n)$ et la référence bruit seul $u(n)$, minimiser la puissance du signal de sortie équivaut à minimiser la puissance du bruit résiduel. On choisit comme fonction de coût :

$$J(\underline{w}) = E \left[(x(n) - \underline{w}^T \cdot \underline{U}(n))^2 \right] \quad (\text{EQ II-11})$$

On peut développer cette fonction :

$$J(\underline{w}) = \underline{w}^T \cdot R \cdot \underline{w} - 2 \cdot \underline{p}^T \cdot \underline{w} + \sigma_x^2 \quad (\text{EQ II-12})$$

où $R = E\{\underline{U}(n) \cdot \underline{U}(n)^T\}$ est la matrice d'autocorrélation statistique de la référence bruit seul ;
 $\underline{p} = E\{x(n) \cdot \underline{U}(n)\}$ est le vecteur d'intercorrélacion entre la voie signal et la référence bruit seul.

Enfin, $\sigma_x^2 = E\{x(n)^2\}$ représente la variance de la voie signal.

La fonction de coût est bien quadratique en fonction de \underline{w} . Sa métrique est définie par la forme quadratique $\underline{w}^T \cdot R \cdot \underline{w}$, où R est réelle symétrique. De plus, elle possède la structure Toeplitz, qui découle directement de l'hypothèse de stationnarité de la référence bruit seul. Enfin, elle est définie non-négative, et même presque toujours définie strictement positive, à condition qu'il n'y ait pas de dépendance linéaire entre les termes $u(n-i)$ pour $i = 0, \dots, M-1$. On se place désormais dans cette hypothèse.

Le minimum absolu de la surface d'erreur est obtenu en annulant le vecteur gradient, ce qui conduit à l'équation normale statistique de Wiener, définissant le filtre optimal :

$$R \cdot \underline{w}^{opt} = \underline{p} \quad (\text{EQ II-13})$$

On l'appelle équation normale car elle s'interprète géométriquement par une projection orthogonale dans l'espace des variables aléatoires de carré intégrable [Haykin 93].

Pour aboutir à l'algorithme LMS-transverse, on utilise une méthode newtonienne d'optimisation de type Gradient sur la fonction de coût statistique. Cette démarche est tout à fait indiquée, car la fonction de coût est quadratique. Comme nous l'avons vu, la méthode du Gradient consiste à suivre à chaque itération n la direction de plus grande pente sur la surface d'erreur :

- choix d'un point de départ \underline{w}^0 ;
- calcul itératif :

$$\underline{w}^{n+1} = \underline{w}^n - \mu^n \cdot \nabla J(\underline{w}^n) \quad (\text{EQ II-14})$$

On peut exprimer facilement le vecteur gradient de la fonction de coût calculé au point \underline{w}^n en fonction de la matrice d'autocorrélation R et du vecteur d'intercorrélacion p :

$$\nabla J(\underline{w}^n) = 2 \cdot (R \cdot \underline{w}^n - p) \quad (\text{EQ II-15})$$

Ainsi, l'équation (EQ II-14) peut se réécrire de la manière suivante :

$$\underline{w}^{n+1} = \underline{w}^n - \mu^n \cdot (R \cdot \underline{w}^n - p) \quad (\text{EQ II-16})$$

Cette dernière équation définit l'algorithme du gradient statistique. Etant donnée la boucle de retour dans l'adaptation du filtre, il existe des conditions de stabilité sur cet algorithme. Il a été démontré [Widrow 75] que l'algorithme du gradient statistique à pas d'adaptation constant μ ne peut converger vers la solution optimale de Wiener que si et seulement si :

$$0 < \mu < 2/\lambda_{\max} \quad (\text{EQ II-17})$$

où λ_{\max} représente la valeur propre maximale de la matrice d'autocorrélation R . Le taux de convergence de cet algorithme dépendant du pas d'adaptation, il dépend également du conditionnement de la matrice R .

L'algorithme défini par l'équation (EQ II-16) est un algorithme statistique, c'est-à-dire qu'il fait intervenir les fonctions de corrélation statistiques des signaux. Le but étant d'aboutir à un algorithme qui fonctionne directement sur les données, on utilise l'approximation statistique directe du gradient, c'est-à-dire qu'on supprime l'opérateur d'espérance mathématique. Cette approximation revient en fait à remplacer le vecteur gradient par sa valeur instantanée. On obtient l'algorithme LMS transverse, plus justement appelé algorithme du gradient stochastique, dérivé par Widrow en 1975 [Widrow 75]. Cet algorithme est complètement défini par les deux équations suivantes :

$$\hat{\underline{w}}^{n+1} = \hat{\underline{w}}^n + \mu^n \cdot e(n) \cdot \underline{U}(n) \quad (\text{EQ II-18})$$

$$e(n) = x(n) - \left(\hat{\underline{w}}^n \right)^T \cdot \underline{U}(n) \quad (\text{EQ II-19})$$

Il est important d'évaluer l'impact de cette dernière approximation sur la stabilité et la convergence de l'algorithme. Cette analyse est relativement récente et complexe [Macchi 86]. Une grandeur intéressante est l'écart entre le filtre estimé à l'itération n et le filtre optimal (c'est-à-dire l'erreur instantanée) :

$$\underline{\hat{E}}^n = \hat{\underline{w}}^n - \underline{w}^{opt} \quad (\text{EQ II-20})$$

On peut montrer, en introduisant les grandeurs $F^n = \underline{U}(n) \cdot \underline{U}(n)^T$ et $\underline{C}^n = x(n) \cdot \underline{U}(n) - F^n \cdot \underline{w}^{opt}$ la relation de récurrence suivante sur l'erreur instantanée :

$$\underline{\hat{E}}^{n+1} = \left(I_M - \mu^n \cdot F^n \right) \cdot \underline{\hat{E}}^n + \mu^n \cdot \underline{C}^n \quad (\text{EQ II-21})$$

Cette équation représente l'évolution de l'état d'un système discret linéaire, non stationnaire, aléatoire, caractérisé par les matrices de transition $(I_M - \mu^n \cdot F^n)$ et l'entrée $\mu^n \cdot \underline{C}^n$. Une décomposition canonique de l'erreur instantanée a été trouvée [Macchi 86], qui permet de séparer l'étude du comportement asymptotique du LMS en deux aspects différents :

- le transitoire, qui représente l'influence des conditions initiales sur la phase d'approche ; c'est la période pendant laquelle le paramètre \underline{w}^n va évoluer de sa valeur initiale jusqu'au voisinage de \underline{w}^{opt} ;
- la fluctuation résiduelle, due à l'excitation aléatoire du système par l'entrée $\mu^n \cdot \underline{C}^n$.

Cette fluctuation résiduelle est liée au fait que le LMS fait une approximation bruitée du vecteur gradient en prenant sa valeur instantanée. Par conséquent -et à la différence du gradient statistique-, le paramètre \underline{w}^n va effectuer des fluctuations autour de \underline{w}^{opt} une fois la phase transitoire terminée. En effet, le gradient instantané n'est jamais rigoureusement nul autour de \underline{w}^{opt} .

L'étude rigoureuse du comportement asymptotique du LMS transverse est très compliquée. Néanmoins, on peut donner les résultats suivants :

- le LMS transverse à pas décroissant offre une convergence stricte vers \underline{w}^{opt} , mais il n'a de sens que dans un contexte fortement stationnaire puisqu'il perd progressivement sa capacité d'adaptation ;
- le LMS transverse à pas constant μ offre une convergence stricte du transitoire, mais il subsiste une Erreur Quadratique Moyenne Résiduelle (EQMR) que l'on peut néanmoins rendre arbitrairement petite en diminuant la valeur du pas d'adaptation. Une condition nécessaire de stabilité de cet algorithme est : $0 < \mu < 2/\alpha \cdot \text{Trace}(R)$ où α est un facteur de sécurité qu'il convient de choisir strictement supérieur à 1.

Enfin, il est important de remarquer qu'en contexte non-stationnaire, \underline{w}^{opt} varie avec le temps. Il faut alors ajouter à l'EQMR de fluctuation du LMS transverse à pas constant une EQMR de poursuite. On peut montrer que l'EQMR de fluctuation varie linéairement avec μ , alors que l'EQMR de poursuite varie en $1/\mu$. D'où un compromis à faire sur le choix de μ : il faut choisir μ le plus grand possible pour minimiser l'erreur de poursuite, tout en restant compatible avec la stabilité du LMS. Ces considérations mettent en évidence l'existence d'un pas optimal qui représente le meilleur compromis entre ces deux contraintes.

II.7. ALGORITHME FAST TRANSVERSAL FILTER (FTF)

Pour cet algorithme, on conserve un modèle RIF transverse d'ordre fini M . Au lieu d'utiliser l'approche par filtrage de Wiener, on va utiliser une approche des Moindres Carrés, dont la principale différence avec l'approche de Wiener est de prendre en compte dès le départ une fonction de coût calculable sur les données. L'algorithme FTF appartient à la famille des algorithmes RLS rapides, dont la complexité est en $o(M)$ contre $o(M^2)$ pour le RLS. L'idée de base de ces algorithmes est de supprimer les mises à jour matricielles du RLS, trop coûteuses en calculs ; pour cela, on va orthogonaliser l'espace d'observation, ce qui permet de calculer des projections orthogonales sans recourir à la résolution des équations normales empiriques. Pour comprendre ce procédé, nous allons dans un premier temps rappeler la méthode de dérivation de l'algorithme RLS avec facteur d'oubli exponentiel dans l'approche des Moindres Carrés.

II.7.1. ALGORITHME RLS AVEC FACTEUR D'OUBLI : DERIVATION

On remplace la fonction de coût statistique $J(\underline{w})$ (voir l'équation EQ II-11) par une suite de critères stochastiques $V_n(\underline{w})$ convergeant vers $J(\underline{w})$ quand n tend vers l'infini. L'intérêt est qu'à chaque instant n , on a une propriété exacte de minimisation du critère $V_n(\underline{w})$ pour le paramètre \underline{w}^n . De plus, la récursivité sur n apparaît naturellement dans le calcul. Le critère stochastique $V_n(\underline{w})$ n'est pas unique ; d'une manière générale, on définit un critère stochastique de la manière suivante :

$$V_{n, M, L}(\underline{w}) = \sum_{i=n-L+1}^n |e(i)|^2 \cdot \Omega_n(i) \quad (\text{EQ II-22})$$

avec :

$$e(i) = x(i) - (\underline{w}^i)^T \cdot \underline{U}(i) \quad (\text{EQ II-23})$$

M étant fixé à priori, il disparaît désormais des notations. L définit la largeur de la fenêtre utilisée, cette fenêtre étant définie par le terme $\Omega_n(i)$, qui doit prendre ses valeurs dans l'intervalle $[0 ; 1]$. Pour opérer dans un contexte faiblement non-stationnaire, nous choisissons un fenêtrage de type exponentiel, qui permet d'oublier les données anciennes avec une vitesse exponentielle liée au facteur d'oubli λ choisi [Cioffi et Kailath 85]. Cette fenêtre est de taille $L = n$ croissante avec le temps : on dit qu'elle est à mémoire infinie, puisqu'elle prend en compte toute l'histoire du processus. Il existe évidemment des fenêtres à mémoire finie, qui permettent notamment de s'adapter à des contextes plus fortement non-stationnaires.

La fonction de coût permettant de dériver l'algorithme RLS avec facteur d'oubli exponentiel s'écrit :

$$V_n(\underline{w}) = \sum_{i=1}^n |e(i)|^2 \cdot (1-\lambda) \cdot \lambda^{n-i} \quad (\text{EQ II-24})$$

Bien entendu, le facteur d'oubli λ doit être compris entre 0 et 1 strictement. Pour ce type de fe-

nêtrage, on travaille dans l'espace \mathbb{R}^n muni du produit scalaire suivant :

$$(\underline{X}, \underline{Y}) = \sum_{i=1}^n (1-\lambda) \cdot \lambda^{n-i} \cdot X(i) \cdot Y(i) \quad (\text{EQ II-25})$$

Il faut interpréter les équations temporelles suivantes en termes de vecteurs de \mathbb{R}^n :

$$x(n) = s(n) + \underline{U}(n)^T \cdot \underline{w}^{opt} \quad (\text{EQ II-26})$$

$$y(n) = \underline{U}(n)^T \cdot \underline{w}^n \quad (\text{EQ II-27})$$

$$e(n) = x(n) - y(n) \quad (\text{EQ II-28})$$

Notons :

- $\underline{E} = [\lambda^{(n-1)/2} \cdot e(1), \lambda^{(n-2)/2} \cdot e(2), \dots, e(n)]^T$
- $\underline{X} = [\lambda^{(n-1)/2} \cdot x(1), \lambda^{(n-2)/2} \cdot x(2), \dots, x(n)]^T$
- $\underline{S} = [\lambda^{(n-1)/2} \cdot s(1), \lambda^{(n-2)/2} \cdot s(2), \dots, s(n)]^T$
- $\underline{Y} = [\lambda^{(n-1)/2} \cdot y(1), \lambda^{(n-2)/2} \cdot y(2), \dots, y(n)]^T$

On peut alors réécrire les équations précédentes sous forme vectorielle :

$$\underline{X} = \Phi \cdot \underline{w}^{opt} + \underline{S} \quad (\text{EQ II-29})$$

$$\underline{Y} = \Phi \cdot \underline{w} \quad (\text{EQ II-30})$$

où $\Phi(n, M)$ est la matrice des données, dont les vecteurs lignes sont les vecteurs de régression $\underline{U}(i)$ pour $i = 1, \dots, n$. La fonction de coût définie par l'équation (EQ II-24) peut alors s'écrire

$$V_n(\underline{w}) = (1-\lambda) \cdot \|\underline{E}\|^2 \quad (\text{EQ II-31})$$

soit encore :

$$V_n(\underline{w}) = (1-\lambda) \cdot (\underline{X} - \Phi \cdot \underline{w})^T \cdot (\underline{X} - \Phi \cdot \underline{w}) \quad (\text{EQ II-32})$$

ce qui donne :

$$V_n(\underline{w}) = (1-\lambda) \cdot \left(\|\underline{X}\|^2 - 2 \cdot \underline{c}_n^T \cdot \underline{w} + \underline{w}^T \cdot \Gamma_n \cdot \underline{w} \right) \quad (\text{EQ II-33})$$

avec :

$$\underline{c}_n = \Phi^T \cdot \underline{X} \quad (\text{EQ II-34})$$

$$\Gamma_n = \Phi^T \cdot \Phi \quad (\text{EQ II-35})$$

La fonction de coût est quadratique en fonction du paramètre \underline{w} , elle a donc un minimum unique qui est le minimum global, que l'on obtient en annulant simplement le vecteur gradient. La solution globale pour ce critère est finalement l'équation normale stochastique :

$$\Gamma_n \cdot \underline{w}^n = \underline{c}_n \quad (\text{EQ II-36})$$

Pour qu'il y ait une solution unique, il faut que les colonnes de Φ soient linéairement indépendantes, ce qui n'est possible que pour $n > M$ avec le fenêtrage utilisé. En effet, tant que $n < M$, le rang de Φ est inférieur à M : Γ_n est alors singulière, donc non inversible. Néanmoins, on peut montrer que le problème des moindres carrés a une solution unique même pour $n < M$, c'est-à-dire lorsque le système est sous-déterminé. Il suffit dans ce cas d'introduire $\Phi^\#$ l'inverse généralisée de Moore-Penrose ; cette matrice définit alors une solution unique, qui est la solution à norme minimale dans le sens euclidien [Stewart 73]. On supposera désormais que les colonnes de Φ sont linéairement indépendantes.

On introduit la matrice d'autocorrélation stochastique : $R_n = (1 - \lambda)\Gamma_n$ et le vecteur d'intercorrélation stochastique : $p_n = (1 - \lambda)c_n$. Ces grandeurs sont définies de la manière suivante :

$$\frac{R_n}{(1-\lambda)} = \begin{bmatrix} \sum_{i=1}^n \lambda^{n-i} \cdot u^2(i) & \sum_{i=1}^n \lambda^{n-i} \cdot u(i) \cdot u(i-1) & \dots & \sum_{i=1}^n \lambda^{n-i} \cdot u(i) \cdot u(i-M+1) \\ \dots & \sum_{i=1}^n \lambda^{n-i} \cdot u^2(i-1) & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \sum_{i=1}^n \lambda^{n-i} \cdot u^2(i-M+1) \end{bmatrix} \quad (\text{EQ II-37})$$

$$\frac{p_n}{(1-\lambda)} = \begin{bmatrix} \sum_{i=1}^n \lambda^{n-i} \cdot x(i) \cdot u(i) \\ \sum_{i=1}^n \lambda^{n-i} \cdot x(i) \cdot u(i-1) \\ \dots \\ \sum_{i=1}^n \lambda^{n-i} \cdot x(i) \cdot u(i-M+1) \end{bmatrix} \quad (\text{EQ II-38})$$

Pour obtenir l'algorithme RLS avec facteur d'oubli exponentiel, on fait intervenir la récursivité en n sur les deux équations précédentes et on utilise la formule de Woodbury pour calculer récursivement la matrice d'autocorrélation stochastique inverse. On pourrait interpréter cet algorithme comme une approximation stochastique d'une méthode Quasi-Newton à pas fixe $(1-\lambda)$. Cette équivalence montre que l'introduction d'un facteur d'oubli strictement inférieur à 1 dote l'algorithme d'une capacité d'évolution permanente, ce qui est fondamental en contexte faiblement non-stationnaire. Enfin, pour garantir le caractère défini strictement positif des matrices à inverser pendant la phase de démarrage de l'algorithme ($n < M$), on initialise la matrice d'autocorrélation stochastique par une matrice $\delta \cdot I_M$, où δ est une constante strictement positive

intervenant comme une perturbation initiale volontaire, qui n'a aucune influence sur la vitesse de convergence de l'algorithme.

La méthode des Moindres Carrés, appliquée directement au problème, conduit à des algorithmes qu'il est possible de retrouver par l'approche de Wiener : RLS et RLS avec facteur d'oubli exponentiel. Même si ces algorithmes ont des propriétés intéressantes -notamment une vitesse de convergence indépendante du conditionnement du problème-, leur mise en oeuvre en temps réel est difficile, voire impossible à cause de leur complexité liée à la formulation matricielle utilisée. Nous allons conserver cette approche par Moindres Carrés, mais nous placer dans un cadre géométrique permettant d'obtenir des algorithmes transversaux rapides.

II.7.2. CADRE GEOMETRIQUE POUR LA DERIVATION DU FTF

On travaille désormais dans l'espace de Hilbert R^T où $T > N$ (N définit la taille maximale des observations). Au lieu de définir une fonction de coût statistique, on définit un critère stochastique (calculable sur les données) des Moindres Carrés à mémoire croissante :

$$V_n(\underline{w}) = \frac{1}{n} \cdot \sum_{i=1}^n |e(i)|^2 \quad (\text{EQ II-39})$$

et on définit le produit scalaire associé entre deux vecteurs réels \underline{X} , \underline{Y} de taille T :

$$(\underline{X}, \underline{Y}) = \sum_{i=1}^T X(i) \cdot Y(i) = \underline{X}^T \cdot \underline{Y} \quad (\text{EQ II-40})$$

On définit alors les vecteurs d'observation : $\underline{x}_n = [x(n), x(n-1), \dots, x(1), 0, \dots, 0]^T$. On travaille dans un espace vectoriel de dimension T constante, par différence avec l'approche du paragraphe précédent où l'on travaillait dans un espace vectoriel de dimension n . Le prolongement par 0 revient ici à supposer que les signaux sont nuls pour $n < 0$ (Prewindowed case). On définit une matrice des observations : $X_{k,j}(n) = [\underline{x}_{n-k}, \underline{x}_{n-k-1}, \dots, \underline{x}_{n-j}]$ de taille $(T, j-k+1)$. Cette matrice possède notamment la propriété de décalage :

$$X_{1,m}(n) = X_{0,m-1}(n-1) = z^{-1} \cdot X_{0,m-1}(n) \quad (\text{EQ II-41})$$

Contrairement à l'espace de Hilbert des variables aléatoires de carré intégrable, on perd ici la stationnarité du produit scalaire car $(\underline{x}_p, \underline{x}_q) = (\underline{x}_{p-1}, \underline{x}_{q-1}) + x(p) \cdot x(q)$. Pour rétablir la stationnarité du produit scalaire sur R^T , on définit un nouveau produit scalaire qui supprime la première composante par projection sur le sous-espace orthogonal au vecteur $\underline{\sigma} = [1, 0, \dots, 0]^T$. Ce vecteur intervient comme un annihilateur temporel : $(\underline{x}_p, P_{\underline{\sigma}} \underline{x}_q) = (\underline{x}_{p-1}, \underline{x}_{q-1})$. Pour deux matrices U (T, p) et V (T, q) compatibles, on étend alors le produit scalaire par la relation suivante :

$$\left(U, P_{\underline{\sigma}} V \right) = (z^{-1} \cdot U, z^{-1} \cdot V) \quad (\text{EQ II-42})$$

où $P_{\underline{\sigma}}$ représente le projecteur (opérateur matriciel) sur le sous-espace orthogonal au vecteur $\underline{\sigma}$.

On introduit le calcul des filtres prédicteurs direct et rétrograde, qui vont permettre d'orthogonaliser l'espace des observations. Le problème n'est plus de prédire un échantillon $u(n)$, mais un vecteur d'observation \underline{u}_n . Dans l'espace de Hilbert des variables aléatoires de carré intégrable, on prédit un échantillon $u(n)$ à partir d'une combinaison linéaire des M échantillons précédents. De même ici, on prédit un vecteur d'observation \underline{u}_n à partir de la matrice des observations $U_{1,M}(n)$. Cette matrice $U_{1,M}(n)$ définit un sous-espace de \mathbb{R}^T . La prédiction optimale au sens des Moindres Carrés $\hat{\underline{u}}_n^{\text{opt}}$ sera obtenue par projection orthogonale de \underline{u}_n sur $U_{1,M}(n)$. En effet, la projection orthogonale garantit que le vecteur d'erreur de prédiction est à norme minimale.

On peut alors écrire $\hat{\underline{u}}_n^{\text{opt}} = P_{1,M}(n) \underline{u}_n$, où $P_{1,M}(n)$ désigne le projecteur orthogonal sur le sous-espace $U_{1,M}(n)$. Dans ce cas, le vecteur d'erreur de prédiction avant d'ordre M sera :

$$f_{M,n} = \underline{u}_n - P_{1,M}(n) \underline{u}_n = P_{1,M}^{\perp}(n) \underline{u}_n \quad (\text{EQ II-43})$$

et l'erreur scalaire instantanée sera la première composante de ce vecteur :

$$f_M(n) = (\sigma, f_{M,n}) \quad (\text{EQ II-44})$$

De même pour la prédiction rétrograde : il s'agit de prédire le vecteur d'observation \underline{u}_{n-M} à partir de $U_{0,M-1}(n)$. La prédiction optimale au sens des Moindres Carrés $\hat{\underline{u}}_{n-M}^{\text{opt}}$ sera de la même façon obtenue par projection orthogonale de \underline{u}_{n-M} sur $U_{0,M-1}(n)$. Dans ce cas, le vecteur d'erreur de prédiction arrière d'ordre M sera :

$$b_{M,n} = \underline{u}_{n-M} - P_{0,M-1}(n) \underline{u}_{n-M} = P_{0,M-1}^{\perp}(n) \underline{u}_{n-M} \quad (\text{EQ II-45})$$

L'erreur scalaire instantanée $b_M(n)$ est la première composante de ce vecteur ; elle est obtenue de la même manière que $f_M(n)$.

Nous sommes donc amenés à définir des projecteurs orthogonaux. Les propriétés suivantes sont fondamentales et nous permettront de dériver l'algorithme FTF.

Soit $U = [\underline{u}_1, \underline{u}_2, \dots, \underline{u}_M]$ une collection de vecteurs libres de \mathbb{R}^T . Alors la matrice (T, T) du projecteur orthogonal sur le sous-espace U est donnée par :

$$P_U = U \cdot (U^T \cdot U)^{-1} \cdot U^T \quad (\text{EQ II-46})$$

On note P_U^{\perp} le projecteur orthogonal sur le sous-espace orthogonal à U . On a alors les propriétés suivantes :

$$P_U^2 = P_U \quad (\text{EQ II-47})$$

$$P_U \cdot P_U^{\perp} = 0 \quad (\text{EQ II-48})$$

et pour tous vecteurs $\underline{x}, \underline{y}$ de \mathbb{R}^T :

$$(\underline{x}, P_U \cdot \underline{y}) = (P_U \cdot \underline{x}, \underline{y}) = (P_U \cdot \underline{x}, P_U \cdot \underline{y}) \quad (\text{EQ II-49})$$

Soit $V = [U, \underline{v}]$ le sous-espace obtenu par concaténation avec U d'un vecteur d'observation sup-

plémentaire de \mathbb{R}^T . La relation entre le projecteur orthogonal sur U (P_U) et le projecteur orthogonal sur $[U, \underline{v}]$ (noté $P_{U, \underline{v}}$) est importante pour la dérivation du FTF. Pour trouver cette relation, il suffit d'utiliser la décomposition orthogonale suivante :

$$\underline{v} = P_U \cdot \underline{v} + P_{\underline{v}} \cdot \underline{v} = P_U \cdot \underline{v} + \underline{z} \quad (\text{EQ II-50})$$

D'où la concaténation suivante : $V = [U, \underline{v}] = U + \underline{z}$. Il vient alors la relation fondamentale :

$$P_{U, \underline{v}} = P_U + P_{\underline{v}} \cdot \frac{\left(\underline{v}, P_{\underline{v}} \right)}{\left(\underline{v}, P_{\underline{v}} \cdot \underline{v} \right)} \quad (\text{EQ II-51})$$

En utilisant les équations (EQ II-47), (EQ II-48) et (EQ II-49) on obtient également :

$$P_{\underline{v}} = P_U - P_U \cdot \frac{\left(\underline{v}, P_{\underline{v}} \right)}{\left(\underline{v}, P_{\underline{v}} \cdot \underline{v} \right)} \quad (\text{EQ II-52})$$

Ainsi, les filtres prédictifs directs et rétrogrades sont introduits dans le but d'orthogonaliser l'espace d'observation par la génération des erreurs de prédiction rétrograde. On fait naturellement apparaître la théorie des projecteurs orthogonaux, dont les équations (EQ II-47) à (EQ II-52) forment les propriétés de base.

II.7.3. ALGORITHME RAPIDE TRANSVERSAL : FTF

Le but de l'algorithme FTF est le traitement direct des Moindres Carrés sous forme transversale, en évitant l'utilisation de l'algorithme de Levinson, qui conduit à une représentation en treillis. On cherche à calculer à chaque instant n le filtre de Wiener optimal transverse d'ordre M qui permet d'estimer $v(n)$ à partir du vecteur de régression d'ordre M de la référence bruit : $\underline{U}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^T$. On remarque que $\underline{U}(n) = (\underline{\sigma}, U_{0, M-1}(n))$; l'espace d'observation est donc $U_{0, M-1}(n)$. Le filtre optimal sera obtenu en identifiant à chaque instant le projecteur orthogonal $P_{0, M-1}(n)$. Notons $\underline{w}^n = [w^n(0), w^n(1), \dots, w^n(M-1)]^T$ le vecteur des coefficients du filtre transversal d'ordre M à l'instant n .

Le critère à minimiser est défini par l'équation suivante :

$$V_n(\underline{w}) = \frac{1}{n} \cdot \sum_{i=1}^n |e(i)|^2 \quad (\text{EQ II-53})$$

Soit \underline{e}_n le vecteur dans \mathbb{R}^T des erreurs d'estimation (qui est aussi le vecteur du signal estimé, ou débruité, dans notre application). On peut écrire :

$$\underline{e}_n = \underline{x}_n - U_{0, M-1}(n) \cdot \underline{w}^n \quad (\text{EQ II-54})$$

Comme nous l'avons vu précédemment, le vecteur d'erreur optimal sera obtenu par projection orthogonale du vecteur \underline{x}_n sur le sous-espace orthogonal à $U_{0, M-1}(n)$. Le signal estimé à l'instant n sera donc obtenu en prenant la première composante du vecteur d'erreur optimal.

On a ici une approche vectorielle de minimisation d'un critère global (qui prend en compte toute l'histoire du processus vu le fenêtrage utilisé), qui permet d'aboutir à une solution instantanée.

On vient de voir que le filtre optimal \underline{w}^n est défini par la minimisation de la norme du vecteur d'erreur à postériori \underline{e}_n . Pour des commodités de calculs, nous allons également introduire le vecteur d'erreur à priori, défini par l'équation suivante :

$$\underline{\delta}_n = \underline{x}_n - U_{0, M-1}(n) \cdot \underline{w}^{n-1} \quad (\text{EQ II-55})$$

A la différence du vecteur d'erreur à postériori, le vecteur d'erreur à priori n'a aucune propriété d'optimalité et n'est pas obtenu par projection orthogonale. Il intervient ici uniquement comme un intermédiaire de calcul commode permettant de passer du filtre \underline{w}^{n-1} au filtre \underline{w}^n .

On peut définir le filtre optimal \underline{w}^n par la relation : $P_{0, M-1}(n) \cdot \underline{x}_n = U_{0, M-1}(n) \cdot \underline{w}^n$, où $P_{0, M-1}(n)$ est le projecteur orthogonal sur le sous-espace $U_{0, M-1}(n)$. En utilisant l'équation (EQ II-46), on obtient après simplifications :

$$\underline{w}^n = \left[U_{0, M-1}^T(n) \cdot U_{0, M-1}(n) \right]^{-1} \cdot U_{0, M-1}^T(n) \cdot \underline{x}_n \quad (\text{EQ II-56})$$

On peut ici définir l'opérateur transversal K_U par $P_U = U \cdot K_U$; cet opérateur a les propriétés suivantes :

$$K_U \cdot P_U = K_U \quad (\text{EQ II-57})$$

$$K_U \cdot U = I_M \quad (\text{EQ II-58})$$

On utilise l'équation (EQ II-51) pour exprimer l'opérateur $K_{U, \underline{v}}$ pour tout vecteur \underline{v} de \mathbb{R}^T . On obtient alors la mise à jour suivante :

$$K_{U, \underline{v}} = \begin{bmatrix} K_U \\ 0 \end{bmatrix} + \begin{bmatrix} -K_U \cdot \underline{v} \\ 1 \end{bmatrix} \cdot \left(\underline{v}, P_{\underline{U}} \cdot \underline{v} \right)^{-1} \cdot \left(\underline{v}, P_{\underline{U}} \right) \quad (\text{EQ II-59})$$

En choisissant $U = U_{0, M-1}(n)$, l'équation (EQ II-59) permet de trouver une relation de mise à jour temporelle sur le filtre \underline{w}^n . En effet, d'après l'équation (EQ II-56), on peut écrire les relations $\underline{w}^{n-1} = K_{0, M-1}(n-1) \cdot \underline{x}_{n-1}$ et $\underline{w}^n = K_{0, M-1}(n) \cdot \underline{x}_n$. On applique alors l'équation (EQ II-59) avec $U = U_{0, M-1}(n)$ et $\underline{v} = \underline{\sigma}$. Il vient :

$$K_{U, \underline{\sigma}} = \begin{bmatrix} K_U \\ 0 \end{bmatrix} + \begin{bmatrix} -K_U \cdot \underline{\sigma} \\ 1 \end{bmatrix} \cdot \left(\underline{\sigma}, P_{\underline{U}} \cdot \underline{\sigma} \right)^{-1} \cdot \left(\underline{\sigma}, P_{\underline{U}} \right) \quad (\text{EQ II-60})$$

On utilise la décomposition :

$$P_{U, \underline{\sigma}} \cdot \underline{x}_n = P_{\underline{\sigma}} \cdot \underline{x}_n + P_{P_{\underline{\sigma}} \cdot U} \cdot \underline{x}_n = P_{\underline{\sigma}} \cdot \underline{x}_n + P_W \cdot \underline{x}_n \quad (\text{EQ II-61})$$

ainsi que la relation de stationnarité du produit scalaire définie par l'équation (EQ II-42), et on obtient la relation de mise à jour temporelle de K_U où $U = U_{0, M-1}(n)$ et $V = U_{0, M-1}(n-1)$:

$$K_U \cdot x_n = K_V \cdot x_{n-1} + K_U \cdot \underline{\sigma} \cdot \frac{\left(\underline{\sigma}, P_U \cdot x_n \right)}{\left(\underline{\sigma}, P_U \cdot \underline{\sigma} \right)} \quad (\text{EQ II-62})$$

d'où la mise à jour temporelle du filtre transversal adaptatif :

$$\underline{w}^n = \underline{w}^{n-1} + K_{0, M-1}(n) \cdot \underline{\sigma} \cdot \frac{\left(\underline{\sigma}, P_{0, M-1} \cdot x_n \right)}{\left(\underline{\sigma}, P_{0, M-1} \cdot \underline{\sigma} \right)} \quad (\text{EQ II-63})$$

On remarque que $K_{0, M-1}(n) \cdot \underline{\sigma}$ est le vecteur à M coefficients qui permet de trouver la projection orthogonale du vecteur $\underline{\sigma}$ sur $U_{0, M-1}(n)$. Nous le noterons désormais \underline{g}^n . Ce vecteur est en fait le filtre de Wiener particulier obtenu pour estimer le vecteur $\underline{\sigma}$. On l'appelle gain de Kalman d'ordre M . Ainsi, l'introduction du vecteur $\underline{\sigma}$ permet de trouver une récurrence sur le filtre transversal ; ceci nécessite néanmoins l'introduction d'un autre filtre transversal dont le rôle est d'estimer $\underline{\sigma}$. D'autre part, le dénominateur dans l'équation (EQ II-63) représente l'angle entre $\underline{\sigma}$ et sa projection sur $U_{0, M-1}(n)$. Nous le noterons $\gamma(n)$: paramètre angulaire. Enfin, on reconnaît au numérateur (voir l'équation EQ II-54) le signal $e(n)$ estimé à l'instant n . On peut donc réécrire l'équation (EQ II-63) de la manière suivante :

$$\underline{w}^n = \underline{w}^{n-1} + \underline{g}^n \cdot \frac{e(n)}{\gamma(n)} \quad (\text{EQ II-64})$$

Pour mettre à jour le gain de Kalman d'ordre M \underline{g}^n , on calcule de deux manières différentes le gain de Kalman d'ordre $M+1$, noté \underline{g}_{M+1}^n . Cette double décomposition permet de relier la récurrence sur l'ordre à la récurrence temporelle qui nous intéresse. Pour la première décomposition, on applique l'équation (EQ II-59) avec $U = U_{1, M}(n)$ et $\underline{v} = \underline{u}_n$. On obtient alors :

$$\underline{g}_{M+1}^n = \begin{bmatrix} 0 \\ \underline{g}^{n-1} \end{bmatrix} + \frac{f_M(n)}{\alpha(n)} \cdot \begin{bmatrix} 1 \\ -\underline{a}^n \end{bmatrix} \quad (\text{EQ II-65})$$

où \underline{g}^n représente le filtre prédicteur direct d'ordre M et $\alpha(n)$ la puissance d'erreur de prédiction directe (norme du vecteur d'erreur de prédiction avant d'ordre M). Pour la seconde décomposition, on applique l'équation (EQ II-59) avec $U = U_{0, M-1}(n)$ et $\underline{v} = \underline{u}_{n-M}$. On obtient alors :

$$\underline{g}_{M+1}^n = \begin{bmatrix} \underline{g}^n \\ 0 \end{bmatrix} + \frac{b_M(n)}{\beta(n)} \cdot \begin{bmatrix} -\underline{b}^n \\ 1 \end{bmatrix} \quad (\text{EQ II-66})$$

où \underline{b}^n représente le filtre prédicteur rétrograde d'ordre M et $\beta(n)$ la puissance d'erreur de prédiction rétrograde (norme du vecteur d'erreur de prédiction arrière d'ordre M). La mise à jour du gain de Kalman fait donc intervenir les notions de prédiction directe et rétrograde. Les équations (EQ II-65) et (EQ II-66) vont nous permettre de mettre à jour le gain de Kalman d'ordre M . En effet, les M premières lignes de l'équation (EQ II-66) peuvent s'écrire :

$$\underline{g}^n = \underline{G}^n + \frac{b_M(n)}{\beta(n)} \cdot \underline{b}^n \quad (\text{EQ II-67})$$

où \underline{G}^n est le vecteur des M premières lignes du gain de Kalman d'ordre $M+1$.

Or $\underline{b}^n = K_{0,M-1}(n) \cdot \underline{u}_{n-M}$; en utilisant l'équation (EQ II-62), on obtient la mise à jour temporelle du filtre prédicteur rétrograde d'ordre M :

$$\underline{b}^n = \underline{b}^{n-1} + \underline{g}^n \cdot \frac{b_M(n)}{\gamma(n)} \quad (\text{EQ II-68})$$

Par suite, on obtient :

$$\underline{g}^n = [\underline{G}^n + k(n) \cdot \underline{b}^{n-1}] \cdot \left(1 - k(n) \cdot \frac{b_M(n)}{\gamma(n)}\right)^{-1} \quad (\text{EQ II-69})$$

avec $k(n) = b_M(n)/\beta(n)$.

La démarche consiste à calculer \underline{g}_{M+1}^n en connaissant \underline{g}^{n-1} et \underline{a}^n . Ensuite, on obtient \underline{g}^n en connaissant \underline{g}_{M+1}^n et \underline{b}^{n-1} . Pour obtenir $\underline{a}^n = K_{1,M}(n) \cdot \underline{u}_n$, on utilise l'équation (EQ II-62) :

$$\underline{a}^n = \underline{a}^{n-1} + \underline{g}^{n-1} \cdot \frac{f_M(n)}{\gamma(n-1)} \quad (\text{EQ II-70})$$

On remarque que les équations (EQ II-68) et (EQ II-70) font intervenir respectivement $b_M(n)$ et $f_M(n)$, appelées erreurs de prédiction à postériori car elles sont définies à partir des filtres prédicteurs calculés à l'instant n : $f_M(n) = u(n) - \underline{U}(n-1)^T \cdot \underline{a}^n$ et $b_M(n) = u(n-M) - \underline{U}(n)^T \cdot \underline{b}^n$. Or on peut montrer que $b_M(n) = \gamma(n) \cdot \bar{b}_M(n)$ où $\bar{b}_M(n) = u(n-M) - \underline{U}(n)^T \cdot \underline{b}^{n-1}$ est l'erreur de prédiction rétrograde à priori car elle est définie à partir du filtre prédicteur calculé à l'instant précédent. De même, on peut montrer que $f_M(n) = \gamma(n-1) \cdot \bar{f}_M(n)$, où $\bar{f}_M(n) = u(n) - \underline{U}(n-1)^T \cdot \underline{a}^{n-1}$ est l'erreur de prédiction directe à priori. Ainsi, à un instant n , on calcule facilement $\bar{f}_M(n)$ puis $f_M(n)$, qui est nécessaire au calcul de \underline{a}^n (voir l'équation EQ II-70). On peut ainsi calculer les erreurs de prédiction à postériori avant même le calcul des filtres prédicteurs.

Il reste à mettre à jour $\alpha(n)$, $\beta(n)$ et $\gamma(n)$. Pour les résidus $\alpha(n)$ et $\beta(n)$, on utilise la relation fondamentale de mise à jour d'un projecteur :

$$\alpha(n) = \alpha(n-1) + f_M(n) \cdot \bar{f}_M(n) \quad (\text{EQ II-71})$$

$$\beta(n) = \beta(n-1) + b_M(n) \cdot \bar{b}_M(n) \quad (\text{EQ II-72})$$

Enfin, pour la mise à jour du paramètre angulaire $\gamma(n)$ à l'ordre M , on va utiliser le même paramètre à l'ordre $M+1$, noté $\gamma_{M+1}(n)$. En utilisant les équations (EQ II-65) et (EQ II-66), on obtient les relations suivantes, qui achèvent de définir l'algorithme FTF :

$$\gamma_{M+1}(n) = \gamma(n-1) \cdot \frac{\alpha(n-1)}{\alpha(n)} \quad (\text{EQ II-73})$$

$$\gamma(n) = \frac{\gamma_{M+1}(n)}{1 - k(n) \cdot \bar{b}_M(n)} \quad (\text{EQ II-74})$$

On a ainsi rassemblé toutes les équations qui définissent l'algorithme FTF correspondant au critère des Moindres Carrés préfenêtrés à mémoire croissante.

On peut facilement en dériver une version à facteur d'oubli $\lambda < 1$ en utilisant un critère à fenêtre de pondération exponentielle (voir l'équation EQ II-24). On obtient alors un algorithme

à complexité en $8M$, à comparer favorablement à l'algorithme RLS- λ en $o(M^2)$. Cette complexité est comparable aux algorithmes FAEST (Fast A Posteriori Error Sequentiel Technique). Le FTF a été obtenu par Cioffi et Kailath en 1984 [Cioffi et Kailath 84], le FAEST par Carayannis et al. en 1983 [Carayannis et al. 83]. L'intérêt majeur du FTF par rapport aux algorithmes FAEST est qu'il utilise une théorie géométrique basée sur la mise à jour d'opérateurs de projection. Ceci facilite l'interprétation des variables internes de l'algorithme, et surtout permet de dériver assez facilement des variantes à contrôle de stabilité et/ou normalisées [Cioffi et Kailath 85] afin d'améliorer les propriétés de mise en oeuvre en précision finie. Par ailleurs, l'algorithme FTF dans sa version de base abaisse d'environ 40% le coût de calculs des algorithmes RLS-rapides (encore appelés Kalman-rapides dans la littérature) obtenus par Morf, Ljung et Falconer en 1978. Ces derniers algorithmes ne procèdent pas d'une approche géométrique, mais utilisent un partitionnement judicieux de la matrice d'autocorrélation de la référence bruit de telle sorte que chaque sous-matrice puisse être interprétée en termes de filtres prédicteurs direct/rétrograde. Les algorithmes RLS-rapides ont bien sûr l'avantage de faire passer le coût de calculs de la résolution du problème des Moindres Carrés de $o(M^2)$ à $o(M)$, mais ils n'exploitent pas comme FTF et FAEST l'ensemble des erreurs de prédiction.

Enfin, on peut remarquer que l'obtention de cet algorithme rapide est basée sur l'orthogonalisation en treillis de l'espace d'observation, puisque l'on mène le calcul des filtres de prédiction direct et rétrograde de $u(n)$ parallèlement au filtrage de Wiener. On évite simplement le détour par le treillis, qui repose sur la représentation des filtres de prédiction par leurs coefficients PARCOR (algorithme de Levinson), en utilisant une approche transversale directe. La mise à jour en temps réel des filtres prédicteurs sans le passage par l'algorithme de Levinson est rendue possible par une approche géométrique basée sur la mise à jour des opérateurs de projection orthogonale.

II.8. ALGORITHME DU SPECTROFILTRE ADAPTATIF

II.8.1. FILTRAGE ADAPTATIF DANS LE DOMAINE FREQUENTIEL

Le filtrage adaptatif dans le domaine fréquentiel se développe énormément, notamment avec les techniques de décomposition en sous-bandes, qui permettent d'adapter simplement et rapidement des filtres à longue réponse impulsionnelle [Shynk 92] [Morgan et Thi 95] [Jin et al. 96]. Ces techniques sont souvent très lourdes à mettre en oeuvre en temps réel. Nous allons par conséquent restreindre la présentation du filtrage adaptatif dans le domaine fréquentiel aux méthodes plus conventionnelles de type spectrofiltre adaptatif ou gradient fréquentiel. Ces méthodes, qui reposent sur un découpage des signaux temporels en blocs de données, sont toujours plus efficaces que les méthodes temporelles, qui travaillent échantillon par échantillon. En effet, les calculs de convolution/corrélation sont effectués plus rapidement dans le domaine fréquentiel.

Les méthodes fréquentielles permettent ainsi la résolution de l'approche de Wiener à un coût de calculs compatible avec une mise en oeuvre en temps réel sur DSP, grâce notamment aux algorithmes de FFT. La FFT peut être considérée comme une technique d'orthogonalisation des signaux d'entrée, car les canaux fréquentiels sont décorrélés deux à deux à condition que le nombre d'échantillons (N) sur lesquels est effectuée la FFT soit suffisamment grand. Sous cette hypothèse, on peut mettre à jour les N canaux fréquentiels du filtre adaptatif indépendamment les uns des autres ; par conséquent, un filtre adaptatif transversal dans le domaine fréquentiel peut être vu comme la mise en parallèle de N filtres élémentaires à un coefficient complexe.

Dans l'approche de Wiener, on minimise un critère temporel statistique (EQ II-4) ; de la même façon, on peut définir un critère fréquentiel statistique. Le filtre adaptatif est alors défini par ses N coefficients complexes $w^j(v_k)$ pour $k = 0, \dots, N-1$; j représente l'indice de la tranche de signal considérée.

Le filtrage de la voie référence par le filtre $\underline{w}(v)$ à N coefficients complexes se réduit à une simple multiplication complexe par canal fréquentiel. Le critère fréquentiel de Wiener s'écrit alors :

$$J(w) = E\left[\left|X_N(v) - w(v) \cdot U_N(v)\right|^2\right] \quad (\text{EQ II-75})$$

où $X_N(v)$ et $U_N(v)$ représentent respectivement les TFD des signaux $x(n)$ et $u(n)$, calculées sur une tranche de longueur N , pour le canal de fréquence v . Minimiser $J(w)$ par rapport à $w(v)$ revient à résoudre l'équation suivante :

$$E[X_N(v) \cdot U_N^*(v)] = w(v) \cdot E[U_N(v) \cdot U_N^*(v)] \quad (\text{EQ II-76})$$

En divisant cette équation par N et en faisant tendre N vers l'infini, on obtient :

$$w(v) = \frac{\gamma_{XU}(v)}{\gamma_{UU}(v)} \quad (\text{EQ II-77})$$

où $\gamma_{XU}(v)$ et $\gamma_{UU}(v)$ représentent respectivement la densité spectrale d'interaction entre la voie signal et la voie référence, et la densité spectrale de puissance de la voie référence évaluées sur le canal fréquentiel v . Cette dernière équation est l'équivalent de la solution temporelle de Wiener si on fait tendre N vers l'infini. En pratique, les approches de Wiener dans les domaines temporel et fréquentiel sont équivalentes si on choisit des tranches de signal de longueur N suffisante pour considérer que deux canaux fréquentiels adjacents sont décorrélés. Si N n'est pas suffisant, l'équation (EQ II-77) ne définit pas le filtre optimal car on a des termes de corrélation non-nuls entre les différents canaux fréquentiels.

II.8.2. DERIVATION DU SPECTROFILTRE ADAPTATIF

Le spectrofiltre est un algorithme non-récuratif qui met à jour le gain fréquentiel du filtre adaptatif sur des tranches de signal de longueur N (N disparaît désormais des notations par

conclusion) :

$$\hat{w}^j(\nu) = \frac{\hat{\gamma}_{XU}^j(\nu)}{\hat{\gamma}_{UU}^j(\nu)} \quad (\text{EQ II-78})$$

où $\hat{\gamma}_{XU}^j(\nu)$ (resp. $\hat{\gamma}_{UU}^j(\nu)$) est l'estimation de l'interspectre entre la voie signal et la référence bruit (resp. : estimation de l'autospectre de la référence bruit) après acquisition des tranches de signal de rang j , pour le canal fréquentiel ν . Pour les mêmes raisons que dans le domaine temporel, on réalise un moyennage exponentiel des densités spectrales par l'intermédiaire d'un facteur d'oubli λ , qui permet de prendre en compte une éventuelle non-stationnarité des signaux :

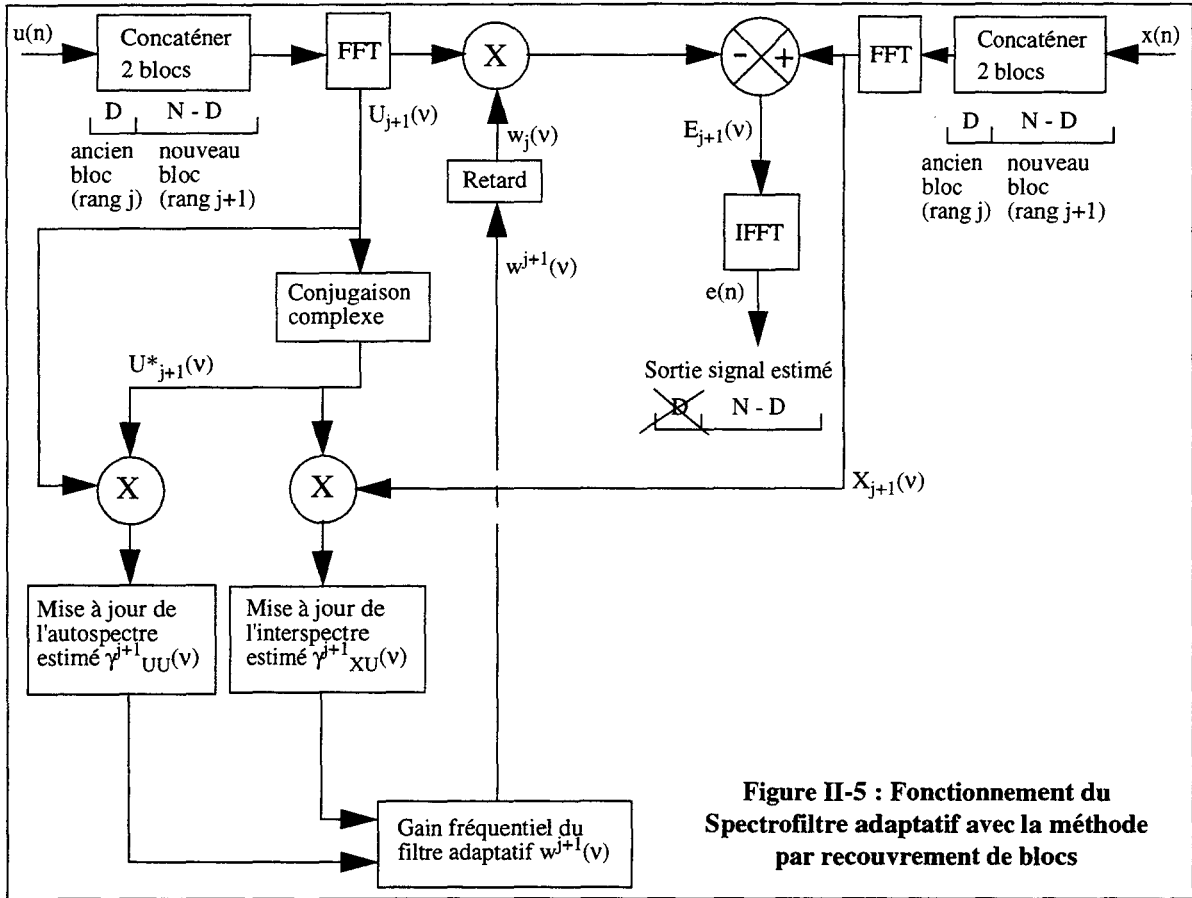
$$\hat{\gamma}_{XU}^j(\nu) = \lambda \cdot \hat{\gamma}_{XU}^{j-1}(\nu) + X_j(\nu) \cdot U_j^*(\nu) \quad (\text{EQ II-79})$$

$$\hat{\gamma}_{UU}^j(\nu) = \lambda \cdot \hat{\gamma}_{UU}^{j-1}(\nu) + U_j(\nu) \cdot U_j^*(\nu) \quad (\text{EQ II-80})$$

Le signal estimé dans le domaine fréquentiel s'écrit alors :

$$E_j(\nu) = X_j(\nu) - \hat{w}^{j-1}(\nu) \cdot U_j(\nu) \quad (\text{EQ II-81})$$

Pour retourner dans le domaine temporel, il suffit de faire une FFT inverse sur N points et on obtient N échantillons du signal $e(n)$ correspondant à la tranche de rang j . L'opération de filtrage dans le domaine fréquentiel correspondant à une convolution circulaire dans le domaine temporel, les premiers points de la tranche obtenue qui sont inclus dans le support temporel du filtre sont faux. Ceci induit la présence de transitoires de début de blocs. Plusieurs méthodes existent pour pallier ce problème, dont la plus efficace est la méthode par recouvrement de blocs (Overlap-Save Method) [Oppenheim et Schaffer 75][Servière et al. 89]. Cette méthode consiste à ne retenir que les estimations significatives (donc hors du transitoire de début de bloc) en travaillant sur des tranches temporelles se recouvrant partiellement (voir Figure II-5). Nous supprimons désormais ^ des notations, étant entendu que les grandeurs spectrales sont estimées à partir des données.



Le spectrofiltre adaptatif dans sa version de base est ainsi complètement défini ; néanmoins, certaines fonctionnalités supplémentaires peuvent lui être apportées en vue d'améliorer ses performances.

II.8.3. FONCTIONNALITES SUPPLEMENTAIRES

Nous avons envisagé trois fonctionnalités supplémentaires à apporter à l'algorithme du spectrofiltre adaptatif :

- contrôle de l'efficacité du traitement ;
- fenêtrage des données ;
- contrainte du gradient.

II.8.3.1. CONTROLE DE L'EFFICACITE DU TRAITEMENT

On sait [Guerre-Chaley 90] qu'un traitement peut ajouter plus de bruit qu'il n'en enlève. Il peut donc être nécessaire de mettre en oeuvre des processus automatiques de contrôle de l'efficacité des traitements employés. Nous envisageons deux types de contrôle. Le premier, "à priori", consiste en une analyse des signaux initiaux (voie signal et référence bruit seul) afin de déterminer si le traitement de soustraction de bruit sera efficace. Ce contrôle repose sur l'examen de la fonction de cohérence entre les deux voies. Le second, "à postériori", compare la puis-

sance de la voie signal avant et après traitement de soustraction de bruit afin de savoir si le traitement est réellement efficace. Une étude théorique sur ce sujet a déjà été réalisée [Guerre-Chaley 90]. Nous allons présenter ces deux types de contrôle dans le domaine fréquentiel et leur application aux algorithmes du spectrofiltre adaptatif et du gradient fréquentiel.

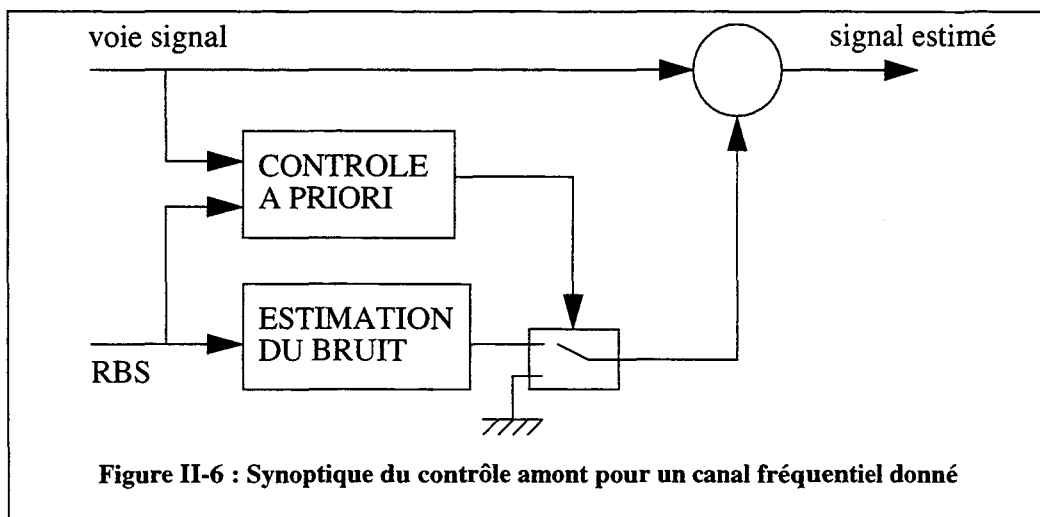
II.8.3.1.1 Contrôle à priori, ou amont

Ce contrôle repose sur une analyse préalable de la cohérence entre voie référence et voie signal. Cette cohérence est calculée pour chaque canal fréquentiel, puis comparée à un seuil prédéfini. Si la cohérence est supérieure au seuil, le traitement sera théoriquement efficace. Dans le cas contraire, on va forcer à zéro le gain fréquentiel du filtre adaptatif sur les canaux fréquentiels correspondants. Le problème principal consiste à déterminer le seuil ; il doit notamment permettre de discriminer entre une cohérence qui résulte d'un bruit de calcul et une cohérence qui traduit effectivement une corrélation entre les deux voies. Il faut également que la cohérence soit correctement estimée pour que le contrôle amont ne dégrade pas les performances du traitement.

Dans le cas du spectrofiltre adaptatif, le surcoût de calculs dû à l'estimation de la cohérence n'est pas très important car l'interspectre entre les deux voies et l'autospectre de la référence bruit sont déjà calculés par l'algorithme. On peut estimer la cohérence entre les deux voies après acquisition du bloc de rang j par la formule suivante :

$$|c_{XU}^j| = \frac{|\gamma_{XU}^j|}{\gamma_{XX}^j \cdot \gamma_{UU}^j} \quad (\text{EQ II-82})$$

Les densités spectrales intervenant dans cette formule sont calculées suivant les équations (EQ II-79) et (EQ II-80). Le synoptique du contrôle amont pour un canal fréquentiel est représenté sur la Figure II-6.



En ce qui concerne le calcul du seuil, des valeurs théoriques ont été trouvées pour les algorithmes du spectrofiltre et du gradient fréquentiel [*Guerre-Chaley 90*] ; en pratique, il convient de le déterminer de manière empirique, comme nous le verrons dans le chapitre IV.

II.8.3.1.2 Contrôle à postériori, ou aval

Contrairement au contrôle amont, le contrôle aval nécessite la réalisation de l'étape de soustraction de bruit. Le principe est simple : il suffit de comparer la puissance de la voie signal avant et après traitement pour chaque canal fréquentiel. Lorsqu'il n'y a pas de modification du signal utile, le traitement est efficace si la puissance du signal débruité est inférieure à la puissance de la voie signal. Si la puissance de sortie est inférieure à la puissance d'entrée, alors le signal estimé est le signal après soustraction ; dans le cas contraire, aucune soustraction de bruit n'est réalisée sur le canal fréquentiel considéré.

Le contrôle aval permet ainsi d'éviter que le traitement ne rajoute du bruit. Par ailleurs, il est plus facile à mettre en oeuvre que le contrôle amont, car il est indépendant de l'algorithme de soustraction de bruit utilisé. Dans [*Guerre-Chaley 90*], la supériorité du contrôle aval sur le contrôle amont est mise en évidence ; en effet, pour une même fenêtre d'intégration, il est démontré que la variance de l'estimateur de la puissance d'un signal est inférieure à celle de l'estimateur de la cohérence. Néanmoins, les deux types de contrôles sont à priori intéressants à mettre en oeuvre.

II.8.3.2. FENETRAGE DES DONNEES

D'une manière générale, toutes les méthodes fréquentielles reposent sur l'approximation de canaux fréquentiels décorrélés. Cette approximation est réaliste si on travaille sur des tranches de longueur N suffisamment grandes [*Servière 89*]. Augmenter la valeur de N permet donc d'améliorer la qualité de l'approximation de décorrélation des canaux fréquentiels ainsi que la résolution fréquentielle. Néanmoins, il a été démontré [*Servière 89*] pour l'algorithme du spectrofiltre adaptatif avec recouvrement de blocs que N intervient dans le biais et la variance de l'estimateur du filtre optimal. Le problème vient du fait que le biais et la variance varient en sens inverse par rapport à N : si on augmente trop la valeur de N , la variance de l'estimateur du filtre optimal va finir par augmenter. Il faut donc faire un choix judicieux de N pour obtenir un compromis acceptable entre le biais et la variance de l'estimateur du filtre optimal.

Pour diminuer le biais de l'estimateur du filtre sans augmenter sa variance, on peut utiliser un fenêtrage de chaque bloc de données avant l'opération de FFT. De plus, grâce au fenêtrage on améliorera l'approximation de décorrélation des canaux fréquentiels à N constant (par une meilleure réjection des lobes latéraux).

En ce qui concerne le choix des fenêtres, nous prendrons en compte les critères suivants (relatifs aux images fréquentielles des fenêtres utilisées) :

- pic central le plus étroit possible (pour ne pas dégrader le pouvoir de résolution);
- lobes latéraux d'amplitude la plus faible possible (pour améliorer l'approximation de décorrélation des canaux fréquentiels et limiter les fuites d'énergie) ;
- corrélation de recouvrement à 50% et 75% la plus faible possible (à cause du recouvrement des blocs, on perd l'indépendance entre les estimées des coefficients du filtre et les observations de la voie signal pour lesquelles sont utilisées ces estimées, d'où un risque plus grand de modifier le signal utile [*Guerre-Chaley 90*]).

Le critère sur l'amplitude des lobes latéraux est important pour notre application car on doit soustraire du bruit mécanique, qui se traduit par un spectre de raies d'amplitudes très différentes. Si on ne prend pas garde au phénomène de fuites d'énergie dans les lobes latéraux, des raies de forte puissance peuvent masquer d'autres éléments (raies de faible puissance par exemple) de la densité spectrale du signal étudié.

Par ailleurs, nous allons ajouter un critère essentiel : il faut que les fenêtres de pondération ne s'annulent pas sur les bords. En effet, après soustraction de bruit dans le domaine fréquentiel, il faut reconstituer le signal dans le domaine temporel et donc multiplier par la fenêtre inverse après IFFT. Nous prendrons également garde à ce que les fenêtres ne prennent pas des valeurs trop faibles sur les bords. En effet, à cause de la méthode par recouvrement de blocs, les données de fin de bloc seront comparativement moins prises en compte par le traitement ; la multiplication par la fenêtre inverse après IFFT risque alors d'amplifier les erreurs de calculs sur ces données.

Les caractéristiques principales des fenêtres de pondération que nous avons choisies pour notre application [*Harris 78*] figurent dans le tableau suivant. Ces fenêtres représentent évidemment un compromis entre les critères précédents. Pour chaque fenêtre, le tableau suivant quantifie les critères imposés.

Tableau 3 : Caractéristiques principales des fenêtres de pondération choisies.

	Niveau maximum du plus haut lobe latéral (en relatif par rapport au lobe principal) (dB)	Taux de décroissance des lobes latéraux (dB/octave)	Bande de bruit équivalente (bins)	Largeur de bande à 6 dB (bins)	Corrélation de recouvrement à 50% (en %)	Valeur minimale sur les bords
Rectangle	- 13	- 6	1.00	1.21	50	1.0
Hamming	- 43	- 6	1.36	1.81	23.5	8.0 E-2
Blackman-Harris à 3 termes	- 61	- 6	1.61	2.19	12.6	12.72 E-3

Tableau 3 : Caractéristiques principales des fenêtres de pondération choisies.

	Niveau maximum du plus haut lobe latéral (en relatif par rapport au lobe principal) (dB)	Taux de décroissance des lobes latéraux (dB/octave)	Bande de bruit équivalente (bins)	Largeur de bande à 6 dB (bins)	Corrélation de recouvrement à 50% (en %)	Valeur minimale sur les bords
Blackman-Harris minimum à 3 termes	- 67	- 6	1.71	1.81	9.6	49.0 E-4
Blackman-Harris à 4 termes	- 74	- 6	1.79	2.44	7.4	27.7 E-4

Nous évaluerons au chapitre IV les performances de ces fenêtres sur des signaux déterministes.

II.8.3.3. LA CONTRAINTE DU GRADIENT

Dans [Shynk 92] figure une procédure pour les algorithmes de type gradient fréquentiel qui permet de garantir que le calcul du vecteur gradient par corrélation rapide corresponde effectivement à une corrélation linéaire dans le domaine temporel : il s'agit de la "contrainte du gradient". Cette procédure alourdit les calculs, mais elle est censée garantir la convergence effective de l'algorithme vers la solution optimale de Wiener. Nous allons montrer dans un premier temps que le spectrofiltre adaptatif est équivalent au gradient fréquentiel, sous réserve d'un choix judicieux des conditions initiales. Etant donnée cette équivalence, il faudra trouver un moyen de prendre en compte la contrainte du gradient dans le spectrofiltre adaptatif.

II.8.3.3.1 Equivalence du gradient fréquentiel et du spectrofiltre

Nous appelons désormais algorithme du gradient fréquentiel l'algorithme suivant :

$$w^j(v) = w^{j-1}(v) + \frac{1}{P_j(v)} \cdot U^*_j(v) \cdot E_j(v) \quad (\text{EQ II-83})$$

avec $P_j(v) = \lambda \cdot P_{j-1}(v) + U_j(v) \cdot U^*_j(v)$

On peut réécrire le gradient fréquentiel de la manière suivante :

$$w^j(v) = w^{j-1}(v) + \frac{1}{P_j(v)} \cdot U^*_j(v) \cdot X_j(v) - \frac{1}{P_j(v)} \cdot w^{j-1}(v) \cdot U^*_j(v) \cdot U_j(v) \quad (\text{EQ II-84})$$

or par définition :

$$U_j(v) \cdot U^*_j(v) = P_j(v) - \lambda \cdot P_{j-1}(v) \quad (\text{EQ II-85})$$

d'où il vient après simplifications et en prenant comme condition initiale $w^0(v) = 0$:

$$w^j(v) = \frac{\sum_{i=1}^j \lambda^{j-i} \cdot U^*_i(v) \cdot X_i(v)}{P_j(v)} \quad (\text{EQ II-86})$$

qui converge vers le gain fréquentiel du filtre calculé avec la méthode du spectrofiltre adaptatif.

Le gradient fréquentiel est donc équivalent au spectrofiltre ; ce n'est pas un algorithme LMS à proprement parler, car le pas d'adaptation est normalisé dans chaque canal fréquentiel par la puissance de la référence bruit seul dans ce canal [Ogue et al. 83].

II.8.3.3.2 Application de la contrainte du gradient au gradient fréquentiel

Dans [Shynk 92], on trouve une procédure qui permet de garantir que le terme $U_{j(v)}^* \cdot E_{j(v)}$ corresponde effectivement à une corrélation linéaire dans le domaine temporel : c'est la contrainte du gradient. Pour comprendre cette contrainte, il faut repartir de l'algorithme équivalent au gradient par blocs dans le domaine temporel.

Dans cet algorithme, le filtre adaptatif n'est mis à jour qu'une fois par bloc de L échantillons. Notons j l'indice de bloc. On peut écrire $k = j.L + m$ avec $m = 0, 1, \dots, (L-1)$.

Isolons un bloc de la référence bruit : $\underline{u}_{jL} = [u(jL), u(jL+1), \dots, u(jL+L-1)]^T$. Pour ce bloc de rang j, l'estimation du signal s'écrit :

$$e(jL+m) = x(jL+m) - [\underline{w}^j]^T \cdot \underline{U}(jL+m) \quad (\text{EQ II-87})$$

pour $m = 0, 1, \dots, (L-1)$.

Soit : $\hat{v}(jL+m) = [\underline{w}^j]^T \cdot \underline{U}(jL+m)$ le résultat du filtrage à l'instant $(j.L+m)$. Il s'agit d'une convolution linéaire, qui fait intervenir les échantillons du bloc temporel de la référence bruit seul de rang $(j-1)$. On le voit plus clairement en traduisant l'équation précédente sous forme matricielle :

$$\begin{bmatrix} u(jL) & u(jL-1) & \dots & \dots & u(jL-M+1) \\ u(jL+1) & u(jL) & \dots & \dots & u(jL-M+2) \\ \dots & \dots & \dots & \dots & \dots \\ u(jL+M-1) & \dots & \dots & \dots & u(jL) \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ u(jL+L-1) & \dots & \dots & \dots & u(jL+L-M) \end{bmatrix} \cdot \begin{bmatrix} w^j(0) \\ w^j(1) \\ \dots \\ \dots \\ w^j(M-1) \end{bmatrix} = \begin{bmatrix} \hat{v}(jL) \\ \hat{v}(jL+1) \\ \dots \\ \dots \\ \hat{v}(jL+L-1) \end{bmatrix} \quad (\text{EQ II-88})$$

L'opération de convolution par bloc fait intervenir les échantillons $u(jL-M+1)$ à $u(jL-1)$, qui appartiennent au bloc de rang $(j-1)$. D'autre part, la mise à jour du filtre s'écrit :

$$\underline{w}^{j+1} = \underline{w}^j + \mu \cdot \underline{\Phi}^j \quad (\text{EQ II-89})$$

avec :

$$\underline{\Phi}^j = \sum_{m=0}^{L-1} \underline{U}(jL+m) \cdot e(jL+m) \quad (\text{EQ II-90})$$

Ce vecteur $(M,1)$ est une estimation du vecteur gradient pour le bloc de données de rang j. L'opération effectuée est une corrélation linéaire, qui fait donc intervenir les échantillons de la référence bruit seul de rang $(j-1)$. On peut reformuler l'équation (EQ II-90) sous forme matricielle :

$$\begin{bmatrix} u(jL) & u(jL+1) & \dots & u(jL+M-1) & \dots & \dots & u(jL+L-1) \\ u(jL-1) & u(jL) & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ u(jL-M+1) & u(jL-M+2) & \dots & u(jL) & \dots & \dots & u(jL+L-M) \end{bmatrix} \cdot \begin{bmatrix} e(jL) \\ e(jL+1) \\ \dots \\ e(jL+M-1) \\ \dots \\ \dots \\ e(jL+L-1) \end{bmatrix} = \begin{bmatrix} \Phi^j(0) \\ \Phi^j(1) \\ \dots \\ \dots \\ \dots \\ \Phi^j(M-1) \end{bmatrix} \quad (\text{EQ II-91})$$

Cette équation met clairement en évidence la contribution des échantillons du bloc de la référence bruit de rang $(j-1)$. On peut remarquer que c'est la même matrice qui intervient dans les équations (EQ II-88) et (EQ II-91), à la transposition près. Notons B_j cette matrice ; elle est rectangulaire de structure Toeplitz. On peut la rendre carrée en choisissant $L = M$; avec ce choix, le filtre traite un nombre de points égal au nombre de points utilisés dans le calcul du vecteur gradient. C'est le choix fait le plus couramment [Clark 83] [Shynk 92] [Haykin 96].

La mise en oeuvre du LMS-bloc dans le domaine temporel présente peu d'intérêt par rapport au LMS standard. En effet, bien que l'estimation du vecteur gradient soit meilleure grâce au moyennage sur le bloc, la contrainte de stabilité de l'algorithme sur le pas d'adaptation est nettement plus sévère [Haykin 96], ce qui rend cet algorithme inutilisable dans un contexte nécessitant une convergence rapide.

Au lieu d'utiliser l'approche par blocs dans le domaine temporel, on peut l'envisager dans le domaine fréquentiel ; en effet, l'idée de l'approche fréquentielle est de calculer rapidement les équations (EQ II-88) et (EQ II-91) par les opérations de FFT/IFFT.

On sait que le traitement fréquentiel sur un bloc isolé de rang j équivaut à une convolution/corrélation circulaire, où n'interviennent que les échantillons du bloc temporel de rang j . Tout se passe dans ce cas comme si \underline{u}_j était périodisé, et donc les équations (EQ II-88) et (EQ II-91) sont calculées avec une matrice B_j fausse (rendue circulante par périodisation implicite).

Comme expliqué précédemment, il existe deux méthodes pour extraire une convolution/corrélation linéaire à partir d'une convolution/corrélation circulaire : la méthode par recouvrement de blocs et la méthode par addition de blocs. Elles exploitent la constatation suivante : si deux séquences ont pour longueur respective N_1 et N_2 (avec $N_1 > N_2$), alors les $(N_1 - N_2 + 1)$ derniers points de la convolution circulaire sont exacts et les $(N_1 - N_2 + 1)$ premiers points de la corrélation circulaire sont exacts [Shynk 92]. On s'intéresse à la méthode par recouvrement de blocs, plus avantageuse dans le cadre d'une mise en oeuvre en temps réel. On notera désormais N la taille des FFT/IFFT utilisées, c'est-à-dire la taille des blocs. Le nombre de points de recouvrement (nombre d'échantillons communs à deux blocs consécutifs) sera noté D . Dans la littérature [Clark 83] [Shynk 92] [Haykin 96], le choix généralement fait est $N = 2M$ et $L = M$, ce qui revient à un recouvrement de 50%, puisqu'on a toujours la relation $N = L + D$.

La méthode par recouvrement de blocs consiste à extraire de la convolution circulaire les points exacts, c'est-à-dire ceux situés au-delà du support temporel du filtre. Avec cette méthode, les (M-1) premiers points obtenus sont faux à cause de la périodicité implicite de la FFT. Il suffit donc de découper le signal en tranches recouvrantes avec un nombre de points de recouvrement $D > M-1$ de façon à retrouver la convolution linéaire en concaténant les (N-D+1) points restants de chaque tranche de sortie. Chaque tranche de signal est de longueur N ; nous noterons désormais :

$$\underline{w}_{-a}^j = \begin{bmatrix} w^j(0) \\ w^j(1) \\ \dots \\ w^j(M-1) \\ \underline{0} \end{bmatrix} = \begin{bmatrix} \underline{w}^j \\ \underline{0}_{[N-M]} \end{bmatrix} \quad (\text{EQ II-92})$$

le vecteur des M coefficients du filtre adaptatif, augmenté de (N-M) zéros à la fin.

Par ailleurs, un bloc de la référence bruit de rang j sera noté :

$\underline{u}_{jL}^a = [u(jL-D), u(jL-D+1), \dots, u(jL-1), u(jL), u(jL+1), \dots, u(jL+L-1)]^T$ où les échantillons $u(jL-D)$ à $u(jL-1)$ appartiennent au bloc de rang (j-1) et les échantillons $u(jL)$ à $u(jL+L-1)$ appartiennent au bloc de rang j.

L'opération de filtrage est obtenue par :

$$\hat{v}_{jL}^a = \text{IFFT} \left\{ \text{FFT}(\underline{w}_{-a}^j) \cdot \text{FFT}(\underline{u}_{jL}^a) \right\} \quad (\text{EQ II-93})$$

où les D premiers points sont à éliminer à cause de la convolution circulaire et les L derniers points sont correctement estimés. On peut traduire l'équation (EQ II-93) sous forme matricielle :

$$\begin{bmatrix} u(jL-D) & u(jL+L-1) & \dots & \dots & u(jL-D+1) \\ u(jL-D+1) & u(jL-D) & \dots & \dots & u(jL-D+2) \\ \dots & \dots & \dots & \dots & \dots \\ u(jL) & \dots & \dots & \dots & u(jL+1) \\ u(jL+1) & u(jL) & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ u(jL+L-1) & u(jL+L-2) & \dots & \dots & u(jL-D) \end{bmatrix} \cdot \begin{bmatrix} w^j(0) \\ w^j(1) \\ \dots \\ \dots \\ w^j(M-1) \\ 0 \\ \dots \\ 0 \end{bmatrix} = \begin{bmatrix} \hat{v}(jL-D) \\ \hat{v}(jL-D+1) \\ \dots \\ \dots \\ \dots \\ \dots \\ \dots \\ \hat{v}(jL+L-1) \end{bmatrix} \quad (\text{EQ II-94})$$

soit encore :

$$\hat{v}_{jL}^a = B_j^a \cdot \underline{u}_{jL}^a \quad (\text{EQ II-95})$$

où B_j^a (N,N) est la matrice B_j (L,M) augmentée et rendue circulante à cause de la périodisation par FFT.

Il reste le problème de la mise à jour du filtre. On a obtenu le signal d'erreur pour le bloc de rang j : $\underline{e}_{jL}^a = [e(jL-D), e(jL-D+1), \dots, e(jL-1), e(jL), e(jL+1), \dots, e(jL+L-1)]^T$; pour estimer correctement la corrélation entre les séquences $\{e(jL), \dots, e(jL+L-1)\}$ et $\{u(jL), \dots, u(jL+L-1)\}$ sur les M premiers points qui correspondent au support temporel du filtre, il suffit de remplacer les D premiers points de \underline{e}_{jL}^a par des zéros : $\underline{e}'_{jL}^a = [0, \dots, 0, e(jL), e(jL+1), \dots, e(jL+L-1)]^T$ et on obtient le vecteur gradient augmenté calculé par corrélation rapide :

$$\underline{\Phi}^a = IFFT \left\{ FFT \left(\underline{e}'_{jL}^a \right) \cdot FFT \left(\underline{u}_{jL}^a \right) \right\} \quad (\text{EQ II-96})$$

De cette manière, seuls les M premiers points du vecteur gradient sont exacts ; les suivants font intervenir la corrélation circulaire. Le fait de forcer à zéro les D premiers points dans \underline{e}_{jL}^a permet de calculer le vecteur gradient sur les estimations du bloc courant ; on aurait pu remplacer ces points par les D points correctement estimés à la tranche précédente, mais dans ce cas le vecteur gradient n'aurait été exact que sur la première valeur.

De même qu'on élimine les D premiers points dans l'équation (EQ II-93) à cause de la convolution circulaire, on supprime les $(N-M)$ derniers points dans l'équation (EQ II-96) à cause de la corrélation circulaire (voir Figure II-7). C'est la contrainte du gradient : si elle n'est pas prise en compte, on risque de modifier le support temporel du filtre adaptatif.

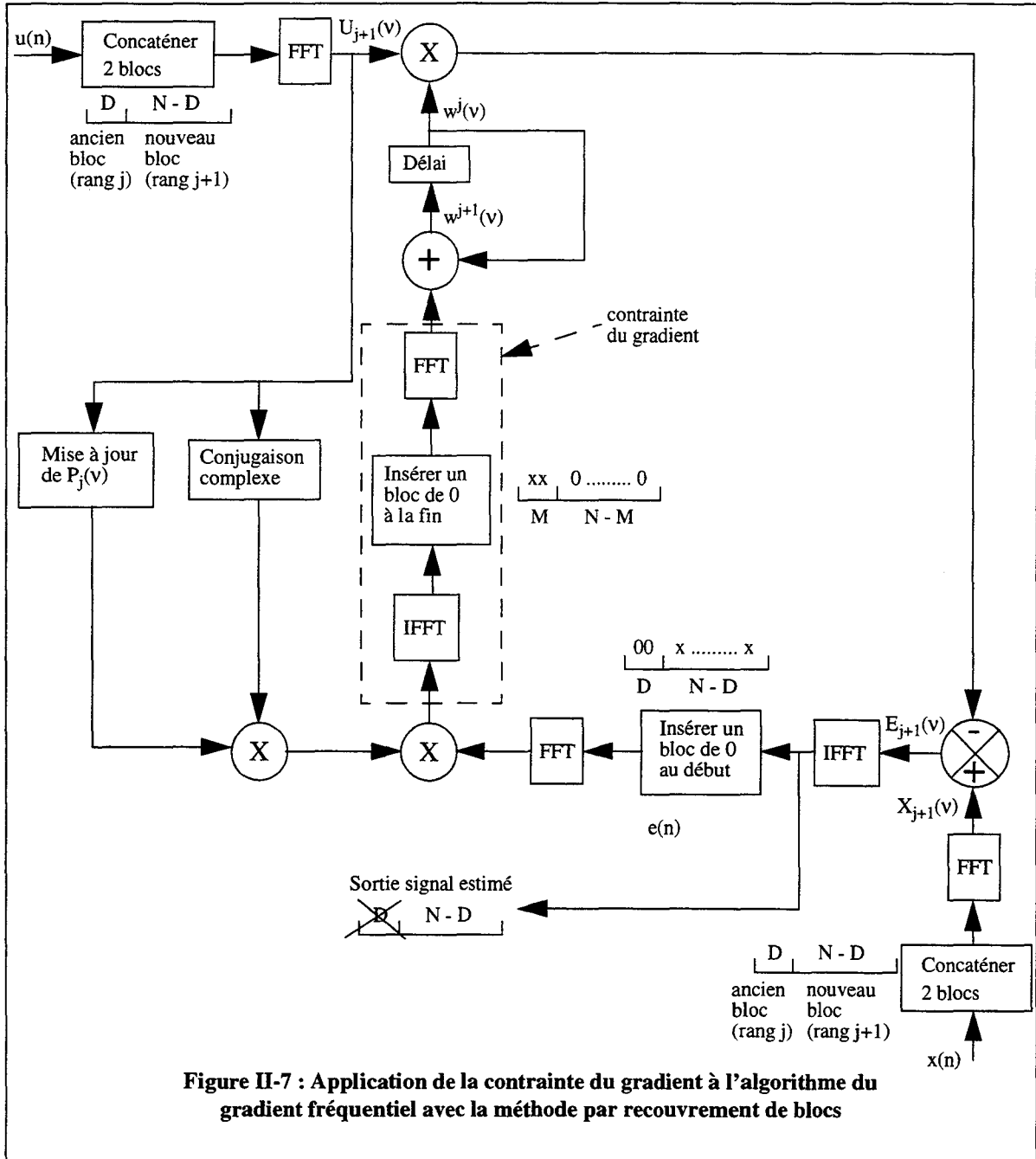


Figure II-7 : Application de la contrainte du gradient à l'algorithme du gradient fréquentiel avec la méthode par recouvrement de blocs

II.8.3.3 Application de la contrainte du gradient au spectrofiltre

On vient de voir que la contrainte du gradient permet d'éviter que le support temporel de la réponse impulsionnelle du filtre ne soit modifié à cause de la corrélation circulaire. Etant donnée l'équivalence entre les algorithmes du spectrofiltre et du gradient fréquentiel, il semble intéressant d'incorporer cette contrainte au spectrofiltre adaptatif. Néanmoins, cet algorithme ne fait pas intervenir un calcul explicite du vecteur gradient.

On peut malgré tout garantir la taille de la réponse impulsionnelle du filtre adaptatif en forçant à zéro ses $(N-M)$ derniers coefficients. Ceci nécessite une IFFT et une FFT supplémentaires par bloc de données.

II.8.4. REMARQUES IMPORTANTES

On choisit ici d'appliquer la méthode par recouvrement de blocs sur des blocs temporels de longueur N avec un recouvrement de D points. La seule contrainte est $D > M-1$ pour s'affranchir de la convolution circulaire. Avec ce choix, le nombre de points calculés pour un bloc est L . Il est tout à fait possible que $L > M$; dans ce cas, le vecteur gradient utilise plus d'informations que le filtre lui-même : on a alors un moyennage plus important sur les données, donc une estimation moins bruitée du vecteur gradient. Si on avait gardé la contrainte $L = M$, on aurait $D = N-L = N-M$, soit un taux de recouvrement $D/N = 1 - M/N$. Or il faut $D > M-1$, soit $N > 2M-1$ et le taux de recouvrement minimum serait alors de 50%. Si on veut améliorer la résolution fréquentielle, il faut augmenter N , mais avec la contrainte $L = M$, on augmente automatiquement le taux de recouvrement, ce qui est difficilement compatible avec les aspects temps réel. Sans cette contrainte, on choisit D indépendamment de N et on peut donc améliorer la résolution fréquentielle tout en restant compatible avec un calcul en temps réel.

Par ailleurs, chaque paramètre a une signification bien particulière pour le filtrage adaptatif dans le domaine fréquentiel. N doit être choisi en fonction de la résolution fréquentielle souhaitée dans l'application et du compromis biais-variance requis sur l'estimateur du filtre [Servière 89] ; la valeur de N conditionne également la qualité de l'approximation de décorrélation des canaux fréquentiels. D doit être choisi en fonction de la taille du support temporel de la réponse impulsionnelle du filtre à identifier pour éliminer les transitoires de début de bloc liés à la convolution circulaire. Une fois N et D choisis, L est fixé. Si on veut suivre un environnement non-stationnaire, il faut que L ne soit pas trop grand pour que l'adaptation du filtre soit faite le plus fréquemment possible. Dans ce cas, on peut soit diminuer N , mais c'est un mauvais choix car on perd en résolution fréquentielle ; une meilleure solution est d'augmenter D à N constant (tout en restant compatible avec la puissance de calcul disponible). Le support temporel du filtre est alors garanti par la contrainte du gradient.

II.9. CONCLUSION

Dans ce chapitre, nous avons étudié différentes méthodes de formulation de la procédure de soustraction de bruit, tant dans le domaine temporel que fréquentiel. Trois algorithmes : LMS, FTF et Spectrofiltre sont sélectionnés a priori sur la base de leurs performances et de la faisabilité de leur mise en oeuvre en temps réel sur un DSP à faible coût. Le choix d'une approche fréquentielle semble tout indiqué pour notre application, car les méthodes fréquentielles présentent des performances supérieures aux méthodes temporelles pour le traitement de signaux à spectre de raies [Servière 89] ; d'autre part, ces méthodes permettent des avantages considérables en temps de calcul.

II.10. BIBLIOGRAPHIE

- [Benallal et Gilloire 88] BENALLAL A., GILLOIRE A., A new method to stabilize fast RLS algorithms based on a first-order model of the propagation of numerical errors, IEEE Trans. on ASSP, 1988.
- [Carayanis et al. 83] CARAYANIS et al., A fast sequential algorithm for LS filtering and prediction, IEEE Trans. on ASSP, vol.31, p.1394, Décembre 1983.
- [Cioffi et Kailath 84] CIOFFI J.M., KAILATH T., Fast, Recursive Least Squares transversal filters for adaptive filtering, IEEE Trans. on ASSP, vol.32, p.304, Avril 1984.
- [Cioffi et Kailath 85] CIOFFI J.M., KAILATH T., Windowed Fast Transversal Filters adaptive algorithms with normalization, IEEE Trans. on ASSP, vol.33, p.607, Juin 1985.
- [Clark 83] CLARK et al., A unified approach to time- and frequency domain realization of FIR adaptive digital filters, IEEE Trans. on ASSP, vol.31, no.5, Octobre 1983.
- [Friedlander 82] FRIEDLANDER B., Lattice filters for adaptive processing, Proc. IEEE, vol.70, no.8, Août 1982.
- [Givens 58] GIVENS W., Computation of plane unitary rotations transforming a general matrix to triangular form, J. Soc. Ind. Appl. Math., vol.6, pp.26-50, 1958.
- [Guerre-Chaley 90] GUERRE-CHALEY J.F., Etude de différentes structures de soustracteurs de bruit adaptatifs multiréférences, Thèse de doctorat INPG, Décembre 1990.
- [Harris 78] HARRIS, On the use of windows for harmonic analysis with the DFT, Proc. IEEE, vol.66, no.1, Janvier 1978.
- [Haykin 93] HAYKIN S., Adaptive filter theory, Prentice Hall, 2nd ed., 1993.
- [Haykin 96] HAYKIN S., Adaptive filter theory, Prentice Hall, 3rd ed., 1996.
- [Jin et al. 96] JIN Q. et al., Optimum filter banks for signal decomposition and its application in adaptive echo cancellation, IEEE Trans. on Signal Processing, vol.44, no.7, p.1669, Juillet 1996.
- [Kung et Leiserson 78] KUNG H.T., LEISERSON C.E., Systolic arrays (for VLSI), Soc. Ind. Appl. Math., pp.256-282, 1978.
- [Macchi 86] MACCHI, Advances in adaptive filtering, Digital Communications, Elsevier Publishing, North-Holland, p.41, 1986.
- [Michaut 92] MICHAUT F., Méthodes adaptatives pour le signal, Editions Hermès, 1992.
- [Morgan et Thi 95] MORGAN D.R., THI J.C., A delayless subband adaptive filter ar-

- [Ogue et al. 83]* chitecture, IEEE Trans. on Signal Processing, vol.43, no.8, p.1819, Août 1995.
- [Oppenheim et Schaffer 75]* OGUE et al., A fast convergence frequency domain adaptive filter, IEEE Trans. on ASSP, vol.31, no.5, Octobre 1983.
- [Servière 89]* OPPENHEIM A.V., SCHAFER R.W., Digital Signal Processing, p.556, Prentice-Hall, 1975.
- [Servière et al. 89]* SERVIERE C., Éléments de comparaison entre différentes méthodes de soustraction de bruit, Thèse de doctorat INPG, Mai 1989.
- [Shynk 92]* SERVIERE et al., Comparaison de deux méthodes d'estimation du gain complexe d'un filtre optimal au sens de Wiener, Traitement du Signal, vol.6, 1989.
- [Slock et Kailath 88]* SHYNK J.J., Frequency domain and multirate adaptive filtering, IEEE Signal Processing Magazine, vol.9, pp.14-37, Janvier 1992.
- [Widrow 75]* SLOCK D.T.M., KAILATH T., Numerically stable fast Recursive Least-Squares transversal filters, IEEE Trans. on ASSP, p.1365, 1988.
- [Widrow et Stearns 85]* WIDROW et al., Adaptive noise cancelling : principles and applications, Proc. IEEE, vol.63, no.12, Décembre 1975.
- [Widrow et Stearns 85]* WIDROW B., STEARNS S.D., Adaptive signal processing, Prentice Hall, Signal Processing series, 1985.

III

SIMULATION DES ALGORITHMES ET MISE EN OEUVRE SUR DSP

III. SIMULATION DES ALGORITHMES ET MISE EN OEUVRE SUR DSP

Dans le chapitre précédent, nous avons mis en évidence l'intérêt que présentent certains algorithmes adaptatifs en termes de performances théoriques et surtout de coût de calculs, en vue d'une mise en oeuvre en temps réel sur un DSP à faible coût. Dans une première partie, ce chapitre fait l'objet d'une validation de ces algorithmes par simulation en langage C (simulations exécutées sur SPARCStation SUN et DSP TMS320C31 de Texas Instruments) sur des signaux réels d'acoustique sous-marine correspondant à des essais en lac, et sur les signaux générés par le modèle mis en place au Chapitre I. Dans une seconde partie, nous présentons les avantages de la plate-forme de développement que nous avons choisie (DSP TMS320C31) pour notre application, ainsi que la mise en oeuvre en temps réel des algorithmes les plus intéressants sur le DSP TMS320C31.

III.1. SIMULATION DES ALGORITHMES

III.1.1. SIMULATION SUR SIGNAUX EXPERIMENTAUX

III.1.1.1. CONDITIONS DE L'EXPERIMENTATION

Le but de cette expérimentation est l'acquisition des signaux temporels d'un hydrophone et d'un accéléromètre sur un tronçon de coque de sous-marin immergé et excité en vibration. Cette structure est équipée de 5 sources d'excitation (3 pots vibrants et 2 hauts-parleurs), d'une matrice de 64 accéléromètres placée en regard du panneau d'hydrophones à mesurer, et de 15 accéléromètres répartis sur la totalité de la structure. Le signal d'excitation sur les pots vibrants est un bruit blanc stationnaire dans la bande [1Hz-10KHz], mais à cause des fonctions de transfert et du montage mécanique du dispositif, la force injectée par le pot vibrant dans la structure présente évidemment des résonances.

Le test des algorithmes a été fait dans deux cas de figure :

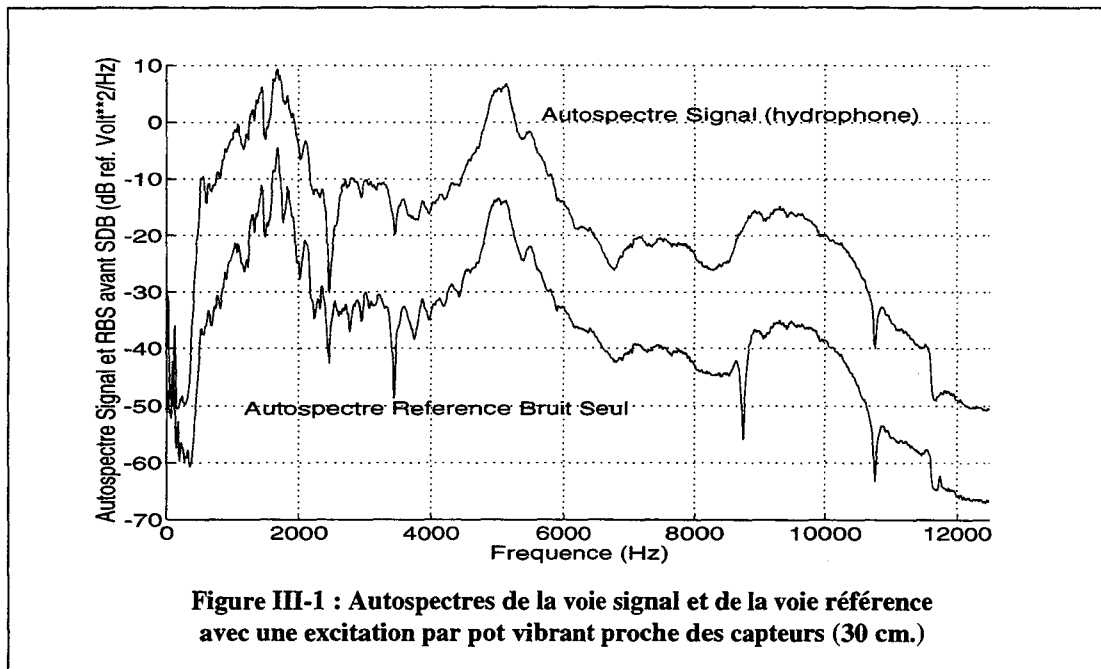
- excitation par pot vibrant proche des capteurs (distance : 30 cm.) ;
- excitation par pot vibrant éloigné des capteurs (distance : 210 cm.).

III.1.1.2. ANALYSE DES DONNEES

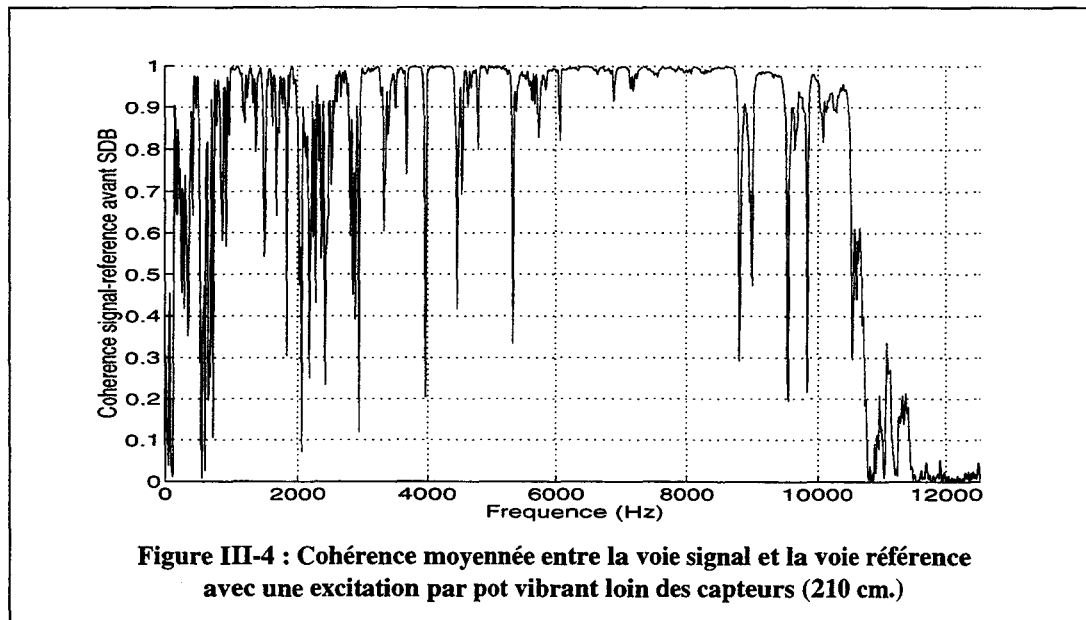
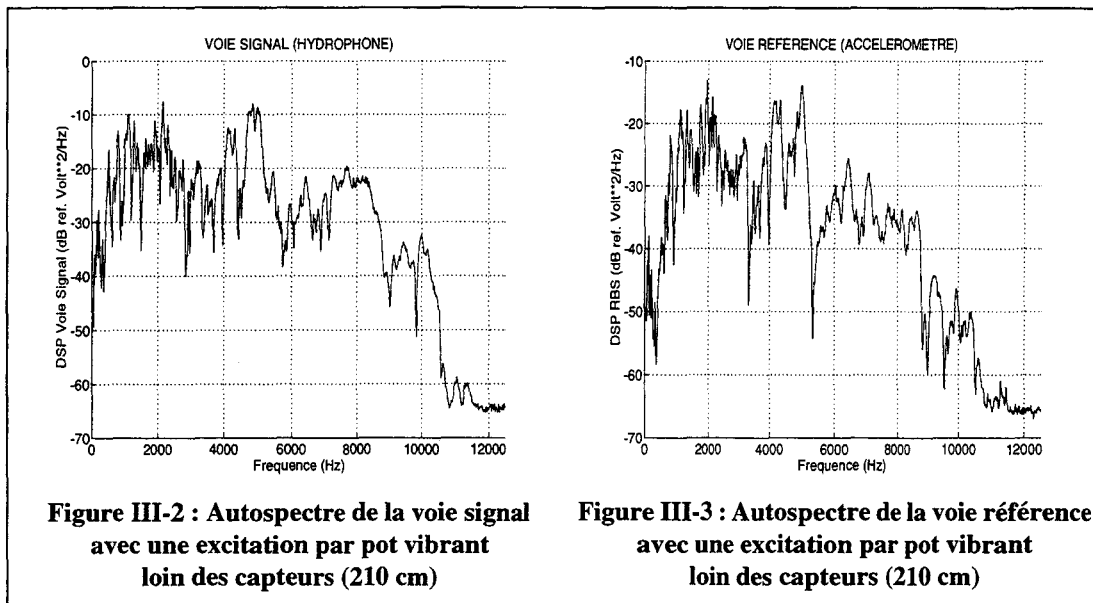
Dans le cadre de cette expérience, il n'y a aucun signal utile à détecter ; les capteurs hydrophone et accéléromètre sont soumis uniquement au bruit perturbateur résultant des vibrations de la structure. Nous avons estimé les densités spectrales de puissance des signaux mesurés par un doublet de capteurs hydrophone-accéléromètre, ainsi que la fonction de cohérence entre la voie signal (hydrophone) et la voie référence (accéléromètre) avant soustraction de bruit pour le premier cas de figure (voir Figure III-1). On constate que l'autospectre de la référence

bruit est quasiment la copie de l'autospectre de la voie signal. Néanmoins, on remarque qu'il manque une composante spectrale dans l'autospectre de la référence bruit autour de 8.7 KHz, ce qui se traduit par une dégradation de la cohérence entre les deux voies autour de cette fréquence. La cohérence entre la voie signal et la voie référence est voisine de 1 sur la bande de fréquences utile, ce qui traduit une excellente corrélation entre les signaux mesurés par ces capteurs. Par ailleurs, la voie signal et la voie référence sont stationnaires en puissance.

Les densités spectrales des signaux ont été estimées par la méthode du périodogramme moyenné de Welch sur environ 20 sec. de signal échantillonné à 25 KHz. Les signaux temporels sont découpés en tranches de 2048 points avec un recouvrement de 50% ; le périodogramme est calculé pour chaque tranche de signal après fenêtrage de Hanning, et on prend comme estimation de l'autospectre final la moyenne des valeurs obtenues sur chaque tranche. La résolution fréquentielle (en tenant compte du fenêtrage utilisé) est de 18.3 Hz.



Dans le second cas de figure, le pot vibrant est plus éloigné des capteurs (210 cm). La structure spectrale est plus fine pour la voie signal et la référence bruit (voir les Figures III-2 et III-3) ; la cohérence avant soustraction de bruit est un peu moins élevée sur l'ensemble du spectre par rapport au premier cas de figure (voir Figure III-4). Par ailleurs, la densité spectrale de puissance de la voie signal est moins élevée : ceci s'explique par un éloignement plus important du bruiteur par rapport aux capteurs. La voie signal et la voie référence sont toujours stationnaires en puissance.



III.1.1.3. METHODE D'EVALUATION DES ALGORITHMES

Pour évaluer les performances des algorithmes, nous avons isolé deux grandeurs représentatives de la qualité du traitement de soustraction de bruit [Servière 89] :

- le gain de traitement, défini par la différence entre l'auto-spectre du signal bruité et l'auto-spectre du signal estimé (exprimé en dB) ;
- la chute de cohérence, définie par la différence entre la cohérence voie signal-voie référence avant traitement et la cohérence voie signal-voie référence après traitement.

En pratique, un gain de traitement positif sur un canal fréquentiel correspond à une diminution du bruit perturbateur et non à une dégradation du signal utile seulement s'il y a en

même temps chute de cohérence. En ce qui concerne cette expérimentation, nous n'avons pas de signal utile à retrouver ; le but sera donc d'obtenir un gain de traitement et une perte de cohérence les plus élevés possibles sur l'ensemble du spectre étudié [0 Hz-10 KHz]. Il peut arriver que le gain de traitement soit négatif pour certains canaux fréquentiels ; ceci correspond toujours à une addition de bruit dû au traitement. S'il s'agit d'un bruit de calcul indépendant de la référence, il y a en même temps diminution de la cohérence sur ces canaux fréquentiels ; en revanche, si le bruit de calcul est corrélé à la référence, il y a augmentation de la cohérence entre la voie signal et la voie référence.

III.1.1.4. RESULTATS OBTENUS

Après avoir testé les différents algorithmes de soustraction de bruit présentés dans le Chapitre II sur ces signaux expérimentaux, nous en avons extrait trois qui ont donné les meilleurs résultats en termes de gain de traitement et chute de cohérence, sans entraîner un coût de calculs trop important. Dans le domaine temporel, il s'agit des algorithmes LMS et FTF ; dans le domaine fréquentiel, il s'agit de l'algorithme du spectrofiltre adaptatif.

Pour chaque algorithme, il existe un réglage optimal des paramètres [Servière 89] ; nous avons extrait ces paramètres sur environ 5 sec. de signal avant de valider les algorithmes sur les signaux temporels complets (soit environ 20 sec. de signal).

III.1.1.4.1 Algorithme FTF

La version de base du FTF présente d'excellents résultats en termes de gain de traitement et de chute de cohérence sur les signaux expérimentaux. Rappelons qu'il s'agit d'une méthode optimale à coût de calculs assez faible (7M opérations par itération), et qu'il existe de nombreuses versions dérivées de cet algorithme, permettant notamment une compensation des instabilités numériques liées à l'utilisation d'un facteur d'oubli exponentiel. Nous verrons que sur ce type de signaux, la version de base du FTF est suffisante dans la mesure où un choix judicieux des paramètres de l'algorithme garantit ici sa stabilité numérique.

Nous avons extrait le jeu optimal de paramètres et testé l'algorithme sur le premier cas de figure (bruiteur proche des capteurs). Puis, nous nous sommes écartés du réglage optimum des paramètres afin de mettre en évidence une dégradation des résultats.

En Annexe III-1 de ce chapitre figurent :

- l'autospectre de la voie signal avant et après soustraction de bruit (exprimé en dB) ;
- le gain de traitement (exprimé en dB) ;
- la chute de cohérence ;

pour les réglages suivants de l'algorithme :

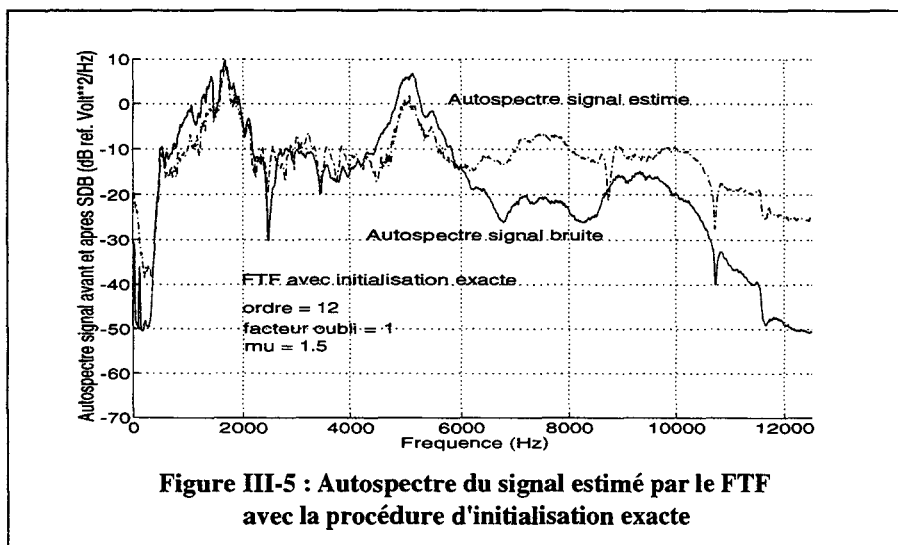
- M (ordre) = 128 , λ (facteur d'oubli) = 1 , $\alpha(0) = \beta(0) = 1$: réglage optimum ;
- $M = 32$, $\lambda = 1$, $\alpha(0) = \beta(0) = 1$;

- $M = 128$, $\lambda = 1$, $\alpha(0) = \beta(0) = 50$.

L'algorithme optimisé ainsi que les versions "dégradées" ont été testés sur les fichiers temporels complets, soit environ 20 sec. de signal. On constate que le gain de traitement et la chute de cohérence sont peu sensibles aux variations de l'ordre du filtre. La valeur optimale trouvée pour l'ordre du filtre correspond approximativement à la taille du support temporel de la fonction d'intercorrélacion entre la voie signal et la voie référence. Par ailleurs, les meilleurs résultats sont obtenus pour un facteur d'oubli de 1, ce qui est normal car on traite des signaux stationnaires. On observe une rapide divergence de l'algorithme pour un facteur d'oubli de 0.999, ce qui signifie qu'il sera difficile d'utiliser le FTF sur des signaux même faiblement non-stationnaires sans l'appoint d'une technique de stabilisation numérique. Concernant la condition initiale sur les erreurs quadratiques de prédiction directe et rétrograde cumulées $\alpha(0)$ et $\beta(0)$, il convient de la choisir la plus petite possible car elle intervient comme une erreur introduite volontairement avant toute estimation, qui détruit l'exactitude de la solution. Une valeur trop grande ralentirait l'algorithme ; malgré tout, on observe une valeur minimale sous laquelle la convergence n'est plus assurée (la condition initiale initialise une valeur qui intervient ensuite comme diviseur). En pratique, on constate l'existence d'une plage de valeurs assez importante pour cette condition initiale donnant des résultats corrects.

Les graphes précédents concernent la version de base du FTF ; il existe de nombreuses versions dérivées permettant un contrôle des instabilités numériques liées à la propagation des erreurs d'arrondi dans la structure récursive de l'algorithme. Bien qu'un choix judicieux des paramètres (en particulier : facteur d'oubli = 1) rende la version de base suffisante, nous avons simulé les versions à contrôle de stabilité les plus intéressantes.

Ainsi, nous avons pu vérifier que la procédure d'initialisation exacte [Cioffi et Kailath 84] souffre d'instabilité numérique dès qu'on augmente l'ordre du filtre : sur les signaux expérimentaux, on ne peut pas dépasser un ordre 12, ce qui donne des résultats nettement moins bons en termes de gain de traitement, notamment sur la partie haute du spectre (voir Figure III-5).



Par ailleurs, la version normalisée du FTF [Cioffi et Kailath 85] n'apporte pas d'amélioration significative par rapport à la version de base (voir Annexe III-1) ; de plus, elle entraîne une augmentation non négligeable de la complexité de l'algorithme (11M opérations par itération).

Finalement, le FTF "7M+M" [Cioffi et Kailath 84] s'avère la version la plus intéressante pour contrôler la stabilité numérique de l'algorithme. En effet, on garantit la stabilité numérique au prix d'une augmentation très modique du coût de calculs par rapport à la version de base (puisqu'on ajoute seulement M opérations par itération temporelle). Cette variante teste le paramètre angulaire à chaque itération et corrige le gain de Kalman lorsque ce paramètre sort de son intervalle de définition [0 ; 1] (on peut montrer que ce paramètre angulaire s'interprète également comme le rapport des erreurs de prédiction à posteriori sur les erreurs de prédiction à priori [Bellanger 89]). Le jeu de paramètres optimal pour le FTF "7M+M" est le même que pour la version de base. Nous avons testé cet algorithme sur les fichiers temporels complets correspondant au premier cas de figure (bruiteur proche des capteurs). Cet algorithme donne des résultats rigoureusement identiques à la version de base en termes de gain de traitement et de perte de cohérence. Par ailleurs, l'observation d'une grandeur interne du FTF "7M+M" nous donne des informations intéressantes sur son comportement (voir Figure III-6).

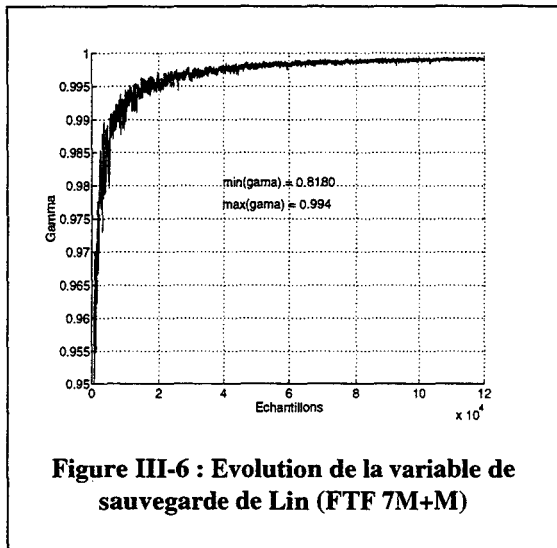


Figure III-6 : Evolution de la variable de sauvegarde de Lin (FTF 7M+M)

La variable de sauvegarde de Lin [Lin 84] reste positive et tend vers l'unité ; ceci traduit d'une part l'absence d'accumulation d'erreurs d'arrondi, d'autre part une bonne convergence de la prédiction rétrograde. Par ailleurs, le paramètre angulaire reste inférieur à l'unité, ce qui est une condition nécessaire à la convergence de l'algorithme.

III.1.1.4.2 Algorithme LMS temporel

Pour mémoire, rappelons que cet algorithme est sous-optimal, mais se distingue par sa simplicité, son faible coût de calculs (2M opérations par itération), ainsi que son aptitude théorique à la poursuite de signaux faiblement non-stationnaires.

Après avoir extrait le jeu de paramètres optimal pour cet algorithme, nous l'avons testé dans le premier cas de figure (bruiteur proche des capteurs). Puis, pour valider ces résultats, nous nous sommes écartés du réglage optimum des paramètres et nous avons mis en évidence une dégradation des résultats. Les simulations confirment les tendances dégagées dans [Servière 89], avec en particulier l'existence d'un ordre optimum pour le filtre adaptatif, ainsi que la nécessité d'un compromis à faire sur la valeur du pas d'adaptation entre une vitesse de convergence élevée et une faible erreur résiduelle. De même que pour le FTF, on trouve un ordre optimal de 128 pour le filtre adaptatif. En ce qui concerne le pas d'adaptation, il faut trouver un compromis en choisissant une valeur assez petite pour obtenir une erreur résiduelle faible, sans pour autant altérer le pouvoir adaptatif du système. Au vu des simulations effectuées, une valeur de 1 pour le pas d'adaptation réalise le bon compromis. Bien qu'il s'agisse d'un algorithme sous-optimal, le LMS donne des résultats comparables à ceux du FTF sur ces signaux expérimentaux pour un réglage optimal des paramètres.

III.1.1.4.3 Algorithme du Spectrofiltre adaptatif

On a représenté pour cet algorithme le gain de traitement et la chute de cohérence pour le réglage optimal des paramètres, puis nous nous sommes écartés de ce réglage pour faire apparaître une tendance à la dégradation des résultats (voir Annexe III-2).

On s'est placé dans les cas suivants :

- N (taille des FFT) = 1024 ; D/N (taux de recouvrement) = 50% ; λ (facteur d'oubli) = 1 : réglage optimum des paramètres ;
- $N = 1024$; $D/N = 50\%$; $\lambda = 0.999$;
- $N = 1024$; $D/N = 75\%$; $\lambda = 1$;
- $N = 2048$; $D/N = 50\%$; $\lambda = 1$.

Ces simulations ont permis de mettre en évidence les points suivants :

- robustesse de l'algorithme par rapport à l'utilisation d'un facteur d'oubli significatif (c'est-à-dire strictement inférieur à 1) ;
- existence d'un taux de recouvrement qui représente le meilleur compromis entre performances et coût de calculs (50 %) ;
- existence d'une valeur optimale de N au-dessus de laquelle le gain de traitement diminue nettement.

III.1.2. SIMULATION SUR LES SIGNAUX EXTRAITS DU MODELE

III.1.2.1. GENERALITES

Les signaux expérimentaux précédents fournissent une bonne base pour simuler et tester les algorithmes. Malheureusement, il n'y a pas de signal utile à retrouver. Pour pallier cet

inconvenient, nous allons utiliser les signaux générés par le modèle physique développé au Chapitre I.

Les simulations sur signaux expérimentaux ont confirmé l'intérêt que présentent les algorithmes LMS, FTF et Spectrofiltre adaptatif par rapport aux autres méthodes existantes. Par souci de concision, nous nous limitons aux résultats obtenus par ces algorithmes sur les signaux extraits du modèle. Pour chaque algorithme, nous utilisons le jeu de paramètres optimal mis en évidence sur les simulations précédentes. Un autre résultat est présenté, qui indique une tendance à la dégradation lorsqu'on s'écarte de l'optimum du réglage des différents paramètres. L'horizon temporel examiné est trop court (0.7 sec.) pour accéder aux notions de temps de convergence et de stabilité à long terme des algorithmes. Néanmoins, ces signaux nous permettent de traiter un cas classique de soustraction de bruit : l'élimination de raies fines en présence d'un signal acoustique large bande.

III.1.2.2. RESULTATS OBTENUS

En Annexes III-3 et III-4 de ce chapitre figurent les résultats obtenus respectivement par le FTF "7M+M" et le Spectrofiltre adaptatif sur les signaux générés par le modèle physique. Ces signaux prennent ici en compte l'ajout de raies de bruit dans le spectre de la force d'excitation.

La version optimisée du FTF "7M+M" présente un gain de traitement positif sur la plupart des raies du bruiteur. Ce gain s'accompagne d'une perte de cohérence significative : il y a donc bien diminution du bruit perturbateur. On constate également qu'on ne soustrait absolument pas de signal utile. Si on change le facteur d'oubli (de 1 à 0.999), on observe une nette tendance à la dégradation des résultats : la soustraction de bruit est moins bonne sur la plupart des raies de bruit (voir Annexe III-3). Il est à noter que l'on obtient à nouveau des résultats tout à fait comparables avec l'algorithme du LMS temporel.

Pour obtenir de bons résultats avec le Spectrofiltre, il a fallu augmenter le taux de recouvrement à 75%. On obtient avec le jeu de paramètres optimal des résultats analogues au FTF "7M+M" en termes de gain de traitement et perte de cohérence ; malgré tout, deux accidents se produisent au voisinage des raies 1 et 6, qui correspondent à une addition de bruit dû au traitement (gain de traitement négatif). Ce phénomène est amplifié lorsqu'on s'écarte du réglage optimum des paramètres, par exemple lorsqu'on abaisse le taux de recouvrement à 50% (voir Annexe III-4).

Pour achever cette phase de validation, nous avons testé les algorithmes sur les mêmes signaux sans les raies ajoutées dans le spectre de la force d'excitation. Nous avons déjà constaté que dans ce cas, la cohérence entre voie signal et voie référence est trop faible pour espérer un gain notable en soustraction de bruit. C'est effectivement ce que l'on observe en testant par exemple le FTF "7M+M" sur les mêmes signaux privés des raies introduites dans le spectre de la force d'excitation. On trouve un gain de traitement large bande inférieur à 1 dB, et la chute de cohérence associée demeure inférieure à 0.2.

III.1.3. CONCLUSION

Nous avons traité des signaux stationnaires avec les algorithmes LMS, FTF et Spectrofiltre adaptatif. Globalement, la soustraction de bruit fonctionne bien ; les simulations sur le DSP TMS320C31 donnent les mêmes résultats que sur SPARCStation Sun. En ce qui concerne les signaux expérimentaux, l'horizon temporel examiné est suffisamment long pour qu'on ait des renseignements valides sur la stabilité à long terme des algorithmes. Les simulations nous ont permis de vérifier en grande partie les résultats obtenus dans [Servière 89] sur des signaux sonar ainsi que les considérations générales sur les conditions de bon fonctionnement des algorithmes. Ainsi, le LMS se caractérise par une faible complexité de calculs et une bonne stabilité numérique à long terme. Sa vitesse de convergence dépend essentiellement du choix du pas d'adaptation (plus il est faible, plus l'algorithme converge lentement) ; le pas d'adaptation influe également sur la puissance de l'erreur résiduelle. Enfin, l'ordre du filtre n'influe pas sur le temps de convergence, mais sur l'erreur résiduelle. En ce qui concerne l'algorithme FTF, l'emploi d'un facteur d'oubli significatif (<1) conduit à une divergence rapide, même pour la version "7M+M". L'ordre du filtre influe sur la stabilité numérique de l'algorithme. Enfin, sa vitesse de convergence dépend essentiellement de la condition initiale sur les erreurs quadratiques de prédiction directe et rétrograde cumulées. Une valeur trop importante pour cette condition initiale ralentit l'algorithme. Nous avons mis en évidence la nécessité d'utiliser des techniques de stabilisation numérique pour une mise en oeuvre en horizon temporel infini. Néanmoins, ces techniques imposent des tests sur certaines variables internes de l'algorithme, ce qui à priori ne facilite pas une mise en oeuvre en temps réel sur DSP. Enfin, en ce qui concerne l'algorithme du Spectrofiltre adaptatif, nous avons vérifié sa robustesse par rapport à l'emploi d'un facteur d'oubli significatif, d'où la possibilité d'une mise en oeuvre en contexte non-stationnaire. L'augmentation du taux de recouvrement contribue à améliorer les résultats en termes de gain de traitement et d'erreur résiduelle.

Suite à l'étude théorique et aux simulations effectuées sur signaux expérimentaux et modélisés, il apparaît clairement que les algorithmes de soustraction de bruit LMS, FTF "7M+M" et Spectrofiltre adaptatif sont les mieux adaptés pour notre application. L'algorithme LMS temporel, dont les résultats sont comparables au FTF "7M+M" (voir tableau ci-dessous), constitue notamment une alternative très intéressante à faible coût de calculs (2M opérations par itération contre 8M pour le FTF). Le Spectrofiltre adaptatif donne les meilleurs résultats sur signaux expérimentaux et s'avère très prometteur pour le traitement de signaux à spectre de raies.

Table 4: Gains de traitement après convergence sur les signaux expérimentaux

Algorithme/Configuration	Excitation proche des capteurs	Excitation éloignée des capteurs
LMS	14 dB	6 dB
FTF "7M+M"	14 dB	7.3 dB
Spectrofiltre adaptatif	22 dB	12 dB

III.2. MISE EN OEUVRE SUR LE DSP TMS320C31

III.2.1. CHOIX DU SYSTEME DE DEVELOPPEMENT

La principale contrainte concernant le système de développement est d'atteindre le meilleur compromis possible entre coût et performances. Le premier problème réside dans le choix entre un DSP à virgule fixe et un DSP à virgule flottante pour notre application.

La différence entre ces deux types de DSP réside essentiellement dans la représentation numérique des données. Dans le format virgule flottante, le processeur gère lui-même la représentation interne des données ; l'utilisateur n'a pas besoin de connaître précisément l'ordre de grandeur des variables internes de l'algorithme. A l'inverse, le format virgule fixe contraint l'utilisateur à comprendre précisément la représentation interne des données qu'il manipule ; à l'utilisateur de gérer les problèmes qui se produisent lors de la manipulation de données d'ordres de grandeur différents. Ainsi, un calcul en virgule fixe se fait de la même manière qu'en virgule flottante, à l'exception que c'est à l'utilisateur de normaliser ses données et de gérer l'exposant. Il existe néanmoins des avantages à la virgule fixe, notamment une meilleure précision sur la représentation des nombres (pour un même nombre de bits) ; en effet, sur une représentation 32 bits en virgule flottante, 8 bits sont perdus pour l'exposant. La mantisse est donc codée sur 23 bits, ce qui signifie que le plus petit incrément entre deux nombres consécutifs dans l'intervalle $[-1 ; +1]$ est de $1.92E-07$. Sur une représentation 32 bits en virgule fixe, le plus petit incrément est de $4.65E-10$. Ainsi, il peut être plus intéressant d'utiliser un DSP virgule fixe pour une application dans laquelle les grandeurs internes peuvent facilement être normalisées entre -1 et +1. Par contre, si le DSP doit manipuler des données dont les ordres de grandeurs peuvent être très différents, alors la virgule fixe perd son avantage, car il faut shifter les mantisses avant de faire toute opération arithmétique.

Un autre aspect à prendre en compte dans le choix entre un DSP virgule fixe ou virgule flottante est le prix du DSP. En effet, un DSP virgule fixe peut coûter entre quelques US \$ et 80\$ pour les plus performants d'entre eux. Par contre, un DSP virgule flottante coûte en moyenne 150\$, à cause de la complexité de l'unité flottante.

Pour clore ce débat, on peut dire qu'il est très difficile de se livrer à une comparaison réelle entre ces deux types de DSP, car les architectures de pipeline, cache etc... sont très différentes. En ce qui concerne notre application de soustraction de bruit, les simulations ont permis de mettre en évidence une grande disparité dans les ordres de grandeur des variables internes des algorithmes, d'où notre préférence pour un DSP virgule flottante.

III.2.2. PRESENTATION DU DSP TMS320C31

Pour mettre en oeuvre les algorithmes sélectionnés précédemment, nous avons fait l'acquisition d'une carte équipée d'un processeur de traitement du signal Texas Instruments (TI) TMS320C31 à virgule flottante. Le DSP TMS320C31 est une version faible coût du TMS320C30, avec une puissance de calcul de 40 MFLOPS (Million Floating-Point Operations

Per Second), soit 20 MIPS (Million Instructions Per Second) avec un temps de cycle de 50 nsec. L'utilisation de la virgule flottante permet notamment de ne pas devoir se soucier des problèmes de précision et de dépassement de dynamique dans les algorithmes. En outre, la carte DSP utilisée (Loughborough Sound Images, LSI) possède un module d'entrée/sortie sur 2 voies, équipé de convertisseurs ADC/DAC 16 bits Delta-Sigma. La fréquence d'échantillonnage des convertisseurs est paramétrable de 10 KHz à 50 KHz au moyen d'un ASIC. Nous allons maintenant présenter rapidement les caractéristiques principales du C31 afin de rendre plus explicite la présentation de la mise en oeuvre des algorithmes au paragraphe suivant.

III.2.2.1. ASPECTS MEMOIRE

Le C31 possède 32K mots (32 bits) de SRAM (Static RAM), 2K mots (32 bits) de RAM interne et 2K mots (32 bits) de DPRAM (Dual-Port RAM) accessible au PC en écriture/lecture. La RAM interne autorise notamment deux accès par cycle. Le C31 a également un cache programme de 64 mots (32 bits), qui peut se révéler bien utile lors de l'exécution de boucles par exemple. En effet, dans ce cas les instructions sont stockées dans le cache lors de leur premier fetch ; elles seront ensuite fetchées à partir du cache au lieu de la mémoire externe, d'où un nombre de cycles moindre et une plus grande disponibilité des bus externes pour d'autres transferts de données (voir Figure III-7).

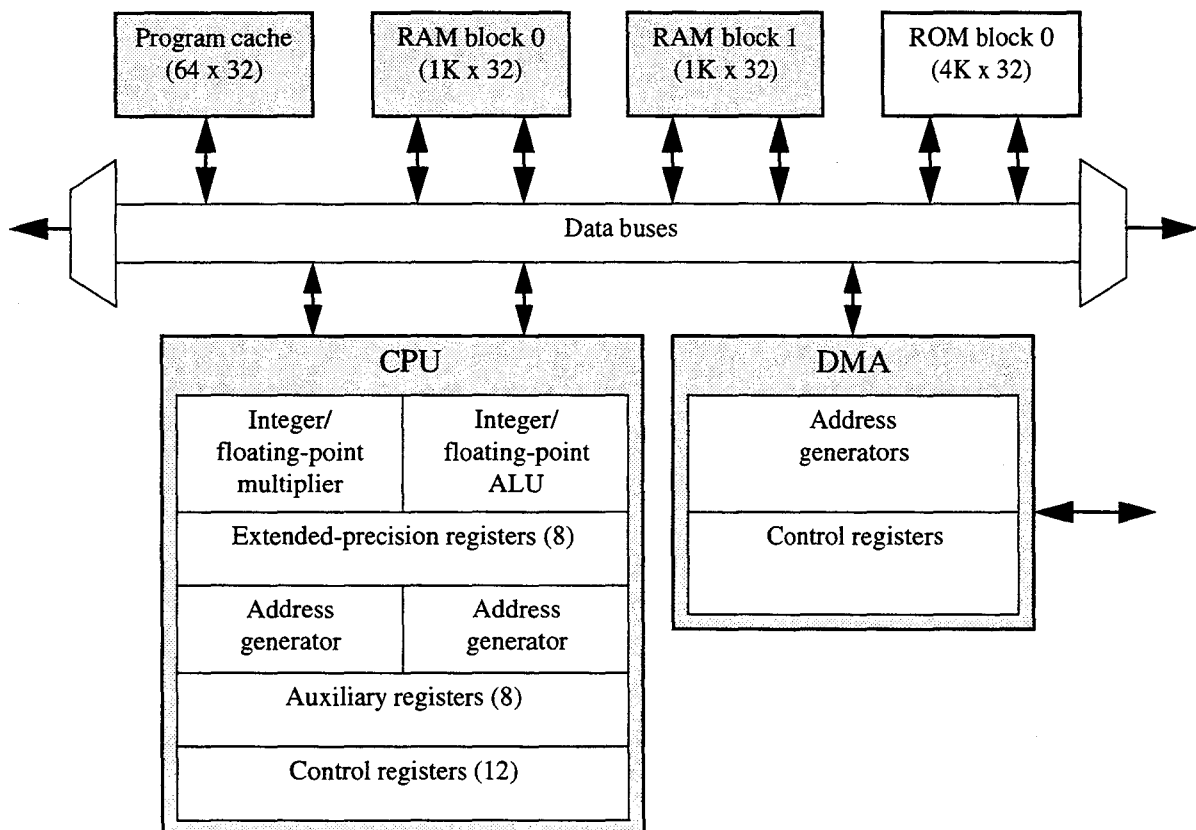


Figure III-7 : Architecture interne du TMS320C30/C31 (aperçu)

III.2.2.2. ARCHITECTURE INTERNE

L'architecture globale du C31 est de type Harvard dans le sens où il a des bus multiples pour accéder aux mots d'instructions, aux données, ou pour faire des transferts sur le DMA (Direct Memory Access). Néanmoins, il a également un parfum de Von Neumann car l'espace mémoire est unifié : il n'y a pas de séparation entre l'espace programme et l'espace données. De cette manière, l'utilisateur peut choisir où il met son code et ses données (voir Figure III-7).

III.2.2.3. CENTRAL PROCESSING UNIT (CPU)

La CPU (voir Figure III-7) se compose d'une ALU (Arithmetic & Logic Unit), d'un multiplieur hardware et d'un jeu de registres internes (appelé register file).

Ces registres internes se répartissent de la manière suivante :

- 8 registres à précision étendue (40 bits) : R0-7 ;
- 8 registres auxiliaires (32 bits) utilisés notamment pour la génération d'adresses : AR0-7 ;
- 12 registres de contrôle (32 bits) : registre d'état, pointeur de pile, registre d'interruption...

Les registres à précision étendue R0-R7 fonctionnent comme des accumulateurs et peuvent contenir des nombres entiers comme des nombres à virgule flottante. Quand ils sont utilisés pour des nombres flottants, les 8 bits de poids fort représentent l'exposant et les 32 bits de poids faible la mantisse. Les 8 registres auxiliaires AR0-AR7 peuvent fonctionner comme des pointeurs en adressage indirect, comme compteurs de boucle, ou comme registres généraux pour des opérations arithmétiques ou logiques. A ces registres auxiliaires sont associées deux ARAUs (Auxiliary Register Arithmetic Unit), capables de générer deux adresses mémoire en parallèle pour les instructions qui en ont besoin.

L'ALU peut réaliser des opérations flottantes (sur 40 bits), entières (sur 32 bits) ou logiques (sur 32 bits) sur des registres internes, des opérandes mémoire ou des valeurs immédiates. Le multiplieur hardware est très important, car toute application temps réel de traitement du signal repose sur une exécution rapide des opérations de multiplication. Le multiplieur peut effectuer des multiplications flottantes ou entières sur des mots de 32 bits, avec un résultat sur 40 bits dans un registre à précision étendue. Dans l'ALU comme le multiplieur, les résultats sont normalisés automatiquement. C'est la capacité à effectuer des multiplications/additions (ou multiplications/soustractions) en virgule flottante en un seul cycle qui donne au C31 sa performance de 40 MFLOPS.

III.2.2.4. DIRECT MEMORY ACCESS (DMA)

Le C31 possède un contrôleur DMA (voir Figure III-7), capable d'effectuer des lectures/écritures à n'importe quel emplacement en mémoire sans interférer avec la CPU. Le contrôleur DMA a ses propres bus d'adresses et de données, de manière à éviter tout conflit avec la CPU. De cette manière, il est possible d'interfacer le C31 avec des mémoires externes lentes ou

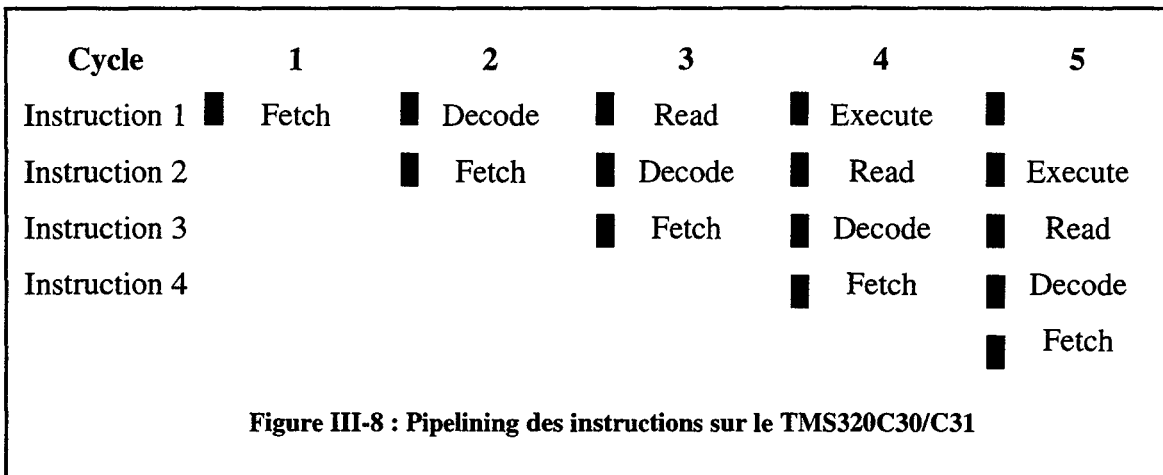
des périphériques (convertisseurs ADC, port série...) sans affecter les performances de la CPU.

III.2.2.5. PIPELINE

Les instructions du C31 sont contrôlées par 5 unités fonctionnelles, qui constituent le pipeline (voir Figure III-8) :

- Fetch Unit (F) : contrôle la mise à jour du Program Counter et fetch le mot d'instruction à partir de la mémoire ;
- Decode Unit (D) : décode le mot d'instruction et supervise la génération des adresses des opérandes ;
- Read Unit (R) : contrôle la lecture des opérandes en mémoire ;
- Execute Unit (E) : lit les opérandes dans le register file, exécute l'instruction, et stocke le résultat soit dans le register file, soit en mémoire ;
- DMA channel : lit/écrit en mémoire de manière concurrente avec les opérations de la CPU.

Chaque instruction passe dans les 4 premières unités fonctionnelles. Pour obtenir la puissance de calcul la plus élevée possible, il faut écrire le code de manière à ce que chacune des unités travaille en parallèle sur des instructions différentes (pipelining), ce qui revient à éviter les conflits de pipeline (voir Figure III-8). Néanmoins, le pipeline est protégé, ce qui signifie que des NOPs sont insérés automatiquement dès qu'un conflit se produit. L'utilisateur peut donc écrire son code sans se soucier à priori du pipeline.



III.2.2.6. JEU D'INSTRUCTIONS

Le jeu d'instructions du C31 est parfaitement bien adapté aux applications de traitement du signal. En particulier, les instructions parallèles supportent des modes d'adressage très intéressants. Elles permettent non seulement d'exécuter des multiplications/additions et des multiplications/soustractions en un seul cycle, mais également de charger deux registres en parallèle, ou encore d'effectuer une opération arithmétique (ou logique) en parallèle avec une écri-

ture en mémoire. L'exécution de multiplications/additions en parallèle permet d'atteindre la performance pic de 40 MFLOPS ; l'exécution d'une écriture en mémoire en parallèle avec une opération arithmétique permet également de masquer un cycle de transfert de données. Par exemple, l'instruction suivante additionne le contenu de la mémoire pointée par le registre d'adresse AR1 avec le registre R0 et stocke le résultat dans le registre R1. En parallèle, le contenu précédent de R1 est stocké à l'emplacement mémoire pointé par le registre AR3 :

```
ADDF  *AR1, R0, R1
II    STF   R1, *AR3
```

Les instructions de Delayed Branch (Branch retardé) fournissent un autre exemple de la flexibilité du jeu d'instructions. En effet, si on exécute un Branch normal, on perd 3 cycles car il faut flusher le pipeline (voir Figure III-8) ; l'astuce est alors d'utiliser un Delayed Branch, pour lequel les 3 instructions suivantes sont exécutées de toute façon. On récupère de cette manière les 3 cycles perdus.

III.2.2.7. MODES D'ADRESSAGE

Le C31 supporte plusieurs modes d'adressage qui permettent à l'utilisateur d'accéder les données. Les modes d'adressage de base sont les suivants :

- registre : l'opérande se trouve dans un registre de la CPU (register file).
- direct : l'adresse mémoire de l'opérande est générée en concaténant les 16 bits de poids faible du mot d'instruction avec les 8 bits du pointeur de page (le C31 a des bus d'adresses de 24 bits), il faut donc dans ce mode d'adressage que l'espace mémoire soit segmenté en pages de 64 Kmots chacune.
- indirect : l'adresse mémoire de l'opérande est contenue dans un registre auxiliaire ; ce registre peut être modifié par l'instruction (incrément/décément simple ou par un registre d'index). Il existe deux formes d'adressage indirect particulièrement utiles : l'adressage circulaire et l'adressage bit-reversed. Ce dernier est utilisé notamment dans les FFT pour compenser le fait que les données d'entrée présentées dans le bon ordre se retrouvent entrelacées en sortie. Quant à l'adressage circulaire, il permet de définir des buffers circulaires, particulièrement utiles pour le filtrage numérique. En effet, chaque nouvelle donnée écrite dans un buffer circulaire peut venir écraser la plus ancienne, ce qui permet de mettre en oeuvre très facilement une fenêtre glissante sur les données en temps réel. Dans l'adressage circulaire, le registre de contrôle BK définit la taille du buffer circulaire ; ensuite, on peut modifier circulairement tout pointeur initialisé dans ce buffer (incrément/décément).
- immédiat : ce mode d'adressage peut encoder directement un opérande de 16 bits pour des opérations arithmétiques, ou une adresse absolue de 24 bits pour un mot d'instruction.

III.2.2.8. OUTILS DE DEVELOPPEMENT

Les outils de développement fournis par TI permettent de générer un exécutable extrêmement optimisé sur le C31, directement à partir du code C. En effet, le compilateur C comporte différents niveaux d'optimisation permettant à un bon programme C d'approcher l'efficacité d'un programme écrit directement en assembleur. Néanmoins, pour les applications qui ont des contraintes temps réel importantes, il peut être nécessaire d'écrire du code assembleur (ce sera notre cas, comme expliqué dans la suite de ce chapitre). Dans ce cas, il faut isoler les parties critiques du code et les écrire en assembleur dans des fonctions appelables à partir du C, ce qui se fait assez facilement (le passage de paramètres du C vers l'assembleur par la pile se fait de manière assez intuitive).

L'assembleur génère un fichier objet à partir du code assembleur généré par le compilateur C ou écrit par l'utilisateur. Enfin, le linker permet de combiner plusieurs fichiers objet en un seul fichier exécutable. Il alloue les sections du code assembleur en mémoire suivant les directives de l'utilisateur.

Il existe un simulateur du C30/C31 permettant notamment de déboguer les programmes et de visualiser le pipeline. Nous avons utilisé le débogger MPCVIEW sous Windows 3.1 fourni par LSI avec la carte DSP.

III.2.3. MISE EN OEUVRE DES METHODES FREQUENTIELLES

Nous avons mis en oeuvre les algorithmes du spectrofiltre et du gradient fréquentiel sur le C31 en leur incorporant les fonctionnalités décrites au chapitre précédent (contrôle amont, contrôle aval, fenêtrage des données, contrainte du gradient). Nous avons également développé un interface graphique en C++ permettant au PC d'afficher graphiquement en temps réel les densités spectrales calculées par le DSP : autospectre de la voie signal, autospectre de la référence bruit, interspectre entre voie signal et voie référence, cohérence entre voie signal et voie référence. Nous allons maintenant détailler les principaux aspects de la programmation des algorithmes fréquentiels sur le C31.

III.2.3.1. MISE EN OEUVRE DES FFT/IFFT

Comme nous l'avons vu au chapitre précédent, les algorithmes du spectrofiltre et du gradient fréquentiel nécessitent le calcul des transformées de Fourier rapides directes (FFT) et inverses (IFFT). Ces transformées étant extrêmement coûteuses en temps de calcul, nous avons utilisé toutes les simplifications possibles et adapté directement le calcul aux paramètres des algorithmes. Ainsi, la plus grosse simplification vient du fait que les signaux à traiter sont réels. Sous cette condition, nous savons que le spectre est symétrique par rapport à la fréquence de Nyquist, ce qui permet de faire l'économie de la moitié des points calculés. Supposons que le résultat de la FFT soit complexe : $Z(k) = \text{Re}(k) + j.\text{Im}(k)$, et que la tranche de signal considérée soit de longueur N , nous savons alors que $\text{Re}(k)$ et $\text{Im}(k)$ satisfont aux relations suivantes :

- $\text{Re}(k) = \text{Re}(N - k)$, $k = 1, \dots, N/2-1$
- $\text{Im}(k) = -\text{Im}(N - k)$, $k = 1, \dots, N/2-1$
- $\text{Im}(0) = \text{Im}(N/2) = 0$.

En d'autres termes, la partie réelle de la transformée est symétrique par rapport à la fréquence de Nyquist, alors que la partie imaginaire est antisymétrique.

Le gain en calculs vient du fait que l'on n'a pas besoin de calculer tous les points. Mais ce gain entraîne aussi un gain en occupation mémoire. Nous avons mis en oeuvre l'algorithme de calcul de FFT de telle manière que, étant donnée une séquence de N nombres réels : $x(0)$, $x(1)$, ..., $x(N-1)$, nous obtenions le résultat de la FFT sous la forme suivante : $\text{Re}(0)$, $\text{Re}(1)$, ..., $\text{Re}(N/2)$, $\text{Im}(N/2-1)$, ..., $\text{Im}(1)$. $\text{Re}(k)$ et $\text{Im}(k)$ représentent respectivement la partie réelle et la partie imaginaire du nombre complexe $Z(k)$. De cette manière, le calcul d'une FFT sur N points utilise exactement N mots en mémoire (voir Figure III-9).

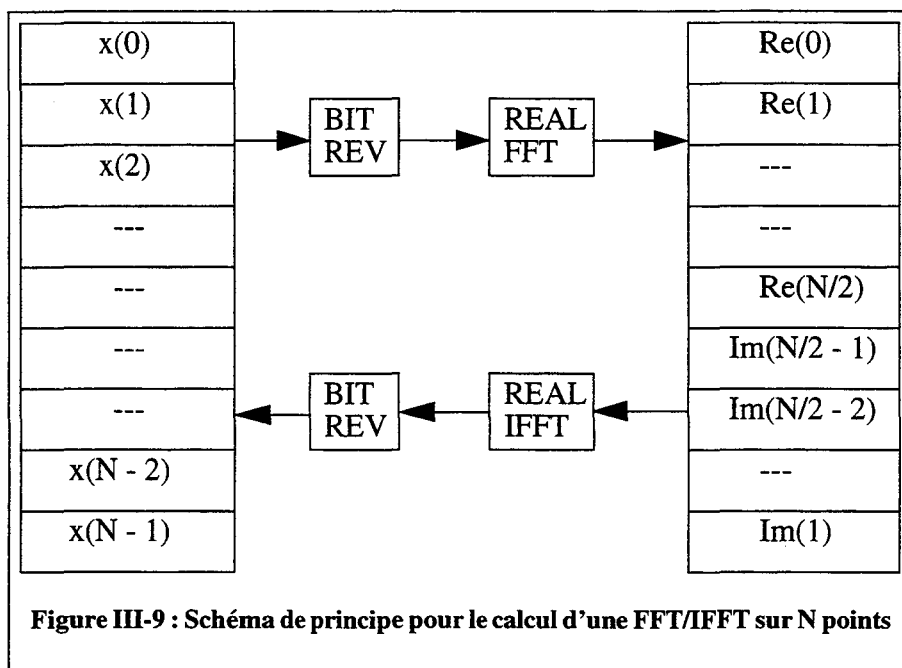


Figure III-9 : Schéma de principe pour le calcul d'une FFT/IFFT sur N points

Bien entendu, si le tableau complet $Z(k)$ est nécessaire, il suffit d'utiliser les relations suivantes:

$$Z(0) = \text{Re}(0) ;$$

$$Z(k) = \text{Re}(k) + j.\text{Im}(k), k = 1, \dots, N/2 - 1;$$

$$Z(N/2) = \text{Re}(N/2) ;$$

$$Z(k) = \text{Re}(N - k) - j.\text{Im}(N - k), k = N/2 + 1, \dots, N - 1.$$

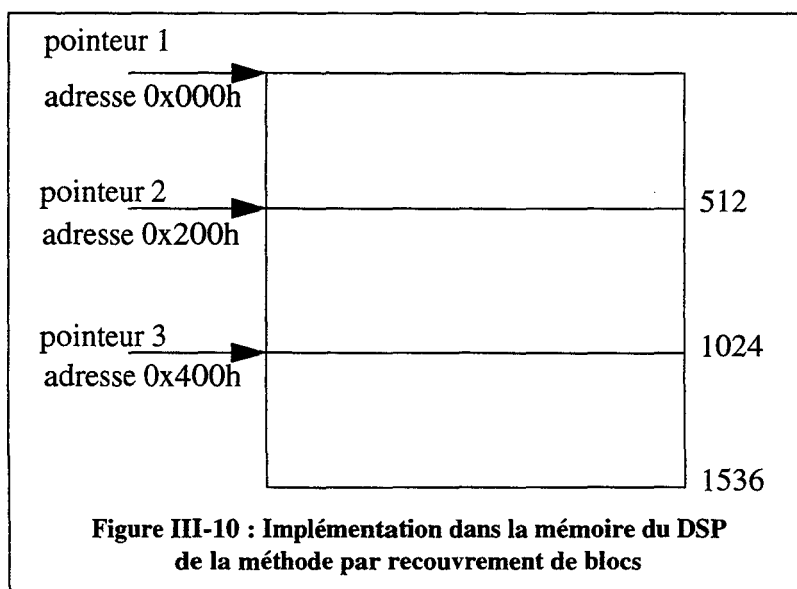
Néanmoins, nous avons adapté les calculs pour que la reconstruction du tableau complet ne soit pas nécessaire. En particulier, pour la transformation inverse, les données d'entrée sont présentées dans l'ordre de sortie de la transformation directe. On obtient ainsi le signal temporel dans l'ordre naturel après transformation inverse.

III.2.3.2. MISE EN OEUVRE DE LA METHODE PAR RECOUVREMENT DE BLOCS

Nous allons maintenant présenter la manière dont nous avons mis en oeuvre la méthode par recouvrement de blocs sur le C31 (voir Annexe III-5). Cette méthode a été programmée de manière à ce que la taille de chaque tranche (N) et le taux de recouvrement ($\alpha = D/N$) soient paramétrables (α peut ainsi être choisi quelconque entre 0 et 1, à condition que D soit entier). Evidemment, N doit être une puissance de 2 pour le calcul des FFT/IFFT ; elle doit être comprise entre 64 et 1024 (1024 étant la valeur maximale car on veut que les FFT/IFFT soient calculées en RAM interne pour des raisons de rapidité).

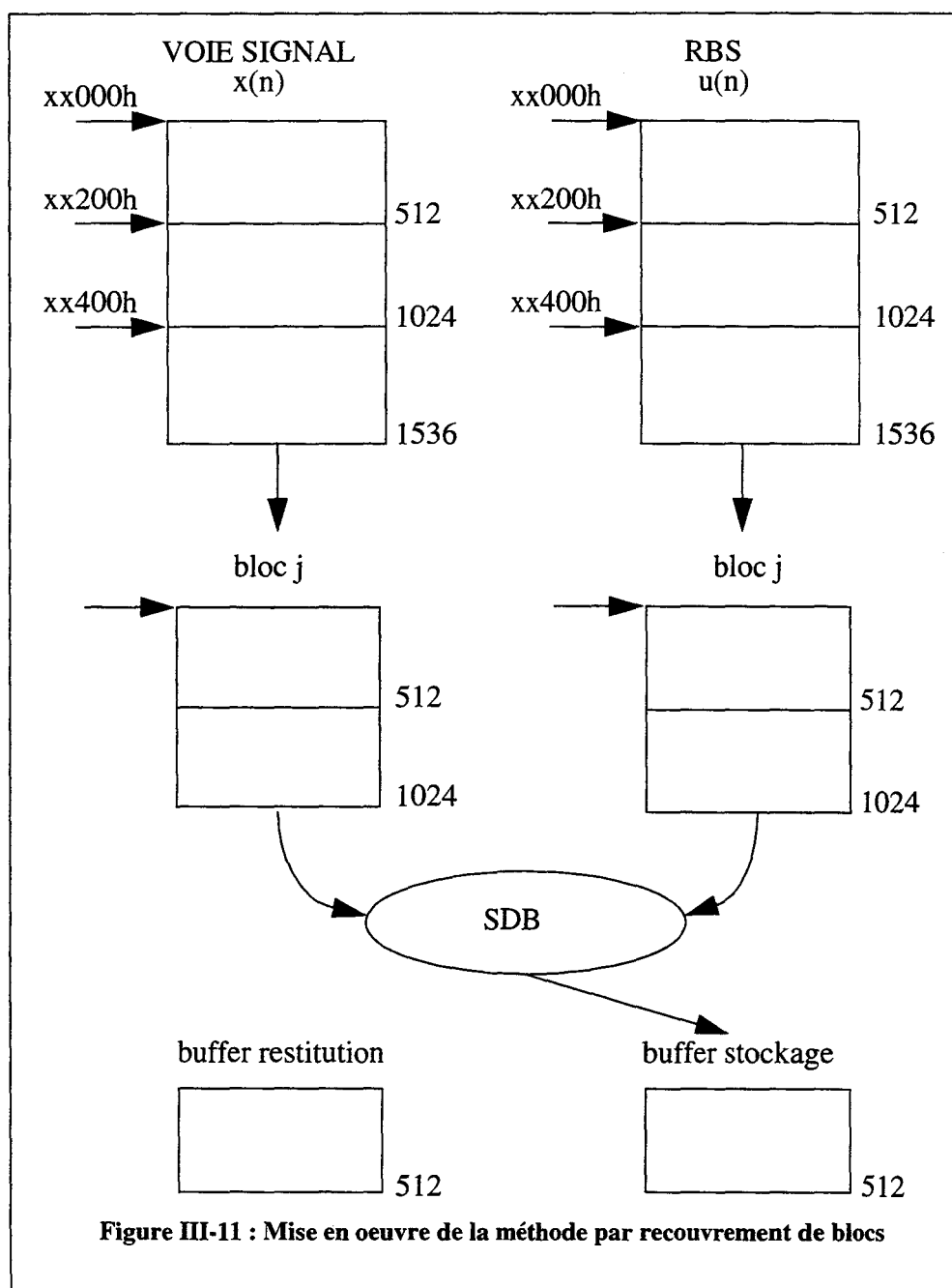
Les signaux analogiques issus de la voie signal et de la voie référence sont numérisés par le convertisseur d'entrée 16 bits au moyen d'une procédure d'interruption synchronisée sur l'échantillonnage des signaux (voir Annexe III-5). La fréquence d'échantillonnage des signaux est paramétrable entre 10 et 50 KHz. On adopte le principe suivant : pendant le traitement des tranches de rang j , le programme fait l'acquisition des $(N-D)$ nouveaux échantillons nécessaires à la constitution des tranches de rang $(j+1)$ et restitue simultanément les $(N-D)$ résultats correspondant aux tranches de rang $(j-1)$. Pour chaque voie, on a donc besoin d'un buffer d'entrée circulaire contenant $N + (N-D) = N(2-\alpha)$ points.

Un compteur initialisé à $(N-D)$ est décrémenté à chaque écriture d'un couple d'échantillons $\{x(n), u(n)\}$ dans leurs buffers d'entrée respectifs par la procédure d'interruption (Annexe III-5). A chaque écriture, la procédure d'interruption se charge également de restituer un échantillon du signal estimé. Lorsque le compteur passe à 0, la procédure d'interruption réveille le programme de soustraction de bruit. Cela signifie qu'une nouvelle tranche de signal est disponible et doit être traitée par l'algorithme (voir Figure III-10 : on a choisi $N = 1024$ et $\alpha = 0.5$ par souci de simplicité ; la tranche disponible est stockée de l'adresse $0x0000h$ à $0x0400h$ dans le buffer circulaire) ; le compteur est alors réinitialisé et l'écriture continue à l'adresse $0x0400h$ pendant que la tranche $0x0000h-0x0400h$ est prise en compte par le programme principal.



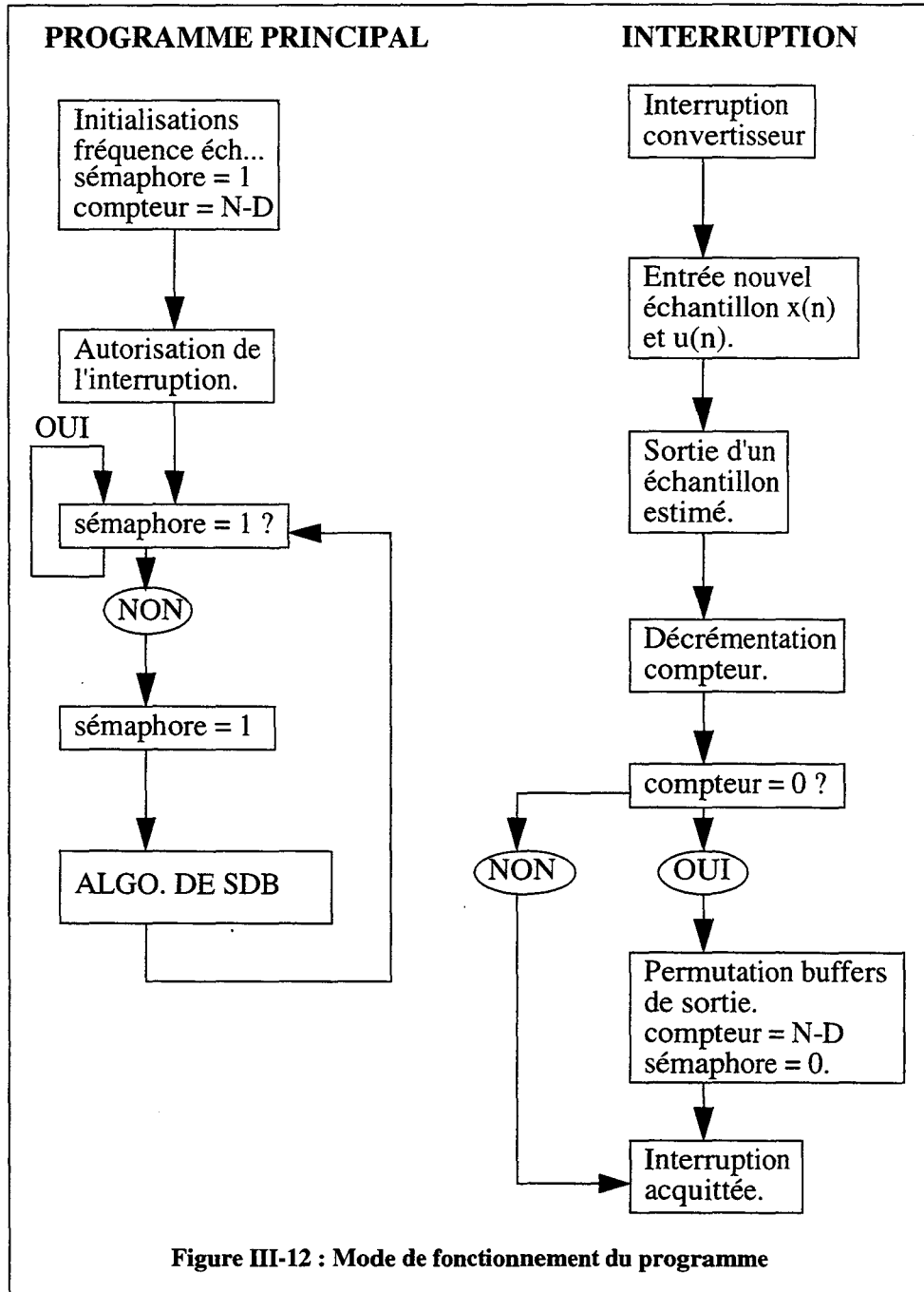
En première approximation, le temps dont dispose le programme pour traiter une tranche de N points correspond à $(N-D)$ interruptions : il est donc donné par $(N-D)/F_s$, où F_s est la fréquence d'échantillonnage. Si on veut que l'application tourne en temps réel, il faut impérativement que le programme réalise son traitement dans un temps inférieur à $(N-D)/F_s$.

Intéressons-nous maintenant au programme de soustraction de bruit (Annexe III-5) : réveillé par la procédure d'interruption, il traite les deux tranches courantes pour la voie signal et la voie référence et stocke le signal estimé dans un buffer de $(N-D)$ échantillons (voir Figure III-11 dans le cas particulier $N = 1024$ et $\alpha = 0.5$) :



Pendant ce temps, la procédure d'interruption continue l'écriture pour chaque voie dans le bloc libre du buffer circulaire d'entrée et envoie les résultats placés dans le buffer de restitu-

tion vers le Convertisseur Numérique Analogique (voir Annexe III-5). Il suffit alors de permuter le buffer de restitution et le buffer de stockage à chaque passage du compteur à 0. Cette méthode permet de prendre en compte un taux de recouvrement quelconque tout en assurant la continuité du signal de sortie. De cette manière, on peut utiliser une fréquence d'échantillonnage quelconque (sous réserve évidemment que le programme ait le temps de produire ses résultats en moins de $(N-D)$ interruptions) (voir Figure III-12).



Les algorithmes du spectrofiltre et du gradient fréquentiel ont été écrits selon ce modèle ; pour optimiser le temps de calcul et la taille du code, nous avons écrit les routines principales directement en assembleur callable à partir du C (voir Annexe III-5). Ceci permet de respecter plus facilement les contraintes temps réel et de garder une partie des ressources CPU

pour remonter des informations vers le PC à travers un interface graphique.

III.2.3.3. ASPECTS TEMPS REEL

Par le positionnement d'un flag (voir Annexe III-5), nous avons pu mesurer à l'oscilloscope la limite de fonctionnement temps réel de chaque algorithme. Nous quantifierons cette limite en terme de taux de recouvrement maximum. Nous avons également mesuré à l'oscilloscope la durée d'une FFT/IFFT sur un bloc élémentaire de N échantillons stocké en RAM interne :

Tableau 5 : Temps de calcul d'une FFT/IFFT élémentaire (données en RAM interne)

	FFT	IFFT
N = 128	0.15 msec	0.2 msec
N = 256	0.3 msec	0.4 msec
N = 512	0.7 msec	0.9 msec
N = 1024	1.5 msec	2 msec

Pour le spectrofiltre adaptatif et le gradient fréquentiel, nous avons comptabilisé le nombre de FFT/IFFT requises suivant que l'on met ou non en oeuvre la contrainte du gradient et suivant que l'on utilise l'interface graphique (en gras dans le tableau) ou non :

Tableau 6 : Nombre de FFT/IFFT requises pour chaque algorithme.

	Spectrofiltre adaptatif	Gradient fréquentiel	Spectrofiltre adaptatif	Gradient fréquentiel
Avec la contrainte du gradient	3 FFT 3 IFFT	4 FFT 5 IFFT	3 FFT 2 IFFT	4 FFT 3 IFFT
Sans la contrainte du gradient	2 FFT 3 IFFT	3 FFT 4 IFFT	2 FFT 1 IFFT	3 FFT 2 IFFT

De même, nous avons mesuré le taux de recouvrement maximum qui marque pour chaque algorithme la limite de fonctionnement temps réel :

Tableau 7 : Taux de recouvrement maximum pour chaque algorithme.

	Spectrofiltre adaptatif	Gradient fréquentiel	Spectrofiltre adaptatif	Gradient fréquentiel
Avec la contrainte du gradient	43.75 %	31.25 %	56.25 %	43.75 %
Sans la contrainte du gradient	50 %	40.625 %	62.5 %	50 %

Ces mesures ont été réalisées sur des signaux échantillonnés à 22 KHz avec la méthode par recouvrement de blocs. Elles sont complètement indépendantes de la taille des blocs. On constate sur ces mesures que la prise en compte de la contrainte du gradient est très pénalisante car elle entraîne une baisse de l'ordre de 8 % du taux de recouvrement maximum compatible avec un fonctionnement en temps réel des algorithmes. De même, l'interface graphique développé entre le PC et le DSP pour la visualisation en temps réel des données calculées par les algorithmes entraîne une chute de l'ordre de 12 % de ce taux critique de recouvrement.

Si on ne prend en compte ni la contrainte du gradient, ni l'interface graphique avec le PC, alors les taux de recouvrement critiques sont respectivement de 62.5 % pour le spectrofiltre adaptatif et de 50 % pour le gradient fréquentiel, ce qui constitue une performance tout à fait acceptable (un taux de recouvrement de 50 % représente le meilleur compromis performances-coût de calculs [Servière 89]).

Pour conclure sur les aspects temps réel, il convient de rappeler que les méthodes fréquentielles ne sont pas de vraies méthodes temps réel dans le sens où il y a toujours un délai entre l'acquisition d'un échantillon de la voie signal sur le Convertisseur Analogique Numérique et la restitution de l'échantillon correspondant du signal estimé sur le Convertisseur Numérique Analogique. Avec la méthode par recouvrement de blocs, ce délai correspond à $2N(1-\alpha)$ périodes d'échantillonnage, soit 46.5 msec pour $N=1024$ et $\alpha=0.5$. Pour notre application, ce délai de traitement ne constitue pas un problème. En effet, il aura la même valeur pour chaque doublet de capteurs utilisé et ne perturbera donc pas le traitement d'antenne qui suit la soustraction de bruit.

III.2.3.4. OCCUPATION MEMOIRE DES ALGORITHMES

Nous nous sommes fixés comme contrainte de n'utiliser que les ressources mémoire du DSP. Le PC ne sert qu'à télécharger le code dans le DSP et à fournir un écran de visualisation pour l'interface graphique que nous avons développé (voir Annexe III-5). De cette manière, on peut garantir un fonctionnement autonome du système. Les ressources mémoire exactes dont on dispose sur la carte DSP sont les suivantes :

- 31 Kmots de SRAM externe ;
- 1.94 Kmots de RAM interne ;
- 2 Kmots de DPRAM permettant l'échange de données avec le PC.

Nous allons quantifier la taille mémoire requise pour le fonctionnement du spectrofiltre et du gradient fréquentiel. Nous avons décidé d'affecter la RAM interne au stockage des données sur lesquelles sont effectués les calculs de FFT/IFFT ainsi qu'au stockage de la table des sinus requise par les routines de FFT/IFFT. La DPRAM sert uniquement à l'interface graphique entre le PC et le DSP (échange de données) ; elle ne sera donc pas prise en compte.

Dans le tableau suivant figurent les tailles respectives du code, des variables globales et de la pile pour le spectrofiltre et le gradient fréquentiel avec l'interface graphique mis en

oeuvre :

Tableau 8 : Tailles mémoire requises avec l'interface graphique mis en oeuvre

	Spectrofiltre adaptatif	Gradient fréquentiel
Code	2.8 Kmots	2.9 Kmots
Variables globales	22.5 Kmots	24.2 Kmots
Pile	1 Kmots	1 Kmots
Total	26.3 Kmots	28.1 Kmots
% d'utilisation RAM externe	80 %	86 %
% d'utilisation RAM interne	77 %	77 %
% d'utilisation RAM	80 %	85 %

Si on ne prend pas en compte l'interface graphique, on obtient les valeurs suivantes :

Tableau 9 : Tailles mémoire requises sans l'interface graphique

	Spectrofiltre adaptatif	Gradient fréquentiel
Code	2.6 Kmots	2.7 Kmots
Variables globales	15.7 Kmots	17.7 Kmots
Pile	1 Kmots	1 Kmots
Total	19.3 Kmots	21.4 Kmots
% d'utilisation RAM externe	57 %	64 %
% d'utilisation RAM interne	77 %	77 %
% d'utilisation RAM	59 %	65 %

Ces chiffres correspondent aux méthodes fréquentielles fonctionnant avec la taille de blocs maximale autorisée (1024 points). On constate que l'interface graphique consomme environ 20% de la mémoire disponible sur la carte, ce qui est extrêmement coûteux.

III.2.4. MISE EN OEUVRE DES METHODES TEMPORELLES

III.2.4.1. CONTRAINTE TEMPS REEL

Dans les simulations précédentes des algorithmes LMS et FTF sur signaux expérimentaux (voir paragraphe III.1), nous avons identifié un ordre optimal voisin de 100 pour le filtre adaptatif ; la fréquence d'échantillonnage des signaux étant de 25KHz, ceci revient à identifier en temps réel un filtre RIF dont le support temporel est de l'ordre de 5 msec. Sur le C31, nous pouvons utiliser une fréquence d'échantillonnage de 22KHz pour traiter les signaux acoustiques dans la bande 0-10KHz ; le temps de cycle du C31 étant de 50 nsec, le nombre de cycles machine disponibles pour le traitement de soustraction de bruit vaut approximativement 900. Il faut donc impérativement que l'algorithme de soustraction de bruit réalise son traitement en moins de 900 cycles pour tenir la contrainte temps réel. En prenant une marge de sécurité de 100 cycles (pour les sauvegardes de contexte, conflits de pipeline etc...), la complexité maximale admissible pour un algorithme de soustraction de bruit dans le domaine temporel est de 8M cycles machine par itération temporelle (où M représente l'ordre du filtre adaptatif). On constate que cette contrainte est impossible à tenir pour les algorithmes de la famille Recursive Least Squares : en effet, en exploitant au mieux le parallélisme du C31 et en évitant au maximum les conflits de pipeline, on aboutit pour le FTF à une complexité en 12M+200 cycles sur le C31, soit un ordre maximal de 60 compatible avec la contrainte temps réel. Qui plus est, nous avons vu que des techniques de stabilisation numérique sont nécessaires pour cet algorithme, ce qui va encore contribuer à diminuer l'ordre maximal du filtre compatible avec la contrainte temps réel. Par ailleurs, la validation des algorithmes en temps réel va se faire en cuve acoustique sur une plaque d'acier de taille relativement réduite (voir chapitre suivant) ; nous aurons donc forcément des échos sur les parois du bassin. Ces échos vont naturellement contribuer à allonger le support temporel du filtre à identifier, dans une proportion que nous ignorons a priori. Si l'on suppose que le support temporel du filtre est multiplié par deux à cause des échos, la complexité maximale compatible avec la contrainte temps réel pour un algorithme de soustraction de bruit passe à 4M cycles par itération temporelle. Sous cette contrainte, il est impossible de mettre en oeuvre les algorithmes RLS rapides (complexité minimale en 12M cycles C31 pour le FTF). Dans le domaine temporel, seuls les algorithmes de type LMS sont susceptibles de tenir la contrainte temps réel. Une alternative pourrait consister à faire le filtrage par blocs dans le domaine temporel (LMS-bloc ou RLS-bloc), mais le gain en coût de calculs n'est pas évident ; par ailleurs, les contraintes de convergence sont nettement plus sévères (ainsi pour le LMS-bloc, le pas d'adaptation μ maximal garantissant la convergence est nettement plus petit que pour le LMS standard [Haykin 96], d'où une convergence beaucoup plus lente de l'algorithme).

III.2.4.2. MISE EN OEUVRE DES ALGORITHMES LMS

III.2.4.2.1 Généralités

Nous avons dérivé au chapitre II l'algorithme LMS transverse dans sa version standard. Il existe cependant des variantes de cet algorithme [Papamichalis 90] ; on peut par exemple adopter une structure symétrique transversale au lieu de la simple structure transversale. Dans ce cas, la réponse impulsionnelle du filtre adaptatif est symétrique, ce qui permet d'avoir une phase linéaire et deux fois moins de multiplications que la structure transversale conventionnelle. Pour chaque structure, on peut avoir plusieurs variantes du LMS :

- LMS normalisé ;
- Algorithmes du signe ;
- Leaky LMS.

La structure symétrique transversale ne présente pas d'intérêt particulier pour notre application, d'autant que le gain en coût de calculs n'est pas très significatif. Dans la structure transversale, seules deux variantes du LMS nous intéressent : le LMS normalisé et le Leaky LMS. Les algorithmes du signe sont plus lents que le LMS standard et conduisent à une EQMR supérieure [Bellanger 89]. Ces algorithmes ont été développés pour réduire le nombre de multiplications du LMS standard. Paradoxalement, on ne tire aucun avantage quand on les met en oeuvre sur le C31 (complexité en $5M+16$ pour le Sign-Data, $3M+16$ pour le Sign-Error et $5M+16$ pour le Sign-Sign contre $3M+15$ pour le LMS standard, la comparaison étant faite dans la structure transversale). En fait, les coûts de calcul sont plus élevés à cause des instructions supplémentaires requises pour déterminer le signe des données.

Dans l'algorithme LMS normalisé, le pas d'adaptation μ est normalisé par l'estimation de la puissance instantanée de la voie référence. Ceci permet de réduire la dépendance de la vitesse de convergence de l'algorithme par rapport à la puissance de la référence bruit. Le coût de calcul supplémentaire est faible car une seule division est requise. On obtient pour cet algorithme un coût de calculs en $3M+47$ contre $3M+15$ pour le LMS standard dans la structure transversale. Cette variante du LMS présente un intérêt sur des signaux non-stationnaires.

L'algorithme leaky-LMS permet un rappel à zéro des coefficients du filtre adaptatif. Ceci permet d'éviter que les erreurs d'arrondi ne s'accumulent sur les coefficients du filtre à cause de la mise en oeuvre en précision finie. Par ailleurs, quand le signal sur la voie référence disparaît, le terme excitateur dans la mise à jour du filtre s'annule et les coefficients du filtre adaptatif sont immobilisés. Dans ce cas, il est préférable de les ramener à zéro, ce qui se réalise par un facteur de rappel dans l'équation de mise à jour du filtre. Ce facteur de rappel est très utile pour traiter les signaux non-stationnaires. En effet, sur de tels signaux, la valeur du facteur de rappel peut être choisie de manière à minimiser la puissance de l'erreur de sortie [Bellanger 89]. Par ailleurs, on peut arranger les instructions sur le C31 pour qu'il n'y ait aucun coût de calcul supplémentaire par rapport au LMS standard [Papamichalis 90].

III.2.4.2.2 Mise en oeuvre du LMS transverse

Comme nous l'avons vu précédemment, le C31 a deux bus d'adresses de données, ce qui permet de générer en même temps deux adresses mémoire. Par ailleurs, le multiplieur hardware et l'ALU étant séparés, le DSP peut effectuer une multiplication et une addition en un seul cycle. En associant ces possibilités dans une instruction RPTS (Repeat Single), la sortie du filtre adaptatif peut être calculée très rapidement comme suit :

```

    MPYF3    *AR0++(1)%, *AR1++(1)%, R1
    RPTS     N-2
    MPYF3    *AR0++(1)%, *AR1++(1)%, R1
II   ADDF3   R1, R2, R2
    ADDF3    R1, R2, R2

```

où les registres auxiliaires AR0 et AR1 pointent respectivement sur les coefficients du filtre adaptatif et sur le vecteur de régression de la voie référence. L'addition dans l'instruction parallèle somme les valeurs précédentes des registres R1 et R2. C'est pourquoi R1 est initialisé avec le premier produit avant l'instruction RPTS. On peut remarquer qu'il n'est pas nécessaire de déplacer les échantillons dans le vecteur de régression de la voie référence pour préparer l'itération suivante. En effet, l'adressage circulaire (%) permet de mettre en oeuvre une ligne à retard où l'échantillon le plus récent vient écraser le plus ancien. Au lieu de déplacer la donnée à l'emplacement mémoire suivant, le pointeur est mis à jour de manière à pointer sur l'emplacement mémoire précédent (voir Annexe III-6) ; on simule ainsi une fenêtre glissante. Par ailleurs, le C31 ayant un multiplieur 32 bits et le résultat des multiplications étant accumulé dans un registre à précision étendue sur 40 bits, nous n'avons aucun bruit d'arrondi après multiplication (les données du Convertisseur Analogique Numérique sont en effet codées sur 16 bits).

Bien que le C31 ne fournisse aucune instruction spécifique (à la différence du C54 par exemple) pour la mise à jour des coefficients du filtre adaptatif, il peut mettre à jour un coefficient en deux cycles :

```

    MPYF3    *AR0++(1)%, R7, R1
    LDI      N-3, RC
    RPTB     LMS
    MPYF3    *AR0++(1)%, R7, R1
II   ADDF3   *AR1, R1, R2
LMS  STF     R2, *AR1++(1)%
    MPYF3    *AR0, R7, R1
II   ADDF3   *AR1, R1, R2
    STF     R2, *AR1++(1)%
    ADDF3    *AR1, R1, R2
    STF     R2, *AR1++(1)%

```

De manière à mettre à jour le pointeur sur le vecteur de régression de la voie référence (AR0 dans l'exemple) pour la prochaine itération, une instruction est sortie de la boucle et modifiée en supprimant une incrémentation de AR0. Avec cette mise en oeuvre, le LMS transverse présente un coût de calcul en $3M+15$ cycles par itération temporelle. A ce coût de calculs, il con-

vient de rajouter 360 cycles pour la sauvegarde de contexte dans la procédure d'interruption et le stockage des données d'entrée (voir Annexe III-6), d'où un ordre maximal de 180 pour le filtre transversal.

III.3. CONCLUSION

Dans ce chapitre, nous avons présenté les résultats de soustraction de bruit obtenus en simulation par les algorithmes LMS, FTF et Spectrofiltre, d'une part sur les signaux expérimentaux provenant d'une campagne de mesures en lac, d'autre part sur les signaux temporels calculés à l'aide du modèle spécialement développé et présenté dans le chapitre I. Globalement, la soustraction de bruit fonctionne bien. En ce qui concerne les méthodes temporelles, le LMS présente des performances comparables au FTF, et offre donc une alternative très intéressante à faible coût de calculs. Le Spectrofiltre adaptatif offre les meilleures performances, et s'avère très prometteur pour le traitement de signaux à spectre de raies.

Nous avons ensuite justifié le choix du DSP TMS320C31 pour la mise en oeuvre en temps réel de ces algorithmes. Les principes et les choix d'optimisation pour la mise en oeuvre du LMS et du Spectrofiltre sont décrits en détail, le FTF ne respectant pas la contrainte temps réel imposée par notre application.

III.4. BIBLIOGRAPHIE

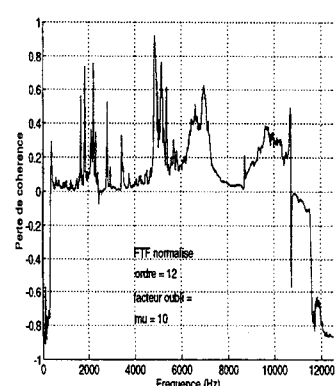
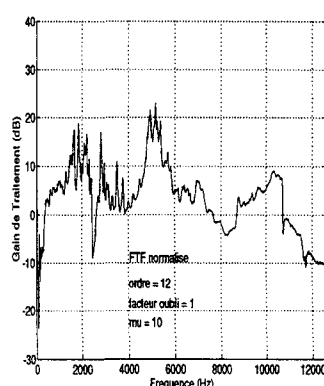
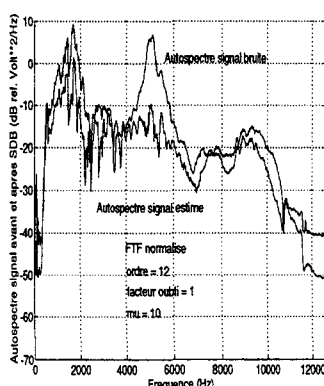
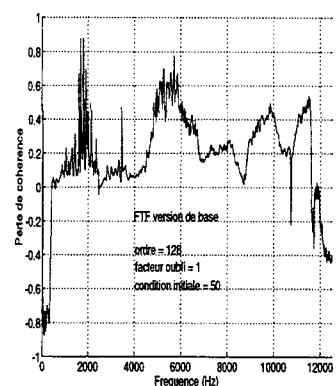
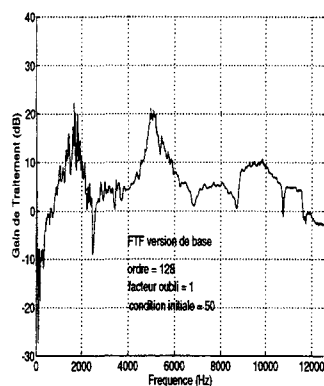
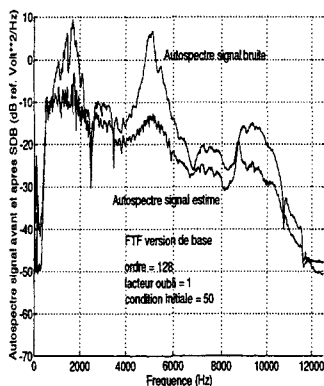
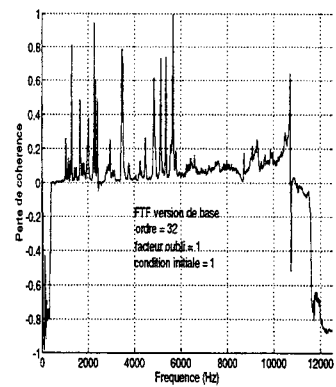
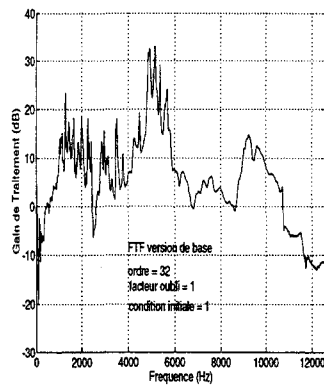
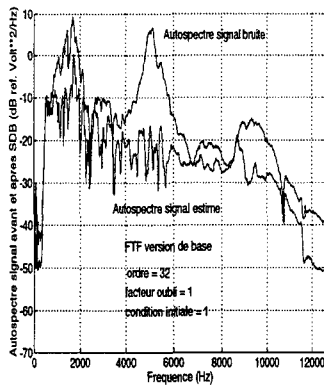
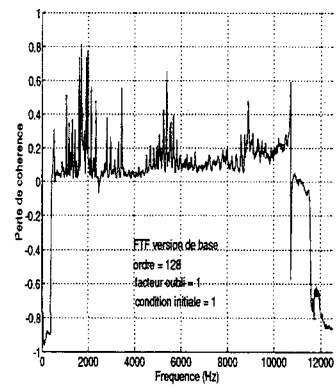
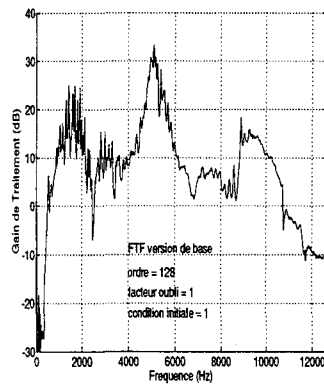
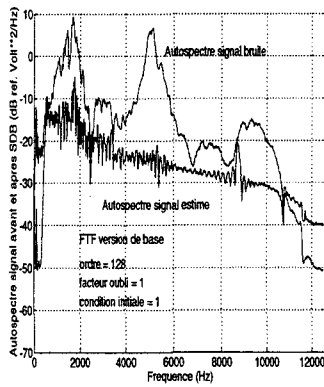
- [Bellanger 89] BELLANGER M., Analyse des Signaux et Filtrage Numérique Adaptatif, Collection Technique et Scientifique des Télécommunications, 1989.
- [Cioffi et Kailath 84] CIOFFI J.M., KAILATH T., Fast, Recursive Least Squares Transversal Filters for Adaptive Filtering, IEEE Trans. on ASSP, vol.32, p.304, Avril 84.
- [Cioffi et Kailath 85] CIOFFI J.M., KAILATH T., Windowed Fast Transversal Filters Adaptive Algorithms with Normalization, IEEE Trans. on ASSP, vol.33, p.607, Juin 85.
- [Lin 84] LIN D.W., On digital implementation of the Fast Kalman Algorithms, IEEE Transactions on ASSP, vol.32, p.998, Octobre 1984.
- [Papamichalis 90] PAPAMICHALIS P.E., Digital Signal Processing Applications with the TMS320 family, vol.3, 1990.
- [Servière 89] SERVIERE C., Eléments de comparaison entre différentes méthodes de soustraction de bruit, Thèse de doctorat INPG, Mai 1989.

Annexe III-1 : Validation du Fast Transversal Filter sur signaux expérimentaux

Autospectre signal avant et après traitement par le FTF

Gain de traitement (dB) par le Fast Transversal Filter (FTF)

Chute de cohérence par le Fast Transversal Filter (FTF)

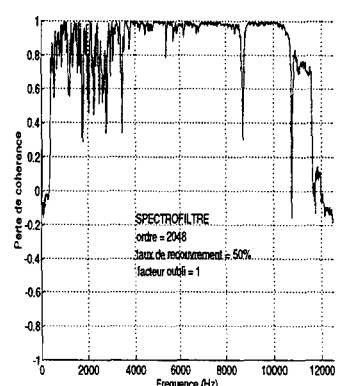
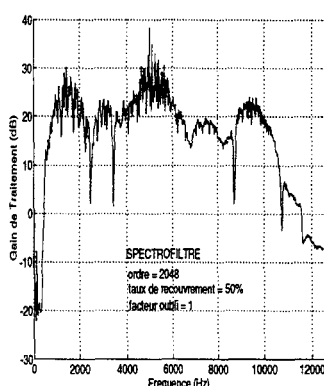
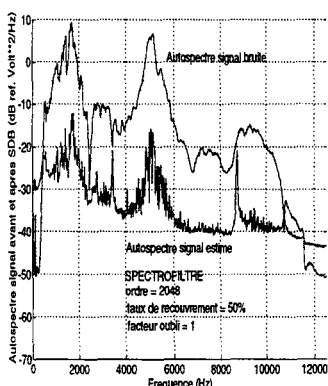
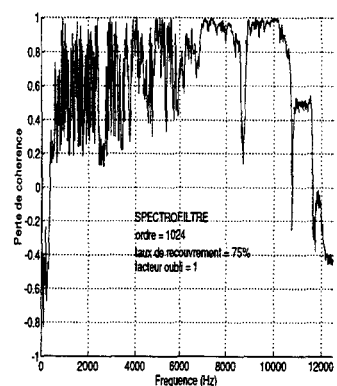
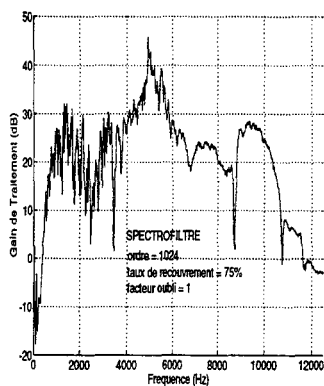
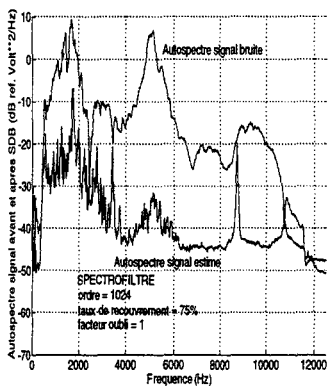
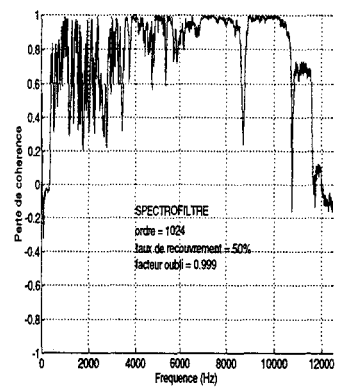
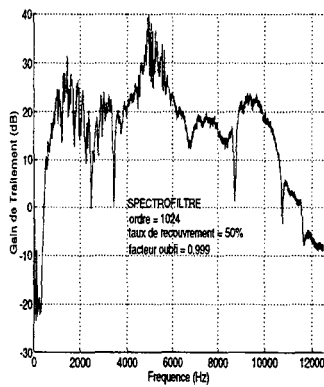
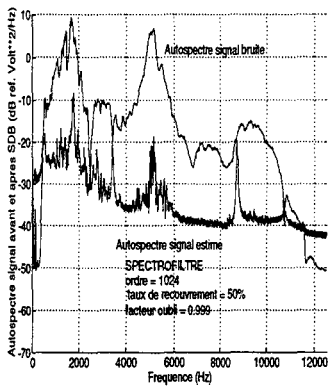
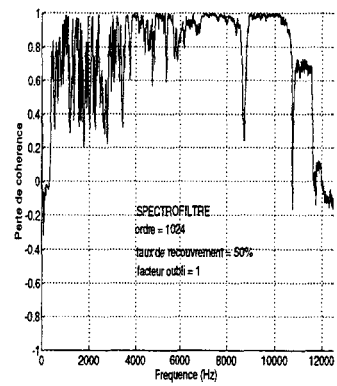
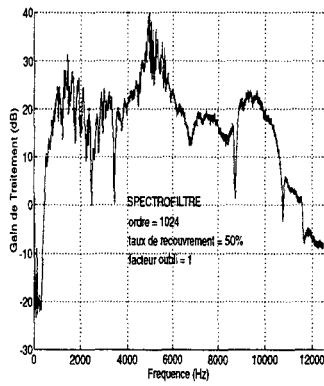
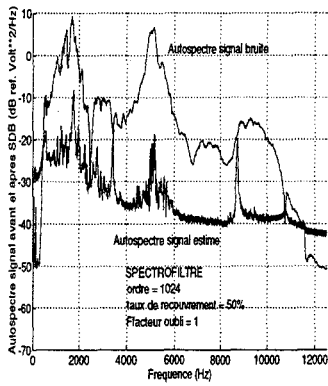


Annexe III-2 : Validation du Spectrofiltre adaptatif sur signaux expérimentaux

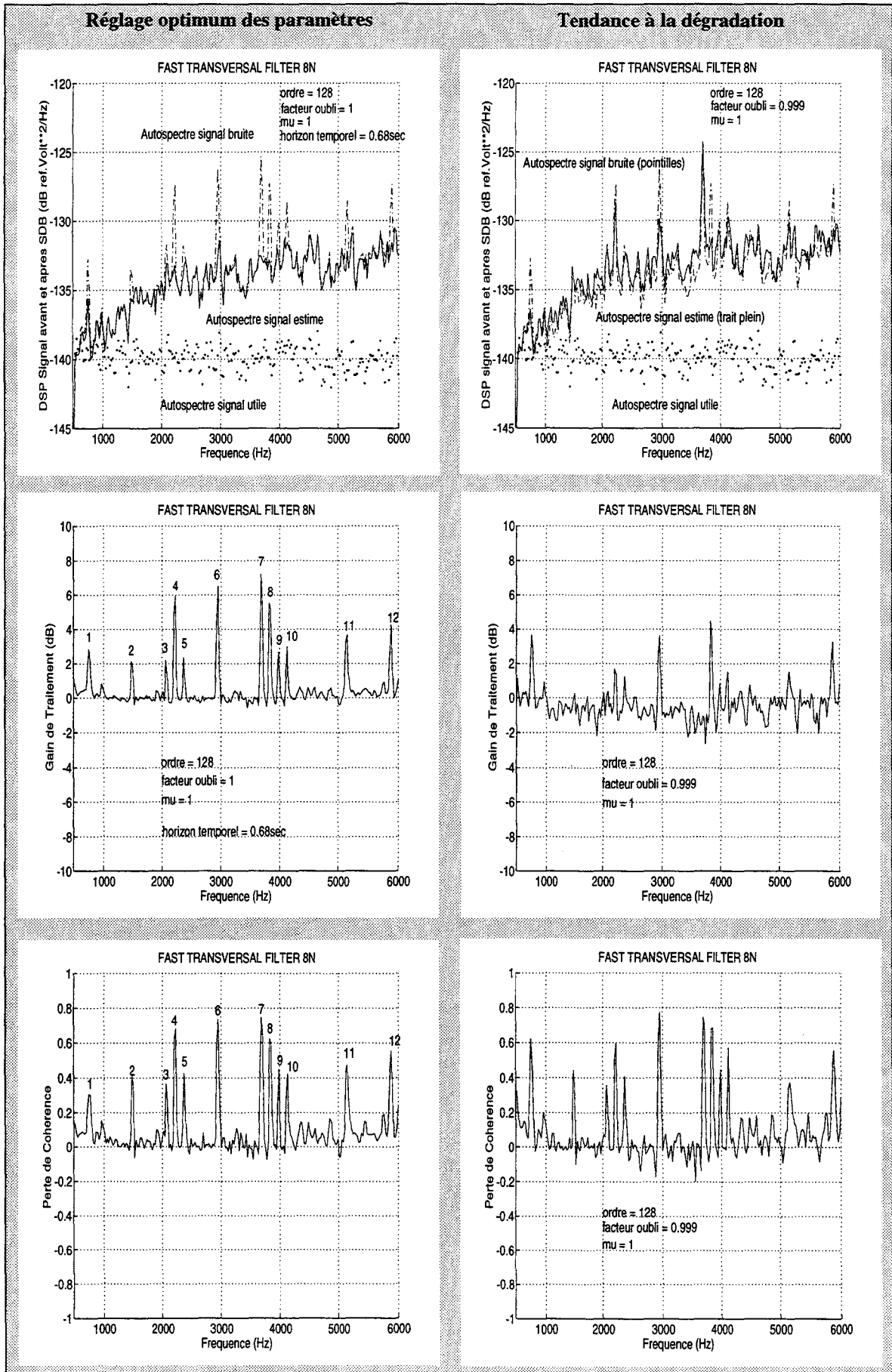
Autospectre signal avant et après traitement par le Spectrofiltre

Gain de traitement (dB) par le Spectrofiltre adaptatif

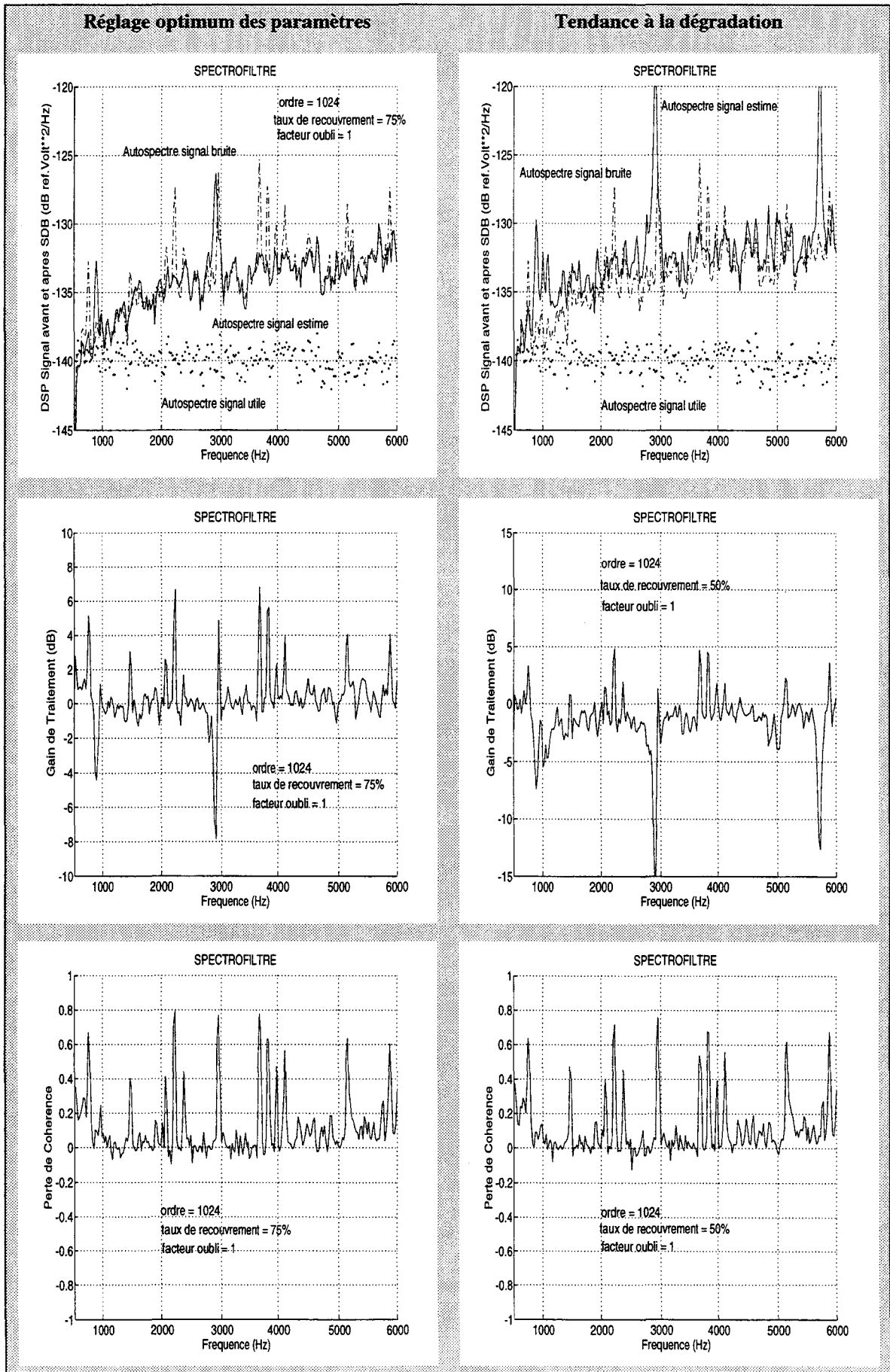
Chute de cohérence par le Spectrofiltre adaptatif



Annexe III-3 : Validation du FTF sur signaux simulés



Annexe III-4 : Validation du Spectrofiltre adaptatif sur signaux simulés



Annexe III-5 : Mise en oeuvre du spectrofiltre adaptatif sur C31 (extraits)

```

/*****
/* Programme : SFL.C */
/* Auteur : E. VERRIEST */
/* Dernière modification : 09/12/96 */
/* DESCRIPTION : */
/* Mise en oeuvre en temps réel du SPECTROFILTRE adaptatif sur le C31 avec la
/* méthode par recouvrement de blocs pour supprimer la convolution circulaire.
/* FONCTIONNALITES :
/* 1/ Fenêtrage de chaque bloc pour diminuer la corrélation entre les canaux fréquentiels.
/* 2/ Contrôle AMONT du traitement par seuillage sur la fonction de cohérence
/* estimée entre la voie signal (hydrophone) et la référence bruit seul (accéléromètre).
/* 3/ Contrôle AVAL sur chaque canal fréquentiel pour garantir que le traitement
/* ne rajoute pas de bruit de calcul
/* 4/ Prise en compte du support temporel du filtre à identifier en termes de nbre d'échantillons
/* (les derniers échantillons temporels du filtre sont mis à zéro s'ils n'ont pas de sens physique).
/* 5/ Utilisation d'un flag pour mesurer la limite de fonctionnement Temps Réel.
/* 6/ Interfaçage graphique sous WINDOWS 3.1 (Visual C++ : projet C31DEMO.MAK).
/* Visualisation TEMPS REEL des grandeurs suivantes :
/* - autospectre de la voie signal (dB) ;
/* - autospectre de la voie référence (dB) ;
/* - cohérence entre voie signal et voie référence ;
/* - gain fréquentiel du filtre adaptatif sans contrôle amont (dB) ;
/* - gain fréquentiel du filtre avec contrôle amont (dB) ;
/* - gain fréquentiel du filtre avec contrôle amont et réponse impulsionnelle limitée (dB) ;
/* - réponse impulsionnelle du filtre sans contrôle amont ;
/* - réponse impulsionnelle du filtre avec contrôle amont ;
/* - réponse impulsionnelle du filtre avec contrôle amont et contrainte sur la taille du support
/* temporel de la réponse impulsionnelle.
*****/
#include <stdlib.h>
#include <values.h>
#include <math.h>
#include «c31board.h»

#define True 1
#define False 0

/*****
/* PROTOTYPAGE DES PRIMITIVES LOCALES */
*****/
void set_c31_for_lsi_card ( void );
void set_amelia_a_for_crystal ( int f );
void c_int01 ( void );
extern float* circmov ();

asm(« .sect \».int01\»);
asm(« BR _c_int01»);
asm(« .text»);

/*****
/* VARIABLES GLOBALES */
*****/
const int F_samp = 22, /* fréquence d'échantillonnage (KHz) */
FFT_size = 1024; /* taille d'une tranche élémentaire */
const float ALPHA = 0.4375, /* taux de recouvrement */
OUBLI = 0.9; /* facteur d'oubli exponentiel */
const int FENETRAGE = 0; /* type de fenêtrage choisi */
const float BETA_AM = 10.0; /* facteur de sécurité sur le seuil utilisé dans le contrôle amont */
const int CONTRAINTE = 0, /* contrainte optionnelle sur le support temporel du filtre adaptatif */
ORDRE = 256; /* support temporel choisi à priori pour le filtre adaptatif */
const float BETA_AV = 0.0; /* facteur de sécurité sur la puissance estimée sur chaque canal dans le ctrl aval */

```

```

int          i, sem, counter, init, recov, in_size, half_fft_size, log2_size, upper, lower;
long         dummy;
float        hyd, acc, theta, tmp, seuil, flag;

extern float HYDin[2048], ACCin[2048], FFT[1024], SinTable[512];
float        HYDwork[1024], ACCwork[1024], out_1[1024], out_2[1024], filtre[1024], ispectre[1024],
aspectre_rbs[513], coherence[513], aspectre_sig[513], window[1024], iwindow[1024],
temp[1024], visu_1[1024], visu_2[1024], visu_3[1024], visu_4[1024], visu_5[1024], visu_6[1024];
float        *HYDin_ptr, *ACCin_ptr, *HYDslice_ptr, *ACCslice_ptr, *stock_ptr, *restit_ptr;

/*****
/* PROGRAMME PRINCIPAL */
*****/
void main ( void )
{
int          run_flag;
volatile unsigned long  *dualport, *size_max, *size_min;
unsigned long  value;

/* INITIALISATIONS */
run_flag = True;
dualport = (unsigned long *)DSP_FUNCTION_SELECT;
size_max = (unsigned long *)DSP_SIZE_MAX_SELECT;
size_min = (unsigned long *)DSP_SIZE_MIN_SELECT;

set_c31_for_lsi_card ( );
set_amelia_a_for_crystal ( F_samp );

theta = 2. * PI/FFT_size;
half_fft_size = (int) (FFT_size/2);
tmp = log10((float)FFT_size)/log10(2.);
lower = (int) floor(tmp); upper = (int) ceil(tmp);
if ((tmp-(float)lower) < 0.5) log2_size = lower;
if (((float)upper-tmp) < 0.5) log2_size = upper;
seuil = BETA_AM*(1. - OUBLI)/2.;      /* seuil minimum de cohérence pour appliquer la SDB (contrôle amont) */
in_size = (int) (FFT_size * (2.-ALPHA));
init = (int) (FFT_size * (1.-ALPHA)); recov = (int) (FFT_size * ALPHA);
counter = init; sem = 1;

for (i=0 ; i<512 ; i++) SinTable[i] = 0.;
for (i=0 ; i<half_fft_size ; i++) SinTable[i] = sin(i*theta);
for (i=0 ; i<1024 ; i++) {window[i] = 0.; iwindow[i] = 0.;}

/* CHOIX DU FENETRAGE */
/* Rectangulaire */
if (FENETRAGE == 0) {for (i=0 ; i<FFT_size ; i++) {window[i] = 1.; iwindow[i] = 1.;} }
/* Hanning */
else if (FENETRAGE == 1) {for (i=0 ; i<FFT_size ; i++) {window[i] = 0.5*(1. - cos(2.*PI*i/(FFT_size-1)));
iwindow[i] = 1./window[i];}}
/* Hamming */
else if (FENETRAGE == 2) {for (i=0 ; i<FFT_size ; i++) {window[i] = 0.54 - 0.46*cos(2.*PI*i/(FFT_size-1));
iwindow[i] = 1./window[i];}}
/* Blackman-Harris (-61 dB) */
else if (FENETRAGE == 3) { for (i=0 ; i<FFT_size ; i++)
{window[i] = 0.44959 - 0.49364*cos(2.*PI*i/(FFT_size-1)) + 0.05677*cos(4.*PI*i/(FFT_size-1));
iwindow[i] = 1./window[i];}}
/* Blackman-Harris (-67 dB) */
else if (FENETRAGE == 4) {for (i=0 ; i<FFT_size ; i++)
{window[i] = 0.42323 - 0.49755*cos(2.*PI*i/(FFT_size-1)) + 0.07922*cos(4.*PI*i/(FFT_size-1));
iwindow[i] = 1./window[i];}}
/* Blackman-Harris (-92 dB) */
else if (FENETRAGE == 5) {for (i=0 ; i<FFT_size ; i++)
{window[i] = 0.35875 - 0.48829*cos(2.*PI*i/(FFT_size-1)) + 0.14128*cos(4.*PI*i/(FFT_size-1)) - 0.01168*cos(6.*PI*i/

```

```

(FFT_size-1));
iwindow[i] = 1./window[i];}
else      exit(EXIT_FAILURE);

for (i=0 ; i<2048 ; i++) {HYDin[i] = 0.; ACCin[i] = 0.;}
for (i=0 ; i<1024 ; i++) {HYDwork[i] = 0.; ACCwork[i] = 0.; out_1[i] = 0.; out_2[i] = 0.; FFT[i] = 0.; filtre[i] = 0.;
ispectre[i] = 0.; temp[i] = 0.; visu_1[i] = 0.; visu_2[i] = 0.; visu_3[i] = 0.; visu_4[i] = 0.; visu_5[i] = 0.; visu_6[i] = 0.;}
for (i=0 ; i<513 ; i++) {aspectre_rbs[i] = 0.; aspectre_sig[i] = 0.; coherence[i] = 0.;}

HYDin_ptr = HYDin + recov; HYDslice_ptr = HYDin;
ACCin_ptr = ACCin + recov; ACCslice_ptr = ACCin;
restit_ptr = out_1; stock_ptr = out_2;
*size_max = (unsigned long)FFT_size;
*size_min = (unsigned long)(half_fft_size+1);

/* Autorisation des interruptions */
asm(« OR 0001h,IE»);
asm(« OR 2000h,ST»);

/* SOUSTRACTION DE BRUIT : SPECTROFILTRE ADAPTATIF */
while(run_flag)
{
  flag = -1.;
  while(sem);
  sem = 1;
  flag = 1.;

  /* Traitement de la nouvelle tranche de signal */
  ACCslice_ptr = circmov ( ACCslice_ptr, FFT ); /* FFT <- b_j(k) */
  fenetre ( window, FFT ); /* FFT <- b_j(k) * window */
  flag = 0.5;
  fftf_rl ( FFT_size, log2_size, FFT, SinTable ); /* FFT <- b_j(f) */
  flag = 1.;
  vecmov ( FFT, ACCwork, FFT_size ); /* ACCwork <- b_j(f) */
  vecmult ( FFT, filtre ); /* filtre <- H_j-1(f) * b_j(f) */
  /* opération de filtrage dans le domaine fréquentiel */
  HYDslice_ptr = circmov ( HYDslice_ptr, FFT ); /* FFT <- X_j(k) */
  fenetre ( window, FFT ); /* FFT <- X_j(k) * window */
  flag = 0.5;
  fftf_rl ( FFT_size, log2_size, FFT, SinTable ); /* FFT <- X_j(f) */
  flag = 1.;
  vecmov ( FFT, HYDwork, FFT_size ); /* HYDwork <- X_j(f) */

  /* Estimation du signal dans le domaine fréquentiel */
  /* Contrôle aval du traitement */
  vecsub ( FFT, filtre ); /* filtre <- S^j(f) avec S^j(f) = X_j(f) - H_j-1(f) * b_j(f) */
  vecmov ( filtre, temp, FFT_size ); /* temp <- S^j(f) */
  module ( temp ); /* temp <- |S^j(f)|^2 */
  module ( HYDwork ); /* HYDwork <- |X_j(f)|^2 */
  ctrl_av ( temp, HYDwork, filtre, FFT ); /* filtre <- S^j(f) avec mise en oeuvre du contrôle AVAL */

  /* Mise à jour des interspectres et autospectres */
  scalmult ( OUBLI, aspectre_rbs, half_fft_size+1 ); /* aspectre_rbs <- OUBLI * aspectre_rbs */
  scalmult ( OUBLI, aspectre_sig, half_fft_size+1 ); /* aspectre_sig <- OUBLI * aspectre_sig */
  scalmult ( OUBLI, ispectre, FFT_size ); /* ispectre <- OUBLI * ispectre */
  cormult ( FFT, ACCwork ); /* FFT <- X_j(f) * conjg(b_j(f)) */
  vecadd ( ispectre, FFT, FFT_size ); /* mise à jour de ispectre(f) */
  vecmov ( ispectre, temp, FFT_size ); /* temp <- ispectre(f) */
  module ( ACCwork ); /* ACCwork <- |b_j(f)|^2 */
  vecadd ( aspectre_rbs, ACCwork, half_fft_size+1 ); /* mise à jour de aspectre_rbs */
  vecadd ( aspectre_sig, HYDwork, half_fft_size+1 ); /* mise à jour de aspectre_sig */

  /* Estimation du signal dans le domaine temporel et mise à disposition des résultats pour la routine d'interruption */

```

```

vecmov ( filtre, FFT, FFT_size );          /* FFT <- S^j(f) */
flag = -0.5;
ifft_rl ( FFT_size, log2_size, FFT, SinTable ); /* FFT <- S^j(k) */
flag = 1.;
fenetre ( iwindow, FFT );                 /* FFT <- S^j(k) * iwindow */
vecmov ( FFT+recov, stock_ptr, init );     /* on ne conserve que les (FFT_size - recov) derniers points */

/* Estimation du gain fréquentiel du filtre adaptatif avec contrôle amont par seuillage sur la cohérence estimée */
/* et contrainte (optionnelle) sur la taille de la réponse impulsionnelle du filtre adaptatif */
vecdiv ( ispectre, aspectre_rbs, filtre ); /* filtre <- Hj(f) */
vecmov ( filtre, visu_1, FFT_size );      /* visu_1 <- Hj(f) gain fréquentiel du FDAF sans contrôle amont */
vecmov ( filtre, FFT, FFT_size );        /* FFT <- Hj(f) */
flag = -0.5;
ifft_rl ( FFT_size, log2_size, FFT, SinTable ); /* FFT <- Hj(k) */
flag = 1.;
vecmov ( FFT, visu_2, FFT_size );         /* visu_2 <- Hj(k) RI du FDAF sans contrôle amont */
module ( temp );                          /* temp <- |ispectre(f)|^2 */
mult(aspectre_rbs, aspectre_sig, FFT, half_fft_size+1); /* FFT <- aspectre_rbs * aspectre_sig */
divi ( temp, FFT, half_fft_size+1 );      /* FFT <- coh(f) cohérence entre voie signal et voie référence */
vecmov ( FFT, coherence, half_fft_size+1 ); /* coherence <- coh(f) */
ctrl_am ( FFT, filtre );                  /* filtre <- Hj(f) avec contrôle amont */
vecmov ( filtre, visu_3, FFT_size );      /* visu_3 <- Hj(f) avec ctrl amont */
vecmov ( filtre, FFT, FFT_size );        /* FFT <- Hj(f) avec contrôle amont */
flag = -0.5;
ifft_rl ( FFT_size, log2_size, FFT, SinTable ); /* FFT <- Hj(k) avec contrôle amont */
flag = 1.;
vecmov ( FFT, visu_4, FFT_size );         /* visu_4 <- Hj(k) RI du FDAF avec contrôle amont */
vecmov ( FFT, visu_6, FFT_size );         /* visu_6 <- Hj(k) RI du FDAF avec contrôle amont */

if (CONTRAINTE == 1)                      /* contrainte optionnelle sur le support temporel du FDAF */
{
zero ( FFT+ORDRE, (FFT_size-ORDRE) );     /* on force à 0 les derniers coefficients de la RI du FDAF */
vecmov ( FFT, visu_6, FFT_size );         /* visu_6 <- Hj(k) RI du FDAF limitée à ORDRE points */
flag = 0.5;
ffft_rl( FFT_size, log2_size, FFT, SinTable ); /* FFT <- Hj(f) avec RI limitée à ORDRE points */
flag = 1.;
vecmov ( FFT, filtre, FFT_size );         /* filtre <- Hj(f) avec RI limitée à ORDRE points */
}
vecmov( filtre, visu_5, FFT_size );       /* visu_5 <- Hj(f) avec RI limitée à ORDRE points */

/* Envoi des données en DPRAM pour le PC (environnement graphique) */
value = *dualport;
switch(value)
{
case 0: run_flag = False; break;
/* Fonction de Coherence estimée */
case 2 : vecmov ( coherence, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Autospectre estimé de la Voie Signal */
case 3 : vecmov ( aspectre_sig, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Autospectre estimé de la Voie Référence bruit seul */
case 4 : vecmov ( aspectre_rbs, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Gain fréquentiel du FDAF sans contrôle */
case 5: module ( visu_1 ); vecmov ( visu_1, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Gain fréquentiel du FDAF avec contrôle AMONT */
case 6: module ( visu_3 ); vecmov ( visu_3, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Gain fréquentiel du FDAF avec contrôle AMONT et support temporel limité à ORDRE points */
case 7: module ( visu_5 ); vecmov ( visu_5, DSP_BUFFER_START, half_fft_size+1 ); break;
/* Réponse Impulsionnelle du FDAF sans contrôle */
case 8: vecmov ( visu_2, DSP_BUFFER_START, FFT_size ); break;
/* Réponse Impulsionnelle du FDAF avec contrôle AMONT */
case 9: vecmov ( visu_4, DSP_BUFFER_START, FFT_size ); break;
/* Réponse impulsionnelle du FDAF avec contrôle AMONT et support temporel limité à ORDRE points */
case 10: vecmov ( visu_6, DSP_BUFFER_START, FFT_size ); break;

```

```

    default : break;
    }
}

/*****/
/* PRIMITIVE LOCALE : INITIALISATION DU DSP */
/*****/
void set_c31_for_lsi_card(void)
{
#define primary_bus_control *((long*)0x808064)
#define lsi_bus_control_value 0x0900
primary_bus_control = lsi_bus_control_value;
}

/*****/
/* PRIMITIVE LOCALE : ECHANTILLONNAGE DES SIGNAUX */
/*****/
void set_amelia_a_for_crystal ( int f )
{
#define A_Ch0_Input_Data *((long*)0x550002)
#define A_Ch0_Output_Data *((long*)0x550002)
#define A_Ch1_Input_Data *((long*)0x550006)
#define A_Ch1_Output_Data *((long*)0x550006)
#define A_Timer_1 *((long*)0x550005)
#define A_User_Control *((long*)0x550008)
#define A_AMELIA_Status *((long*)0x55000a)
#define A_AMELIA_Control *((long*)0x55000a)
#define A_Interrupt_Status *((long*)0x55000b)
#define A_Interrupt_Mask *((long*)0x55000b)
#define A_Configuration *((long*)0x55000f)

if (f==48) {
A_User_Control = 0xa8200000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==44) {
A_User_Control = 0xa9200000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==32) {
A_User_Control = 0xa8200000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==29) {
A_User_Control = 0xa9200000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==24) {
A_User_Control = 0xa8600000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==22) {
A_User_Control = 0xa9600000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000;A_Interrupt_Mask = 0x10000;}
else if (f==16) {
A_User_Control = 0xa8600000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==15) {
A_User_Control = 0xa9600000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==12) {
A_User_Control = 0xa8a00000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==11) {
A_User_Control = 0xa9a00000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else exit(EXIT_FAILURE);}
}

```



```

/*****
/* INTERRUPT SERVICE ROUTINE : ACQUISITION ET RESTITUTION DES DONNEES */
/*****
void c_int01 ( void )
{
dummy = A_Interrupt_Status;
hyd = -A_Ch0_Input_Data/1073741824.0; acc = -A_Ch1_Input_Data/1073741824.0;
A_Ch0_Output_Data = -(*restit_ptr++)*1073741824.0; A_Ch1_Output_Data = -flag*1073741824.0;

asm(«PUSH      AR0»);
asm(«PUSH      R1»);
asm(«PUSHF     R1»);
asm(«LDI       @_in_size, BK»);
asm(«LDI       @_HYDin_ptr, AR0»);
asm(«LDF       @_hyd, R1»);
asm(«STF       R1, *AR0++(1)%»);
asm(«STI       AR0, @_HYDin_ptr»);
asm(«LDI       @_ACCin_ptr, AR0»);
asm(«LDF       @_acc, R1»);
asm(«STF       R1, *AR0++(1)%»);
asm(«STI       AR0, @_ACCin_ptr»);
asm(«POPF     R1»);
asm(«POP      R1»);
asm(«POP      AR0»);
if (--counter == 0)
{
restit_ptr = stock_ptr;
stock_ptr = (stock_ptr==out_1)?out_2:out_1;
counter = init;
sem = 0;
}
asm(«AND      Offfeh,IF»);
return;
}

```

```

*****
* Fichier : circmov.asm
* Usage : ptr = circmov ( float *ptr, float *data )
* Auteur : E. VERRIEST
* Date : 19/06/96
* data[i] <- ptr[i], i = 0,..., FFT_size - 1
* ptr = ptr + circ(FFT_size - recov).
* ptr est l'adresse d'un élément d'un buffer circulaire de taille in_size (var. globale du C).
* Cette fonction copie FFT_size (var. globale du C) points vers le tableau data.
* ptr est incrémenté circulairement de (FFT_size - recov) positions pour la prochaine itération.
*
* Registres modifiés : AR0, AR1, R0, R1, RC, BK
* Résultat : placé dans R0
* Nbre de cycles : FFT_size + 25
*****

```

```

FP      .set      AR3
.global _circmov ; point d'entrée de la routine
.global _FFT_size ; variables globales du C
.global _recov
.global _in_size

.text
_circmov:
    PUSH      FP
    LDI       SP,FP
    LDI       *-FP(2), AR0 ; AR0 -> ptr[0]
    LDI       *-FP(3), AR1 ; AR1 -> data[0]
    LDI       @_FFT_size, RC ; RC (Repeat Counter) <- FFT_size

```

```

LDI      @_in_size, BK          ; taille du buffer circulaire

SUBI     @_recov, RC            ; RC <- (FFT_size - recov)
LDF      *AR0++(1)%, R1        ; R1 <- ptr[0], AR0 -> ptr[1](%)
SUBI     2, RC                  ; RC <- (FFT_size - recov - 2)
RPTB    LOOP1                  ; RPT (FFT_size - recov - 1) fois ; i = 1,..., (FFT_size - recov - 1)
LOOP1LDF *AR0++(1)%, R1        ; R1 <- ptr[i], AR0 -> ptr[i+1](%)
||      STF      R1, *AR1++(1)  ; data[i-1] <- ptr[i-1] ; AR1 -> data[i]
LDI     AR0, R0                 ; nouvelle valeur de ptr pour la prochaine itération
                                           ; R0 <- &ptr[FFT_size - recov](%)
LDI     @_recov, RC            ; RC <- recov
SUBI     1, RC
RPTB    LOOP2                  ; RPT (recov) fois ; i = (FFT_size-recov),..., (FFT_size - 1)
LOOP2LDF *AR0++(1)%, R1        ; R1 <- ptr[i], AR0 -> ptr[i+1](%)
||      STF      R1, *AR1++(1)  ; data[i-1] <- ptr[i-1] ; AR1 -> data[i]
STF     R1, *AR1               ; data[FFT_size-1] <- ptr[FFT_size-1]

POP     FP
RETS

.end

```

* Fichier : vecmult.asm

* Usage : vecmult (float *data0, float *data1)

* Auteur : E. VERRIEST

* Date : 20/06/96

*

* data0 et data1 sont des tableaux de taille FFT_size, au format : Re(0), Re(1), ..., Re(N/2), Im(N/2-1), ..., Im(1)

* Cette fonction réalise la multiplication complexe des tableaux : data1 <- data0 * data1, soit :

* data1.Re(i) <- data0.Re(i)*data1.Re(i) - data0.Im(i)*data1.Im(i)

* data1.Im(i) <- data1.Re(i)*data0.Im(i) + data1.Im(i)*data0.Re(i)

*

* Registres modifiés : AR0, AR1, AR2, AR4, R0, R1, R2, R3, RC

* Nbre de cycles : 3*FFT_size + 21

```

FP      .set      AR3

```

```

.global _vecmult          ; point d'entrée de la routine

```

```

.global _FFT_size

```

```

.global _half_fft_size

```

```

.text

```

```

_vecmult:

```

```

    PUSH     FP

```

```

    LDI     SP, FP

```

```

    PUSH     AR4

```

```

    LDI     *-FP(2), AR0          ; AR0 -> data0.Re(0)

```

```

    LDI     AR0, AR1

```

```

    ADDI    @_FFT_size, AR1

```

```

    SUBI    1, AR1                ; AR1 -> data0.Im(1)

```

```

    LDI     *-FP(3), AR2        ; AR2 -> data1.Re(0)

```

```

    LDI     AR2, AR4

```

```

    ADDI    @_FFT_size, AR4

```

```

    SUBI    1, AR4                ; AR4 -> data1.Im(1)

```

```

    LDI     @_half_fft_size, RC

```

```

    SUBI    2, RC                  ; RC <- FFT_size/2-2

```

```

    MPYF3   *AR0++(1), *AR2, R1   ; R1 <- data0.Re(0)*data1.Re(0), AR0 -> data0.Re(1)

```

```

    STF     R1, *AR2++(1)        ; data1.Re(0) <- R1, AR2 -> data1.Re(1)

```

```

RPTB    LOOP          ; RPT (FFT_size/2-1) fois ; i = 1, ..., FFT_size/2-1

```

```

    MPYF3   *AR0, *AR2, R1        ; R1 <- data0.Re(i)*data1.Re(i)

```

```

    MPYF3   *AR1, *AR4, R3        ; R3 <- data0.Im(i)*data1.Im(i)

```

```

    MPYF3   *AR2, *AR1--(1), R0   ; R0 <- data1.Re(i)*data0.Im(i), AR1 -> data0.Im(i+1)

```

```

|| SUBF3      R3, R1, R2      ; R2 <- data0.Re(i)*data1.Re(i) - data0.Im(i)*data1.Im(i)
MPYF3      *AR4, *AR0++(1), R1 ; R1 <- data1.Im(i)*data0.Re(i), AR0 -> data0.Re(i+1)
ADDF3      R1, R0, R3      ; R3 <- data1.Re(i)*data0.Im(i) + data1.Im(i)*data0.Re(i)
LOOP STF   R2, *AR2++(1)    ; data1.Re(i) <- R2, AR2 -> data1.Re(i+1)
|| STF     R3, *AR4--(1)    ; data1.Im(i) <- R3, AR4 -> data1.Im(i+1)

MPYF3      *AR0, *AR2, R1   ; R1 <- data0.Re(N/2)*data1.Re(N/2)
STF       R1, *AR2         ; data1.Re(N/2) <- R1

POP        AR4
POP        FP
RETS

.end

```

Annexe III-6 : Mise en oeuvre du LMS temporel sur C31 (extraits)

```

/*****
/* FICHER : TLMS.C
/* AUTEUR : E. VERRIEST
/* DATE : 20/01/97
/* Mise en oeuvre sur C31 du filtre adaptatif transversal avec algorithme LMS standard.
/* CONFIGURATION ENTREES/SORTIES :
/*
/* d(n) -----SUB-----> e(n)
/*
/* x(n) -----| AF |-----> y(n)
/*
/*
/* ALGORITHME :
/* k = 0, ..., N-1
/* y(n) = SUM (w(k)*x(n-k) )
/* e(n) = d(n) - y(n)
/* w(k) = w(k) + mu*e(n)*x(n-k), k=0,1,2,...,N-1
/* avec : N = ordre du filtre ;
/* mu = facteur de convergence.
/*
/* COMPLEXITE : 3N + 370 cycles/itération.
/* REMARQUES
/* 1) Le code, les variables globales et la pile sont en RAM interne.
/* 2) Cet algorithme nécessite la mise en oeuvre de 2 buffers circulaires en RAM. Un buffer
/* circulaire de taille R doit avoir au moins les K LSBs de son adresse de départ à zéro,
/* K étant le plus petit entier qui vérifie la condition 2^K > R. La taille maximale autorisée
/* pour chaque buffer circulaire est 511.
/* 3) La limite de fonctionnement temps réel à 22 KHz se trouve à N = 180 coefficients, ce
/* qui signifie une complexité en 3N + 370 cycles/itération temporelle.
*****/
#include <stdlib.h>
#define N 50 /* ordre du filtre adaptatif */

/* Variables globales */
const int ORDRE = 50, /* ordre du filtre adaptatif */
F_SAMP = 22; /* frequence echantillonnage (KHz) */
const float MU = 0.0001; /* facteur de convergence du LMS */
extern float W[N], data[N];
float *ptr = data, hyd, acc, out;
long dummy;

/* Prototypage des primitives locales */
void set_c31_for_lsi_card ( void );
void set_amelia_a_for_crystal ( int f );
void c_int01 ( void );

asm(« .sect \».int01\»);
asm(« BR _c_int01»);

```

```

asm(« .text»);

/*****
/* Programme principal */
/*****
void main ( void )
{
int i;

set_c31_for_lsi_card ( );
set_amelia_a_for_crystal ( F_SAMP );
for (i=0 ; i<N ; i++) { W[i] = 0.0 ; data[i] = 0.0;}          /* Initialisation algorithme LMS */

asm(« OR 0001h,IE»);                                       /* Les interruptions sont autorisées */
asm(« OR 2000h,ST»);

while(1) {}                                               /* Boucle infinie qui sera interrompue à chaque échantillonnage */
}

/*****
/* Routine pour configurer EICR (External Interface Control Register) */
/*****
void set_c31_for_lsi_card ( void )
{
#define primary_bus_control *((long*)0x808064)
#define lsi_bus_control_value 0x0900
primary_bus_control = lsi_bus_control_value;
}

/*****
/* Routine pour configurer interface analogique */
/*****
void set_amelia_a_for_crystal ( int f )
{
#define A_Ch0_Input_Data *((long*)0x550002)
#define A_Ch0_Output_Data *((long*)0x550002)
#define A_Ch1_Input_Data *((long*)0x550006)
#define A_Ch1_Output_Data *((long*)0x550006)
#define A_Timer_1 *((long*)0x550005)
#define A_User_Control *((long*)0x550008)
#define A_AMELIA_Status *((long*)0x55000a)
#define A_AMELIA_Control *((long*)0x55000a)
#define A_Interrupt_Status *((long*)0x55000b)
#define A_Interrupt_Mask *((long*)0x55000b)
#define A_Configuration *((long*)0x55000f)

if (f==48) {
A_User_Control = 0xa8200000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==44) {
A_User_Control = 0xa9200000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==32) {
A_User_Control = 0xa8200000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==29) {
A_User_Control = 0xa9200000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==24) {
A_User_Control = 0xa8600000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration =0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==22) {
A_User_Control = 0xa9600000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;

```

```
A_Configuration = 0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==16) {
A_User_Control = 0xa8600000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration = 0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==15) {
A_User_Control = 0xa9600000; A_AMELIA_Control = 0xb30000; A_AMELIA_Control = 0xf30000;
A_Configuration = 0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==12) {
A_User_Control = 0xa8a00000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration = 0xb3900000; A_Interrupt_Mask = 0x10000;}
else if (f==11) {
A_User_Control = 0xa9a00000; A_AMELIA_Control = 0xa30000; A_AMELIA_Control = 0xe30000;
A_Configuration = 0xb3900000; A_Interrupt_Mask = 0x10000;}
else
exit(EXIT_FAILURE);
}
```

```
/*
*****

```

```
/* Interrupt Service Routine */

```

```
*****

```

```
void c_int01 ( void )

```

```
{
dummy = A_Interrupt_Status;
hyd = -A_Ch0_Input_Data/1073741824.0; acc = -A_Ch1_Input_Data/1073741824.0;

```

```
/* Appel à la routine assembleur de Soustraction De Bruit : algorithme LMS temporel */

```

```
ct31 ( ORDRE, MU, hyd, acc, &ptr, W, &out );

```

```
A_Ch0_Output_Data = -out*1073741824.0;

```

```
asm(«AND Offfeh,IF»);

```

```
return;

```

```
}
```

```
*****

```

```
* CT31.ASM : TMS320C31 adaptive transversal filter with LMS C subroutine.

```

```
* USAGE : ct31 ( n, mu, hyd, acc, &xptr, &W[0], &out )

```

```
*   *-FP(2)  n      ordre du filtre
*   *-FP(3)  mu     pas d'adaptation
*   *-FP(4)  hyd    voie signal
*   *-FP(5)  acc    voie référence
*   *-FP(6)  &xptr  adresse de ptr
*   *-FP(7)  &W[0]  adresse des coefficients du filtre adaptatif
*   *-FP(8)  &out   adresse du signal après soustraction de bruit (signal estimé)

```

```
* REMARQUES :
```

```
* 1) data[] and W[] size have to align the boundary for the circular addressing mode.

```

```
* 2) the value of xptr will be updated for next iteration.

```

```
* COMPUTATIONAL COMPLEXITY : (3n + 34) DSP cycles

```

```
* REGISTERS USED : R0, R1, R2, R3, AR0, AR1, AR2, BK, RC.

```

```
*****

```

```
FP.set      AR3
.global _ct31 ; point d'entrée de la routine

```

```
.text

```

```
_ct31:

```

```
    PUSH      FP
    LDI       SP, FP
    LDI       *-FP(2), BK      ; BK <- n
    LDI       *-FP(2), R0     ; R0 <- n
    LDI       *-FP(6), AR1    ; AR1 <- &xptr
    LDI       *AR1, AR0      ; AR0 -> data[0]
    LDI       *-FP(7), AR1    ; AR1 -> W[0]
    SUBI      2, R0          ; R0 <- n-2
    LDF       *-FP(5), R3     ; R3 <- acc = x(n)
```

```

    STF      R3, *AR0          ; data[0] <- acc = x(n)
; Compute filter output y(n)
    LDF      0.0, R3          ; R3 <- 0.0
    MPYF3    *AR0++(1)%, *AR1++(1)%, R1 ; R1 <- data[0] * W[0]
                                ; AR0 -> data[1] = x(n-1); AR1 -> W[1]
    RPTS     R0
    MPYF3    *AR0++(1)%, *AR1++(1)%, R1 ; R1 <- data[i+1] * W[i+1]
                                ; AR0 -> data[i+2]; AR1 -> W[i+2]
||   ADDF3   R1, R3, R3      ; R3 <- R3 + data[i]*W[i]
                                ; end of the loop
                                ; R1 <- data[n-1] * W[n-1]
                                ; R3 <- sum(data[i]*W[i]), i = 0,...,(n-2)
                                ; AR0 -> data[0]; AR1 -> W[0]
                                ; R3 <- sum(data[i]*W[i]), i = 0,...,(n-1)
                                ; <=> R3 <- y(n)
    ADDF     R1, R3
    LDI      R0, RC          ; RC <- n-2 : initialize loop counter for later
    SUBI     1, RC          ; RC <- n-3
; Compute error signal e(n)
    SUBRF    *-FP(4), R3     ; R3 <- d(n)-y(n) = e(n)
    LDI      *-FP(8), AR2   ; AR2 -> out
    STF      R3, *AR2       ; out <- e(n)
; Update adaptive filter weights
    MPYF     *-FP(3), R3     ; R3 <- mu * e(n)
    MPYF3    *AR0++(1)%, R3, R1 ; R1 <- mu * e(n) * data[0]
                                ; AR0 -> data[1] = x(n-1)
    RPTB     LMS
    MPYF3    *AR0++(1)%, R3, R1 ; R1 <- mu * e(n) * data[i+1]
                                ; AR0 -> data[i+2]
||   ADDF3   *AR1, R1, R2    ; R2 <- W[i] + mu * e(n) * data[i]
LMS  STF     R2, *AR1++(1)%  ; update W[i]
                                ; AR1 -> W[i+1]
                                ; end of the loop
                                ; R1 <- mu * e(n) * data[n-2]
                                ; AR0 -> data[n-1]
                                ; AR1 -> W[n-2]
||   MPYF3    *AR0, R3, R1    ; R1 <- mu * e(n) * data[n-1]
    ADDF3    *AR1, R1, R2    ; R2 <- W[n-2] + mu*e(n)*data[n-2]
    STF      R2, *AR1++(1)%  ; update W[n-2]
                                ; AR1 -> W[n-1]
    ADDF3    *AR1, R1, R2    ; R2 <- W[n-1] + mu*e(n)*data[n-1]
    STF      R2, *AR1++(1)%  ; update W[n-1]
                                ; AR1 -> W[0]
    LDI      *-FP(6), AR2   ; AR2 <- &xptr
    STI      AR0, *AR2      ; update xptr for next iteration

    POP     FP
    RETS

.end

```

IV

VALIDATION TEMPS REEL EN BASSIN D'ESSAIS ACOUSTIQUES

IV. VALIDATION TEMPS REEL EN BASSIN D'ESSAIS

IV.1. INTRODUCTION

Dans le chapitre II, nous avons présélectionné trois algorithmes de soustraction de bruit sur la base de la faisabilité de leur mise en oeuvre en temps réel sur un DSP standard. Il s'agit des algorithmes LMS et FTF dans le domaine temporel, et du spectrofiltre adaptatif dans le domaine fréquentiel. Dans le chapitre III, nous avons présenté les résultats obtenus en soustraction de bruit propre par ces algorithmes sur des signaux expérimentaux et simulés. Nous avons ensuite justifié le choix du DSP TMS320C31 en tant que plate-forme pour la mise en oeuvre en temps réel des algorithmes de soustraction de bruit. Comme nous l'avions prévu au chapitre II, l'algorithme FTF s'avère impossible à mettre en oeuvre en temps réel sur le C31 : la complexité arithmétique des algorithmes (nombre de multiplications par échantillon de sortie) s'avère être une estimation trop optimiste de leur complexité réelle dans le cadre d'une mise en oeuvre sur DSP. Seuls les algorithmes LMS et spectrofiltre sont finalement capables de respecter la contrainte de fonctionnement en temps réel. Leur mise en oeuvre sur le C31 est donc présentée dans le chapitre III.

Dans ce dernier chapitre, nous allons présenter les résultats obtenus en soustraction de bruit propre sur une structure mécanique par les algorithmes LMS et spectrofiltre, mis en oeuvre en temps réel sur le C31. Deux validations successives des algorithmes ont eu lieu en bassin d'essais acoustiques, sur la même structure mécanique. La première validation nous a permis tout d'abord de montrer qu'il était possible de faire l'acquisition de signaux représentatifs de la réalité physique avec les moyens matériels à notre disposition. Elle nous a également permis de mettre en évidence certaines modifications algorithmiques à apporter en vue d'une amélioration des résultats de soustraction de bruit. Nous verrons que la seconde validation, menée sur la même structure mécanique, donne des résultats tout à fait satisfaisants en soustraction de bruit propre.

IV.2. PREMIERE VALIDATION EN BASSIN D'ESSAIS

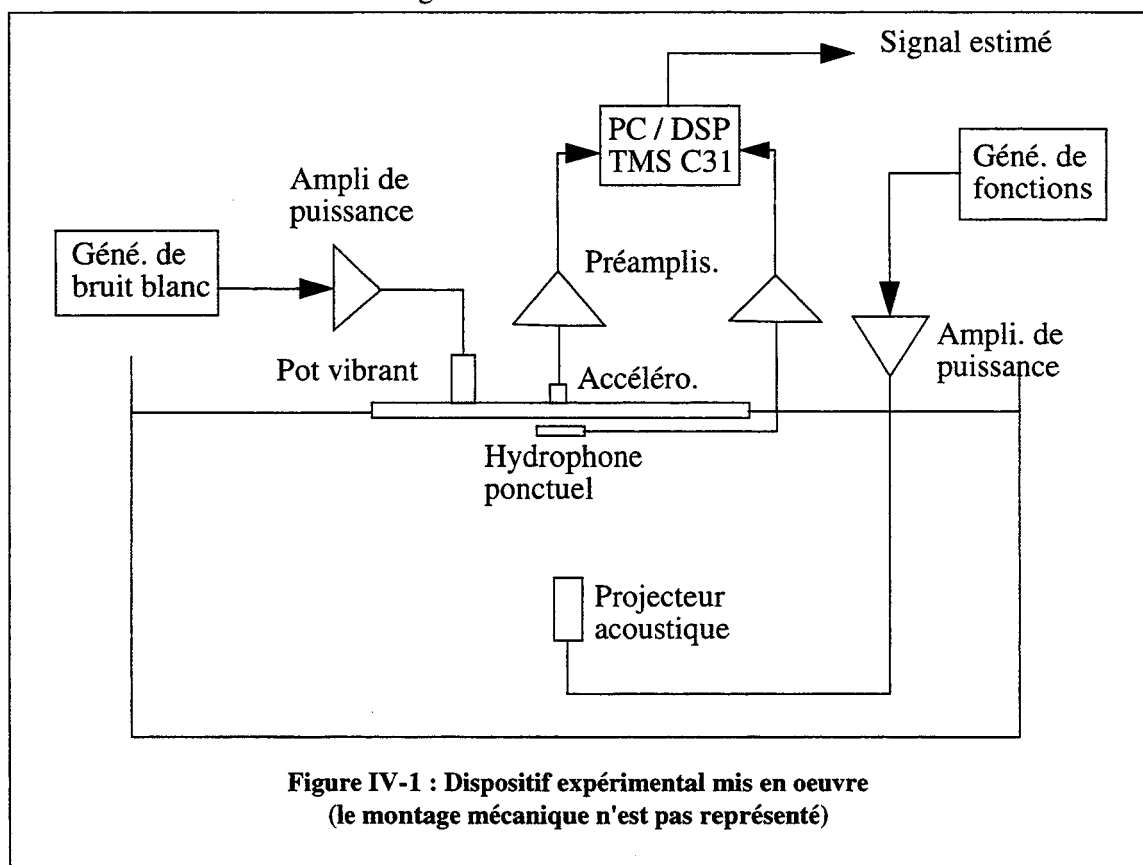
IV.2.1. DISPOSITIF EXPERIMENTAL

Afin de valider les algorithmes LMS et spectrofiltre/gradient fréquentiel en temps réel, nous avons mis en oeuvre une expérimentation spécifique en cuve acoustique dans les locaux de l'ISEN. Cette expérimentation fait intervenir une plaque d'acier plane de dimensions 2m. x 1m. x 1cm., mise en vibration par un dispositif d'excitation ponctuel de type pot vibrant, qui injecte le bruit propre dans la structure. On s'est également muni d'une source acoustique extérieure (projecteur immergé), qui émet le signal utile à détecter. La plaque est placée à la surface de l'eau, de manière à simuler l'interface eau-acier-air. Elle est équipée d'un doublet de capteurs hydrophone-accéléromètre disposés en vis-à-vis. L'hydrophone est découplé mécaniquement de la

plaque par une rondelle de caoutchouc de 5mm. d'épaisseur. L'objectif est de réaliser une soustraction de bruit propre en temps réel à partir de ce doublet de capteurs (voir Figure IV-1).

Dans cette première expérimentation, le seul projecteur que nous avons à notre disposition était un Massa TR1025C, dont la fréquence de résonance est voisine de 8KHz. Le signal injecté sur le projecteur acoustique est un burst sinusoïdal de fréquence 8KHz, afin de générer un spectre de signal utile large bande. Le principal problème est de superposer les supports fréquentiels du bruiteur et du signal utile. Pour cela, nous avons dû exciter le pot vibrant avec un bruit blanc stationnaire dans la bande 0-20KHz. Le bruit mécanique se manifestant essentiellement en basses fréquences (dans la bande 0-6KHz), nous avons dû faire passer le bruit blanc injecté sur le pot vibrant dans un filtre passe-haut avec une fréquence de coupure basse de 8KHz. Ainsi, la plaque est mise en vibration suivant ses modes propres et les supports fréquentiels du bruiteur et du signal utile sont les mêmes.

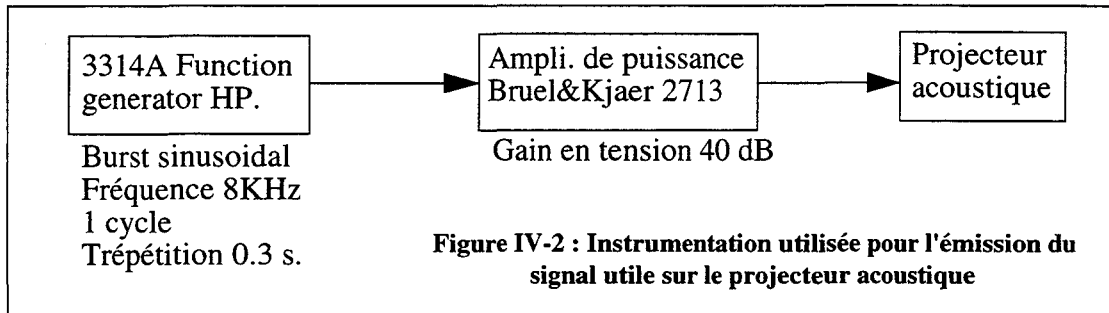
L'impossibilité de faire passer du signal utile sur le projecteur en-dessous de 5KHz nous contraint à travailler dans la bande 0-20KHz, et donc à utiliser une fréquence d'échantillonnage voisine de 40KHz pour les algorithmes de soustraction de bruit, afin de respecter le théorème de Shannon à l'échantillonnage.



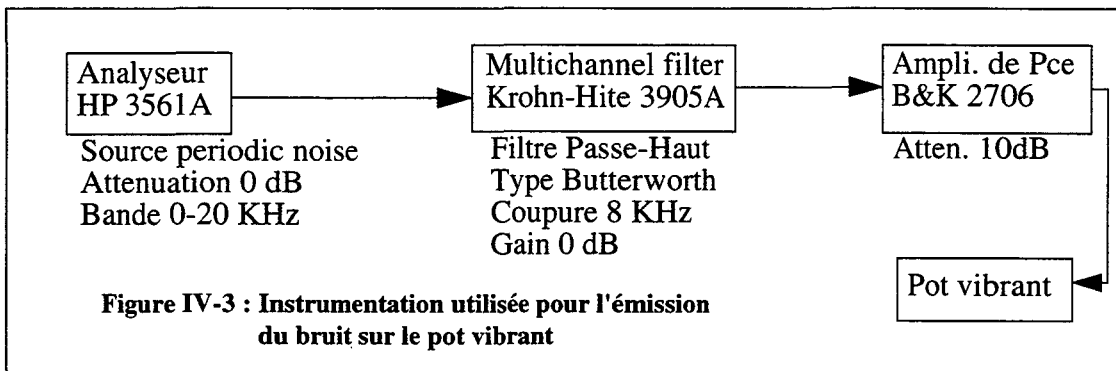
Par ailleurs, travailler avec des signaux dans la bande 0-20KHz sur une plaque de 1cm. d'épaisseur équivaut à simuler le bruit propre dans la bande 0-6KHz sur une plaque d'environ 3cm. d'épaisseur, soit approximativement l'épaisseur d'une coque de SNA (Sous-marin Nucléaire d'Attaque). La structure mécanique choisie est donc tout à fait représentative de la réalité physique.

IV.2.2. INSTRUMENTATION UTILISEE

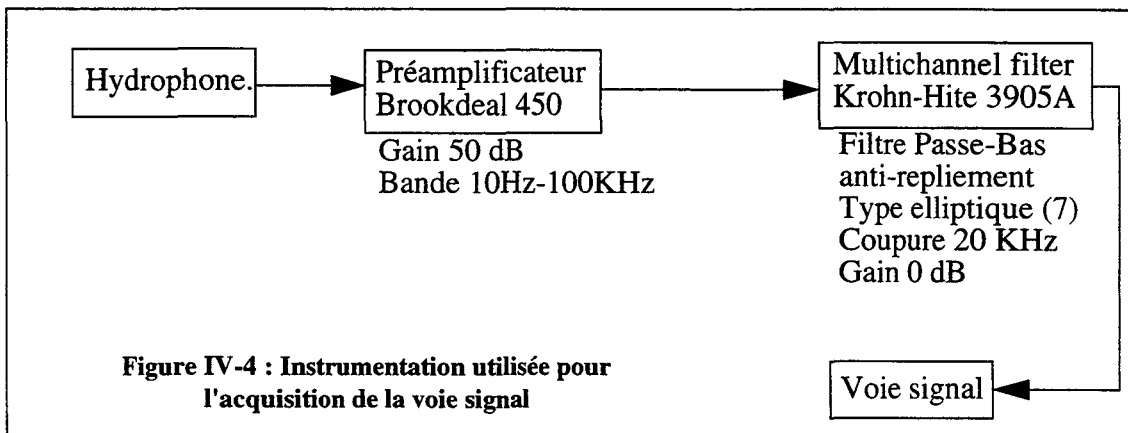
Pour l'émission du signal utile sur le projecteur acoustique, nous avons utilisé un générateur de fonctions HP3314A. Le signal passe dans un amplificateur de puissance Bruel&Kjaer 2713 avant d'attaquer le projecteur Massa TR1025C (voir Figure IV-2).

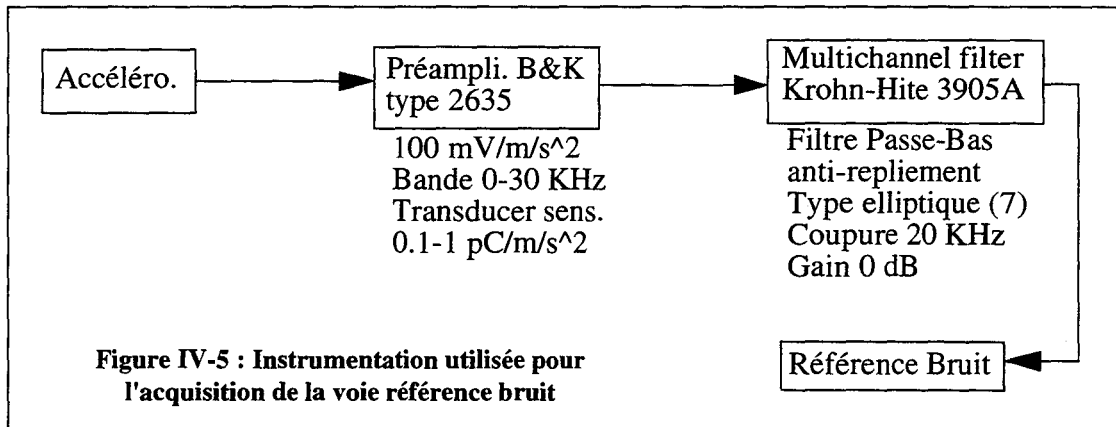


Pour l'émission du bruit sur le pot vibrant, nous avons utilisé le générateur de bruit blanc pseudo-aléatoire de l'analyseur de spectre HP3561A. Le signal est un bruit blanc stationnaire dans la bande 0-20KHz. Avant d'attaquer le pot vibrant, ce signal passe dans un filtre passe-haut (multichannel filter Krohn-Hite 3905A) et un amplificateur de puissance Bruel&Kjaer 2706 (voir Figure IV-3).



Concernant les voies de réception (voie signal et référence bruit) : la voie signal est constituée du signal mesuré par l'hydrophone, préamplifié et filtré (voir Figure IV-4) ; la référence bruit est constituée du signal mesuré par l'accéléromètre, également préamplifié et filtré (voir Figure IV-5).

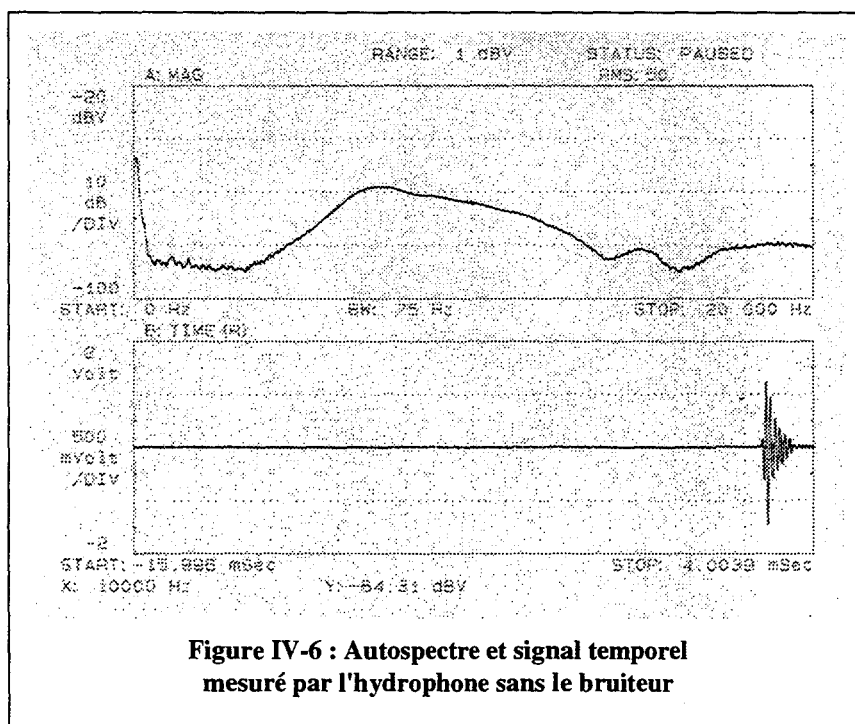




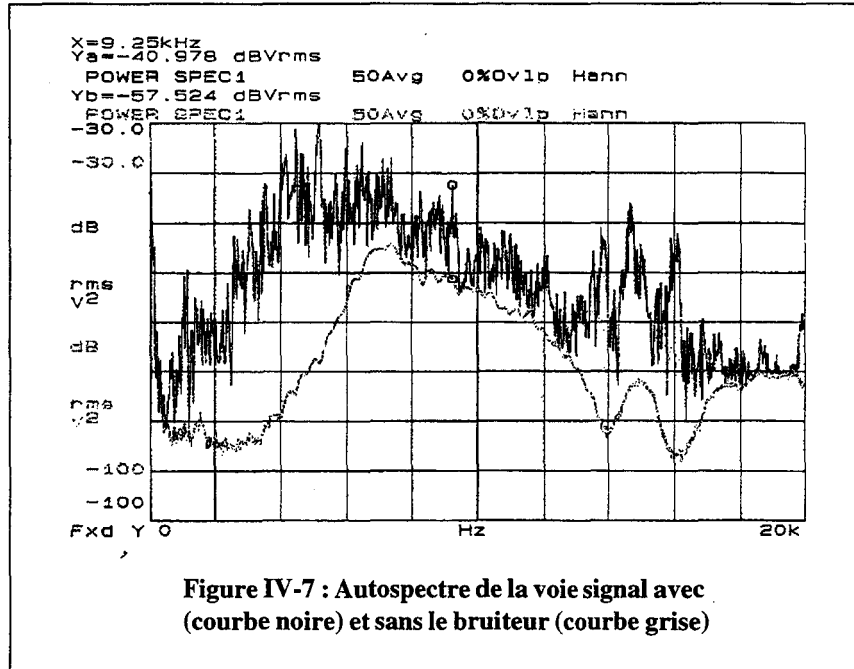
Ce dispositif expérimental, bien que sommaire, nous permet de nous rapprocher d'une situation de bruit propre classiquement rencontrée en Acoustique Sous-Marine. La voie signal (hydrophone) simule la voie audio d'un sonar embarqué et la voie référence (accéléromètre) fournit l'information nécessaire à la soustraction de bruit.

IV.2.3. ANALYSE DES SIGNAUX

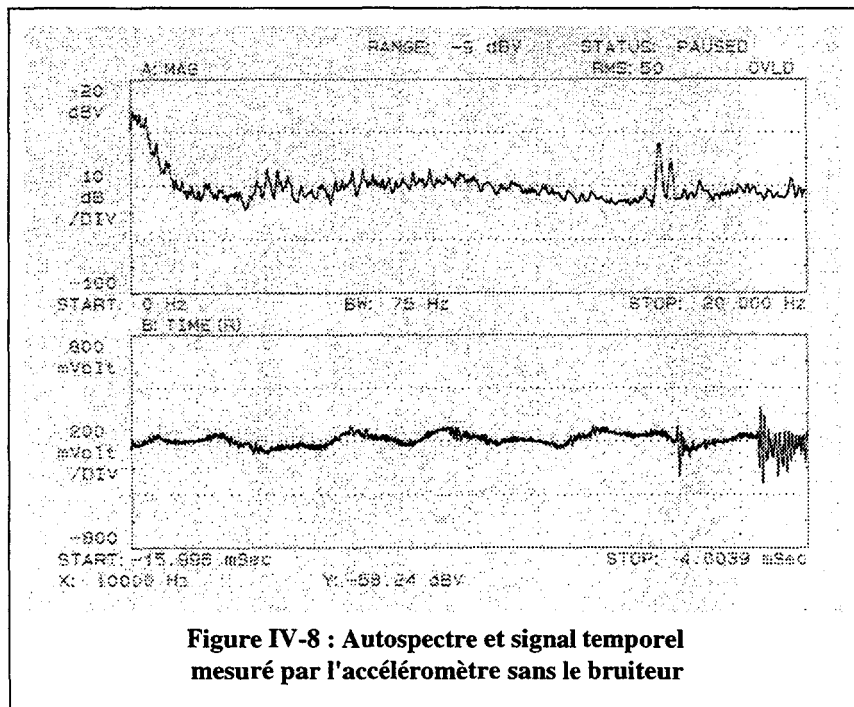
Le signal mesuré par l'hydrophone sans le bruiteur est un burst sinusoïdal d'amplitude +/-1.5V, qui correspond à l'émission du signal utile par le projecteur (voir Figure IV-6). On a fait intervenir un retard de 16msec. dans la fenêtre d'analyse, de manière à s'affranchir des échos sur les parois du bassin. Le spectre correspondant est large bande ; on remarque que le support fréquentiel du signal utile se situe entre 4KHz et 16KHz. L'analyse spectrale est faite par périodogramme moyenné sur 50 coups (FFT sur 1024 points avec fenêtrage de Hanning).



Lors de la mise en route du bruiteur, le signal utile est noyé dans un bruit qui se manifeste essentiellement sous forme de raies (voir Figure IV-7). Ces raies correspondent aux modes propres de vibration de la plaque ; elles ont une amplitude de 10 à 30 dBVrms supérieure au niveau spectral du signal utile. On est bien en présence d'un signal large bande et d'un bruit à spectre de raies, caractéristique du bruit propre d'origine mécanique.



On retrouve le burst de signal utile dans le signal mesuré par l'accéléromètre sans le bruiteur. Le découplage de la référence bruit par rapport au milieu acoustique n'est donc pas très bon (voir Figure IV-8), ce qui n'est pas étonnant avec une plaque de 1cm. d'épaisseur.



Une fois le bruiteur mis en route, l'accéléromètre répond parfaitement et la contribution du signal utile se retrouve noyée dans le bruit de vibration de la plaque (voir Figure IV-9).

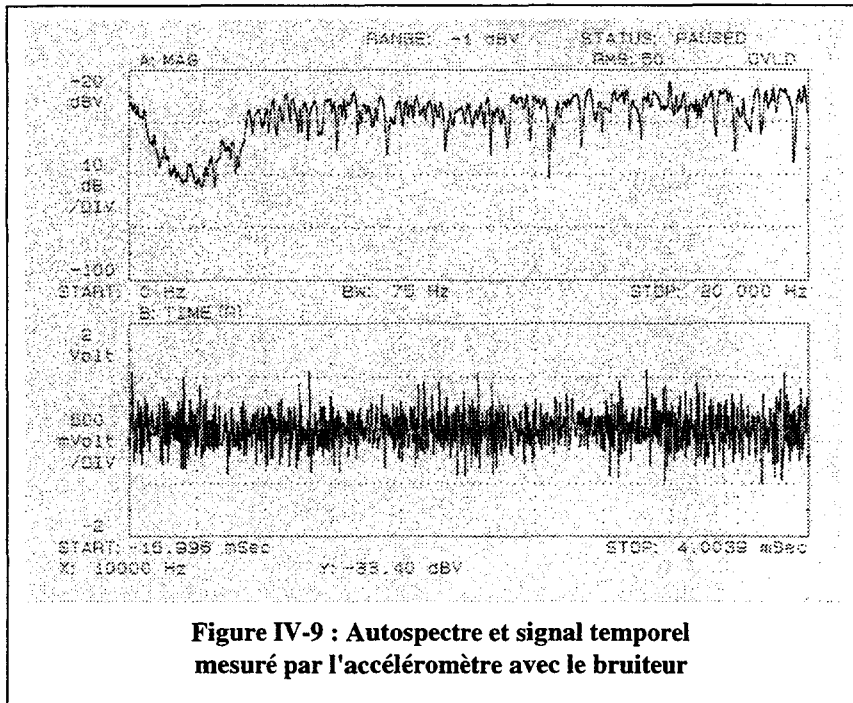


Figure IV-9 : Autospectre et signal temporel mesuré par l'accéléromètre avec le bruiteur

A l'examen de la fonction de cohérence entre la voie signal et la voie référence sans le bruiteur (voir Figure IV-10), on constate que l'hydrophone et l'accéléromètre sont bien corrélés sur le signal utile, d'où à priori un risque de détérioration de ce signal par le traitement de soustraction de bruit. Néanmoins, ils sont également très bien corrélés sur les raies de bruit (voir Figure IV-10 : la cohérence entre la voie signal et la voie référence en présence du bruiteur est proche de 1 sur les bandes 3-7KHz et 13-17KHz), ce qui laisse espérer un traitement de soustraction de bruit efficace pour ces canaux fréquentiels.

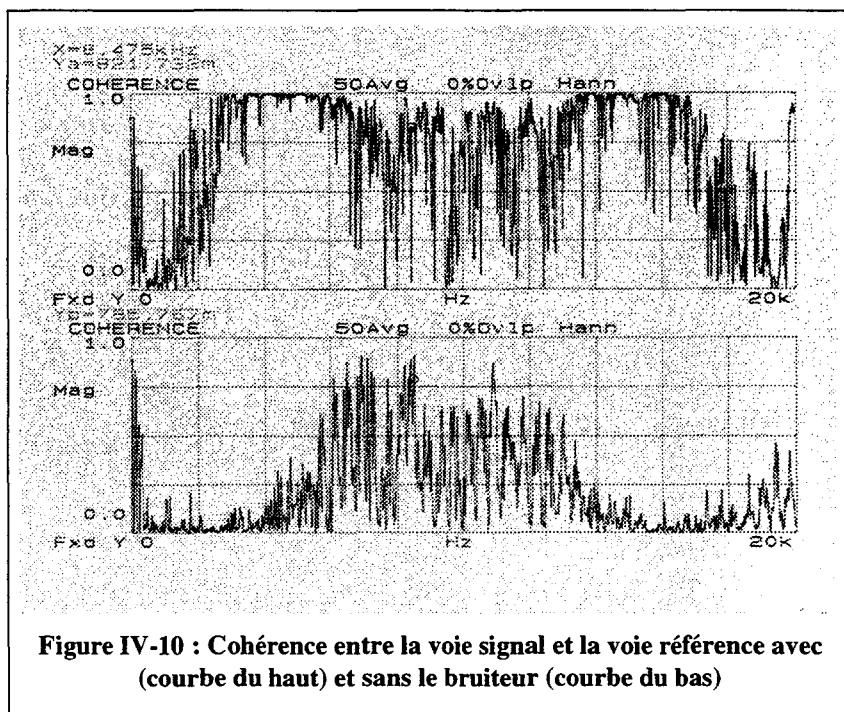
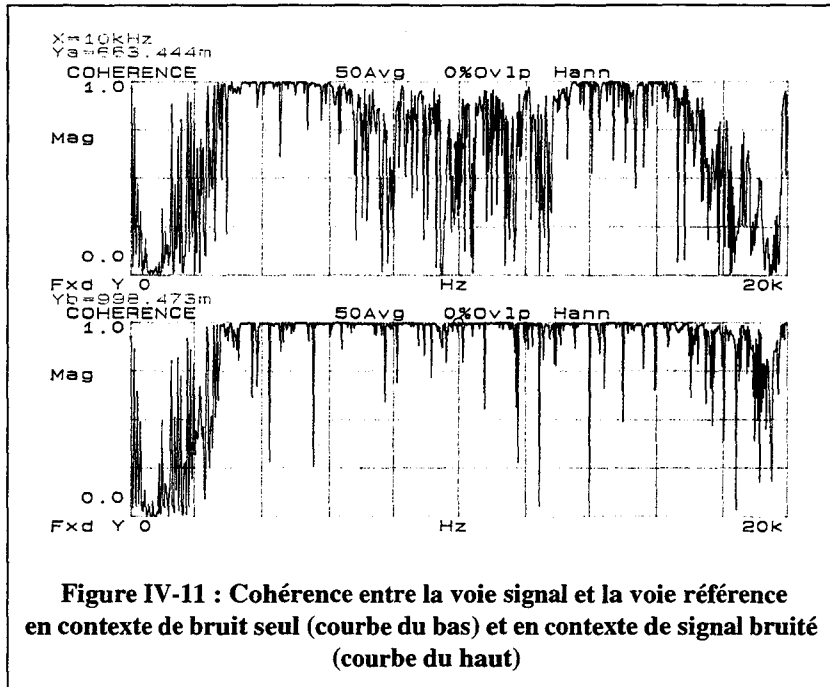


Figure IV-10 : Cohérence entre la voie signal et la voie référence avec (courbe du haut) et sans le bruiteur (courbe du bas)

Le signal utile, dont la densité spectrale de puissance est du même ordre de grandeur que le bruit perturbateur dans la bande 6-13KHz, intervient clairement comme un terme de pollution. En effet, c'est bien le signal utile qui est responsable de la dégradation de la fonction de cohérence entre la voie signal et la voie référence dans la bande 6-13KHz (voir Figure IV-11).



IV.2.4. RESULTATS OBTENUS EN SOUSTRACTION DE BRUIT

IV.2.4.1. ALGORITHME DU SPECTROFILTRE ADAPTATIF

Dans cette première expérimentation, l'algorithme du spectrofiltre adaptatif a été mis en oeuvre sans les fonctionnalités décrites au chapitre II. Pour quantifier ses performances, nous avons isolé 9 raies de vibration dans l'autospectre de la voie signal. Dans le tableau ci-dessous figurent les fréquences de ces raies, ainsi que le gain de traitement et la chute de cohérence obtenus sur chacune de ces raies par traitement de soustraction de bruit. La fréquence d'échantillonnage est de 32KHz, la taille des tranches vaut 1024, le taux de recouvrement 50% et le facteur d'oubli vaut 1.

Table 10: Résultats obtenus par le spectrofiltre adaptatif

Fréquence (KHz)	Gain de traitement (dB)	Chute de cohérence
4.2	26.3	0.38
5.1	25.3	0.77
6.4	18.9	0.35
8.3	17.9	0.84
9.3	19.5	0.58

Table 10: Résultats obtenus par le spectrofiltre adaptatif

Fréquence (KHz)	Gain de traitement (dB)	Chute de cohérence
13.5	11	0.1
13.8	17.4	0.25
14.7	15.3	0.47
16.1	22.1	0.45

Le spectrofiltre adaptatif donne des résultats satisfaisants sur ce type de signaux : la moyenne du gain de traitement sur les raies de bruit vaut approximativement 20dB, et la moyenne de la chute de cohérence associée entre la voie signal et la voie référence vaut approximativement 0.4 (voir Figures IV-12 et IV-13).

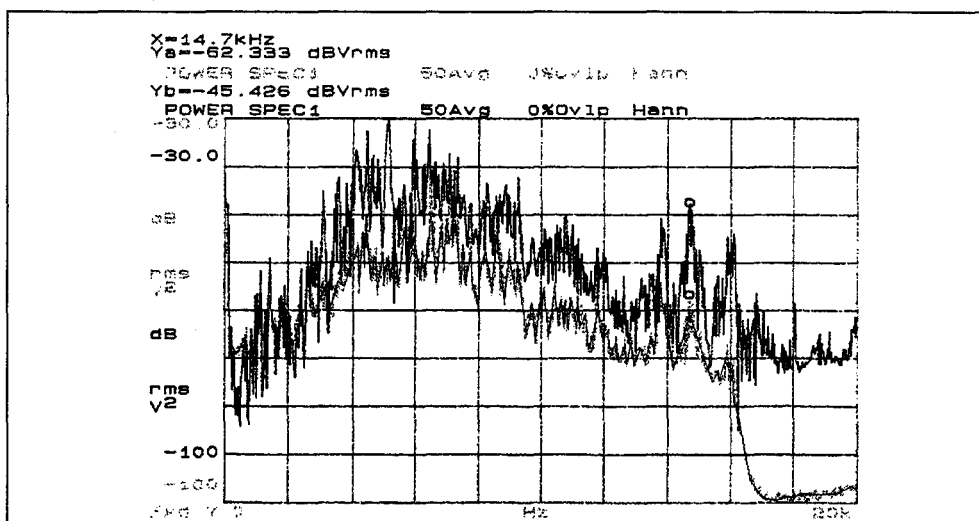


Figure IV-12 : Autospectre de la voie signal avant (courbe noire) et après (courbe grise) soustraction de bruit par le spectrofiltre adaptatif

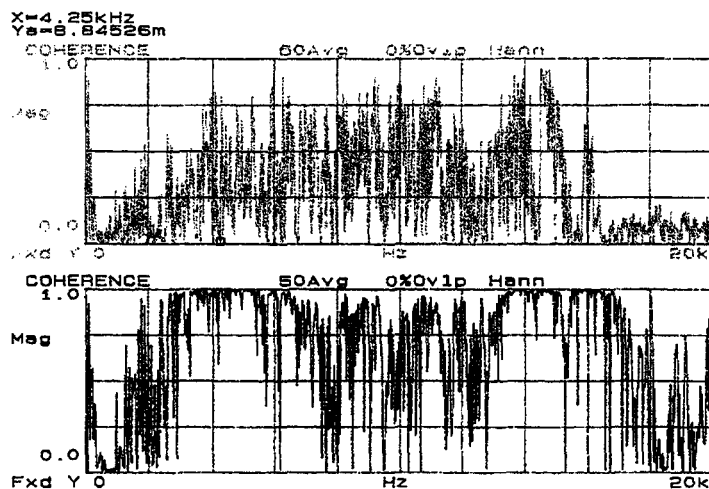
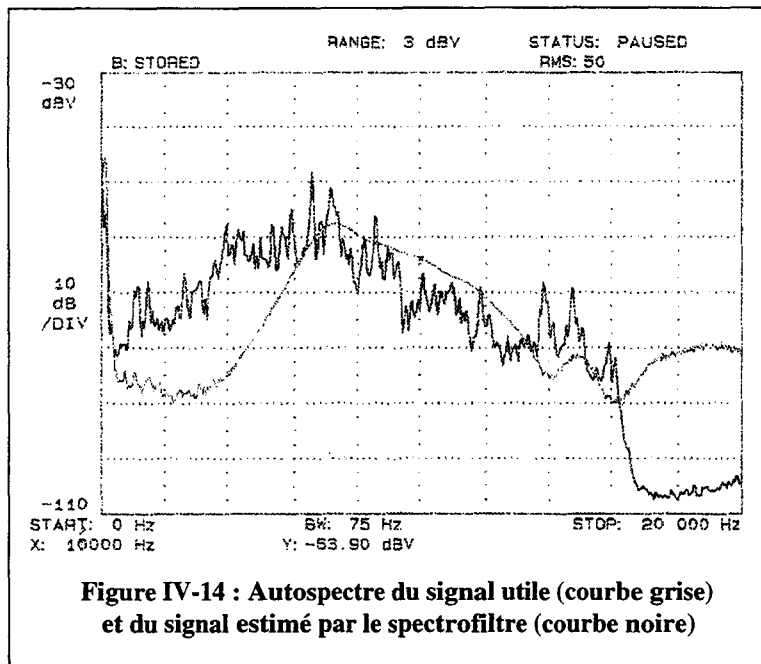


Figure IV-13 : Cohérence entre voie signal et voie référence avant (courbe noire) et après (courbe grise) soustraction de bruit par le spectrofiltre adaptatif

Néanmoins, le gain de traitement est nul, voire négatif pour les fréquences inférieures à 3KHz (voir Figure IV-12). De plus, on observe une dégradation spectrale du signal utile dans la bande 6-13KHz (voir Figure IV-14). Pour cette dernière raison, nous ne parvenons pas à retrouver le signal utile (burst sinusoïdal) dans le domaine temporel après le traitement de soustraction de bruit.



IV.2.4.2. ALGORITHME LMS TEMPOREL

En ce qui concerne l'algorithme LMS temporel, les résultats sont nettement moins bons que pour le spectrofiltre adaptatif. Le gain de traitement et la chute de cohérence sont quasiment nuls sur les raies de bruit. Nous avons attribué ce mauvais résultat au fait que le support temporel du filtre à identifier est trop long. Avec une fréquence d'échantillonnage de 32KHz, nous sommes limités à un ordre de 85 pour le filtre adaptatif, ce qui est manifestement insuffisant, vraisemblablement à cause des échos sur les bords de la plaque et sur les parois du bassin, qui contribuent naturellement à allonger la réponse impulsionnelle du filtre à identifier. Nous remarquons que les résultats obtenus par le LMS transverse sont ici indépendants du choix du pas d'adaptation ; c'est donc bien l'ordre du filtre qui est en cause.

IV.2.5. INTERPRETATION DES RESULTATS OBTENUS

Cette première expérimentation met en évidence la supériorité des méthodes fréquentielles sur les méthodes temporelles pour la soustraction du bruit mécanique de plaque sous une contrainte sévère de fonctionnement en temps réel. En particulier, l'algorithme du spectrofiltre adaptatif donne des résultats intéressants en termes de gain de traitement et de chute de cohérence entre la voie signal et la référence bruit. Néanmoins, nous ne parvenons pas à retrouver le signal utile dans le domaine temporel.

Nous allons montrer que la puissance du signal utile est une cause directe de ce phénomène et proposer des solutions en vue d'améliorer les résultats expérimentaux obtenus jusqu'à présent.

IV.2.5.1. INFLUENCE DE LA PUISSANCE DU SIGNAL UTILE

Plaçons-nous dans un canal fréquentiel v . Les signaux sont découpés en tranches avec la méthode par recouvrement de blocs. Pour une tranche de rang j , on peut toujours écrire les transformées de Fourier de la voie signal et de la voie référence de la manière suivante :

- voie signal : $X_j(v) = S_j(v) + V_j(v)$
- voie référence : $U_j(v) = b_j(v) + a \cdot S_j(v)$

où a représente le coefficient de couplage sur le signal utile entre la voie signal et la voie référence pour le canal fréquentiel v . Après r tranches consécutives, le gain fréquentiel du filtre adaptatif calculé par le spectrofiltre vaut, pour le canal fréquentiel v :

$$\hat{w}^r(v) = \frac{\hat{\gamma}_{XU}^r(v)}{\hat{\gamma}_{UU}^r(v)} \quad (\text{EQ IV-1})$$

où $\hat{\gamma}_{XU}^r(v)$ (resp. $\hat{\gamma}_{UU}^r(v)$) est l'estimation de l'interspectre entre la voie signal $x(n)$ et la voie référence $u(n)$ (resp. : estimation de l'autospectre de la voie référence) après acquisition des tranches de signal de rang r , pour le canal fréquentiel v . Si on développe l'équation (EQ IV-1) avec la décomposition précédente, on obtient l'équation (EQ IV-2) :

$$\hat{w}^r(v) = \frac{\hat{\gamma}_{XU}^r(v)}{\hat{\gamma}_{UU}^r(v)} = \frac{\hat{\gamma}_{Sb}^r(v) + \hat{\gamma}_{Vb}^r(v) + a \cdot \hat{\gamma}_{SS}^r(v) + a \cdot \hat{\gamma}_{VS}^r(v)}{\hat{\gamma}_{bb}^r(v) + a \cdot \hat{\gamma}_{Sb}^r(v) + a \cdot \hat{\gamma}_{bS}^r(v) + a^2 \cdot \hat{\gamma}_{SS}^r(v)} \quad (\text{EQ IV-2})$$

Par ailleurs, la cohérence quadratique entre la voie signal et la voie référence s'écrit :

$$|\hat{C}_{XU}^r(v)|^2 = \frac{|\hat{\gamma}_{XU}^r(v)|^2}{\hat{\gamma}_{UU}^r(v) \cdot \hat{\gamma}_{XX}^r(v)} = \frac{|\hat{\gamma}_{Sb}^r(v) + \hat{\gamma}_{Vb}^r(v) + a \cdot \hat{\gamma}_{SS}^r(v) + a \cdot \hat{\gamma}_{VS}^r(v)|^2}{(\hat{\gamma}_{bb}^r(v) + a \cdot \hat{\gamma}_{Sb}^r(v) + a \cdot \hat{\gamma}_{bS}^r(v) + a^2 \cdot \hat{\gamma}_{SS}^r(v)) \cdot (\hat{\gamma}_{SS}^r(v) + \hat{\gamma}_{VV}^r(v) + \hat{\gamma}_{SV}^r(v) + \hat{\gamma}_{VS}^r(v))} \quad (\text{EQ IV-3})$$

Nous allons étudier qualitativement l'influence du rapport signal à bruit présent sur la voie signal sur le gain fréquentiel du filtre adaptatif défini par l'équation (EQ IV-2) et sur la cohérence quadratique définie par l'équation (EQ IV-3). Pour cela, on peut distinguer plusieurs cas :

- $\hat{\gamma}_{SS}^r(v) \ll \hat{\gamma}_{VV}^r(v)$: le rapport signal à bruit sur la voie signal est très faible dans le canal fréquentiel v . Si la voie référence est bien choisie (coefficient de couplage a très faible), alors on peut négliger la contribution du signal utile sur cette voie. Dans ce cas, les équations (EQ IV-2) et (EQ IV-3) se simplifient et on obtient en première approximation :

$$\hat{w}^r(v) \equiv \frac{\hat{\gamma}_{Vb}^r(v)}{\hat{\gamma}_{bb}^r(v)} \text{ et } \hat{C}_{XU}^r(v) \equiv \frac{|\hat{\gamma}_{Vb}^r(v)|^2}{\hat{\gamma}_{bb}^r(v) \cdot \hat{\gamma}_{VV}^r(v)}.$$

On constate que le filtre adaptatif identifie correctement la fonction de transfert entre le bruit $b(n)$ mesuré sur la voie référence et le bruit perturbateur $v(n)$ présent sur la voie signal. La cohérence dépend ici uniquement du degré de corrélation entre $b(n)$ et $v(n)$. Si cette corrélation est bonne, la soustraction de bruit doit fonctionner correctement dans ce cas de figure.

- $\hat{\gamma}_{SS}^r(v) \equiv \hat{\gamma}_{VV}^r(v)$: le rapport signal à bruit sur la voie signal augmente.

Si le coefficient de couplage a est suffisamment faible pour avoir $a^2 \cdot \hat{\gamma}_{SS}^r(v) \ll \hat{\gamma}_{bb}^r(v)$, alors on peut approximer les équations (EQ IV-2) et (EQ IV-3) de la manière suivante :

$$\hat{w}^r(v) \equiv \frac{\hat{\gamma}_{Vb}^r(v) + \hat{\gamma}_{Sb}^r(v)}{\hat{\gamma}_{bb}^r(v)} \text{ et } \hat{C}_{XU}^r(v) \equiv \frac{|\hat{\gamma}_{Vb}^r(v) + \hat{\gamma}_{Sb}^r(v)|^2}{\hat{\gamma}_{bb}^r(v) \cdot (\hat{\gamma}_{VV}^r(v) + \hat{\gamma}_{SS}^r(v) + \hat{\gamma}_{SV}^r(v) + \hat{\gamma}_{VS}^r(v))}.$$

La puissance du signal joue clairement un rôle perturbateur ; en effet, les densités spectrales étant estimées sur des durées d'observation finies, il existe un terme de corrélation non nul entre $b(n)$ et $s(n)$. Le calcul de la puissance d'erreur résiduelle du système soustracteur de bruit en fonction de la puissance du signal et du temps d'intégration sur lequel le filtre est estimé a été mené dans [Servière 89]. Il est montré que la puissance d'erreur résiduelle peut se décomposer en deux termes. Le premier, indépendant de la présence de signal, prend en compte les erreurs de biais effectuées sur les estimations des densités spectrales ainsi que les termes de variance. Le deuxième est directement proportionnel à la puissance du signal et inversement proportionnel au nombre de tranches r .

La présence de signal, en entraînant une augmentation de la puissance de l'erreur résiduelle, peut donc à priori invalider la soustraction de bruit. En pratique, il existe une valeur minimale de r en-dessous de laquelle le traitement de soustraction de bruit est inefficace [Servière 89], c'est-à-dire en-dessous de laquelle la puissance de l'erreur résiduelle est supérieure à la puissance du bruit perturbateur sur la voie signal. Cette valeur minimale est directement liée au rapport signal à bruit sur la voie signal. Plus le rapport signal à bruit sur la voie signal est élevé, plus il faudra un nombre de moyennes (r) important pour que la soustraction de bruit devienne efficace. Néanmoins, pour que ce raisonnement soit valable, il faut que l'hypothèse $a^2 \cdot \hat{\gamma}_{SS}^r(v) \ll \hat{\gamma}_{bb}^r(v)$ soit vérifiée. Il est intéressant de noter que cette situation peut être détectée en examinant la fonction de cohérence entre la voie signal et la voie référence : celle-ci doit être dégradée car la puissance du signal utile intervient au dénominateur.

Si l'hypothèse $a^2 \cdot \hat{\gamma}_{SS}^r(v) \ll \hat{\gamma}_{bb}^r(v)$ n'est pas vérifiée, on se trouve en présence d'un cas de pollution de la voie référence par le signal utile. Le rapport signal à bruit sur la voie référence

étant élevé, il est fort probable que le rapport signal à bruit sur la voie signal le soit également, auquel cas la cohérence vaut 1 et le gain fréquentiel du filtre adaptatif vaut $1/a$: il y a alors soustraction de signal.

Cette analyse qualitative permet de montrer que la puissance du signal utile joue un rôle perturbateur pour la soustraction de bruit. En fin de compte, deux situations risquent de dégrader le signal pour un canal fréquentiel donné :

- pollution de la voie référence ; dans ce cas, il y a forcément annulation du signal utile. La seule solution est de chercher une autre voie référence moins polluée ;
- pas de pollution de la voie référence, mais la puissance du signal est assez élevée pour apparaître dans l'autospectre de la voie signal. Les coefficients du filtre devraient théoriquement être nuls pour les canaux fréquentiels concernés, mais ne le sont pas car les densités spectrales sont estimées sur des durées d'observation finies. On a alors un bruit de calcul théoriquement non-corrélé avec le signal utile, mais qui va malgré tout dégrader le signal.

C'est à cette seconde situation que nous avons été vraisemblablement confrontés dans la première expérimentation pour les canaux fréquentiels entre 6KHz et 14KHz. Or on vient de voir qu'on peut détecter cette situation en étudiant les valeurs de la cohérence entre la voie signal et la voie référence : il semble donc à priori indispensable de procéder à une estimation préalable de la cohérence avant d'effectuer le traitement de soustraction de bruit.

IV.2.5.2. MODIFICATIONS ALGORITHMIQUES ENVISAGEES

Au paragraphe IV.2.4.1, nous avons lié le fait que nous ne parvenons pas à retrouver le signal utile dans le domaine temporel avec la dégradation spectrale du signal utile observée dans la bande 6-13KHz (voir Figure IV-14). Or cette dégradation spectrale se produit au voisinage des canaux fréquentiels pour lesquels le rapport signal à bruit sur la voie signal est élevé (voir Figure IV-7). Nous pouvons donc penser que nous nous trouvons dans la situation mise en évidence au paragraphe précédent. Nous avons vu que cette situation peut être détectée en estimant la fonction de cohérence entre la voie signal et la voie référence avant le traitement de soustraction de bruit. Le contrôle amont du traitement est donc la principale modification algorithmique à apporter en vue d'une amélioration des résultats de soustraction de bruit.

Nous allons également mettre en oeuvre les autres fonctionnalités envisagées pour les méthodes fréquentielles, afin d'étudier leur influence sur la qualité de la soustraction de bruit.

Ces fonctionnalités sont impossibles à mettre en oeuvre pour les méthodes temporelles. Pour ces méthodes, la seule source d'amélioration possible réside dans une réduction de la fréquence d'échantillonnage de manière à pouvoir mettre en oeuvre en temps réel un filtre adaptatif de taille supérieure.

IV.3. SECONDE VALIDATION DES ALGORITHMES

IV.3.1. VALIDATION SUR SIGNAUX DETERMINISTES

IV.3.1.1. GENERALITES ; FILTRE A IDENTIFIER

Nous voulons d'abord mettre en évidence sur des signaux déterministes les améliorations apportées par rapport à la version de base du spectrofiltre adaptatif. Dans tous les cas, le filtre à identifier sera un filtre passe-bas elliptique d'ordre 7 dont la fréquence de coupure est 5KHz. Nous avons pu caractériser sa réponse impulsionnelle à l'aide de l'analyseur de spectre HP 3562A (voir Figure IV-15).

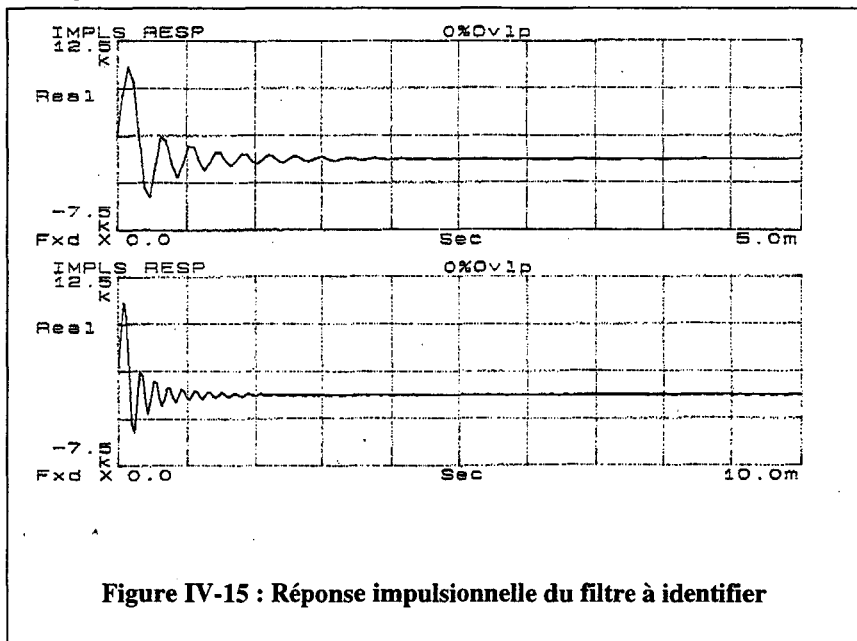


Figure IV-15 : Réponse impulsionnelle du filtre à identifier

Pour la fréquence de coupure utilisée, le filtre à identifier peut donc être approximé par un filtre RIF de longueur (support temporel) environ 2msec. Au cours de la validation, nous avons choisi plusieurs configurations de signaux. La plus utile pour mettre en évidence les différents aspects du filtrage adaptatif dans le domaine fréquentiel est la suivante : le signal utile est une sinusoïde de fréquence approximativement 3KHz ; le bruit perturbateur est un bruit blanc dans la bande 0-10KHz et le filtre à identifier est un filtre passe-bas elliptique d'ordre 7 de fréquence de coupure 5KHz. On peut évidemment faire varier le rapport signal à bruit sur la voie signal, en modifiant soit l'amplitude de la sinusoïde, soit la densité spectrale de puissance du bruit blanc.

Nous allons d'abord montrer que le contrôle amont permet d'améliorer nettement les résultats de la soustraction de bruit, à condition de choisir convenablement la valeur du seuil et le type de fenêtrage. Ensuite, nous mettrons en évidence l'influence de la puissance du signal utile sur la détermination de la valeur du seuil. Nous montrerons également que l'utilisation du contrôle aval pose des problèmes avec la méthode par recouvrement de blocs. Enfin, nous montrerons que les algorithmes permettent de retrouver le signal utile dans le domaine temporel.

IV.3.1.2. CONTROLE AMONT ET FENETRAGE DES DONNEES

IV.3.1.2.1 Analyse des signaux

On se place dans la situation suivante : le signal utile est une sinusoïde d'amplitude 1Volt et de fréquence 3KHz ; le bruit perturbateur est un bruit blanc d'amplitude maximale 1.5Volt. Le filtre à identifier a été décrit précédemment. On peut visualiser les autospectres de la voie signal (voir Figure IV-16) et de la voie référence (voir Figure IV-17), mesurés par l'analyseur de spectre.

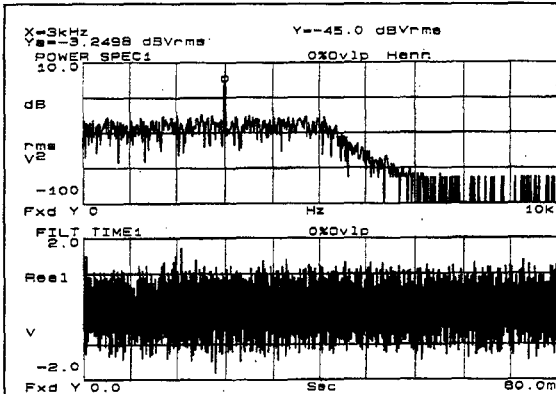


Figure IV-16 : Densité spectrale de puissance de la voie signal et signal temporel correspondant.

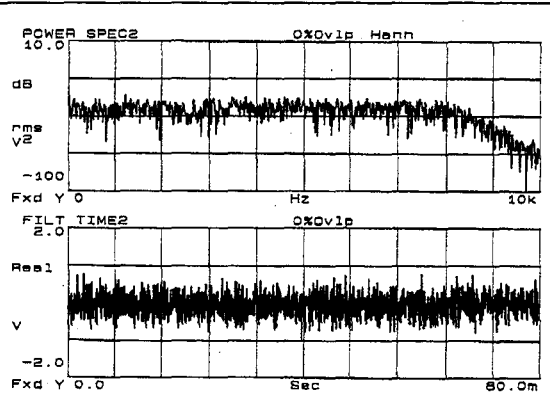


Figure IV-17 : Densité spectrale de puissance de la voie référence et signal temporel correspondant.

On se trouve dans une situation où le rapport signal à bruit est très bon dans le canal fréquentiel de la raie signal (3KHz) ; néanmoins, le signal temporel est nettement bruité (voir Figure IV-16). La fonction de cohérence avant soustraction de bruit (voir Figure IV-18) est très bonne sur le support fréquentiel du filtre à identifier, c'est-à dire dans la bande 0-7KHz. On observe une nette dégradation dans le canal fréquentiel de la raie signal, car cette raie n'apparaît pas dans l'autospectre de la voie référence (voir Figure IV-17).

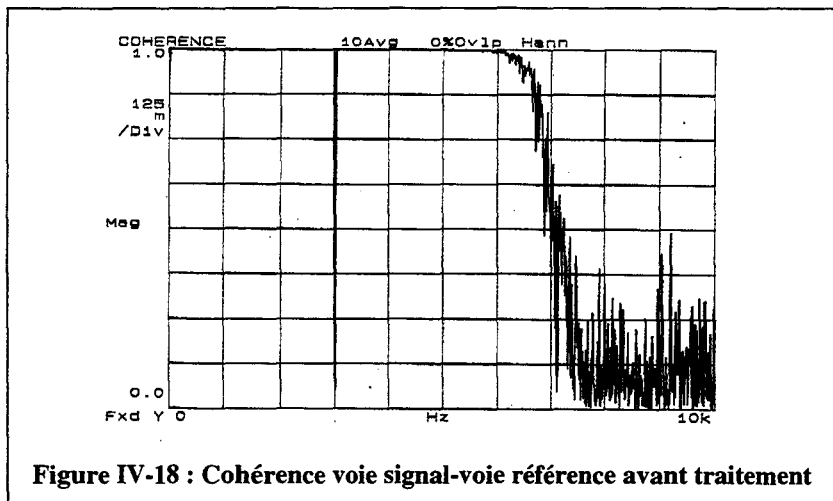


Figure IV-18 : Cohérence voie signal-voie référence avant traitement

Les algorithmes du spectrofiltre adaptatif et du gradient fréquentiel seront testés avec deux types de fenêtrage : rectangulaire (naturel) et Hamming. Pour chaque fenêtrage, deux situations seront analysées : sans et avec mise en oeuvre du contrôle amont.

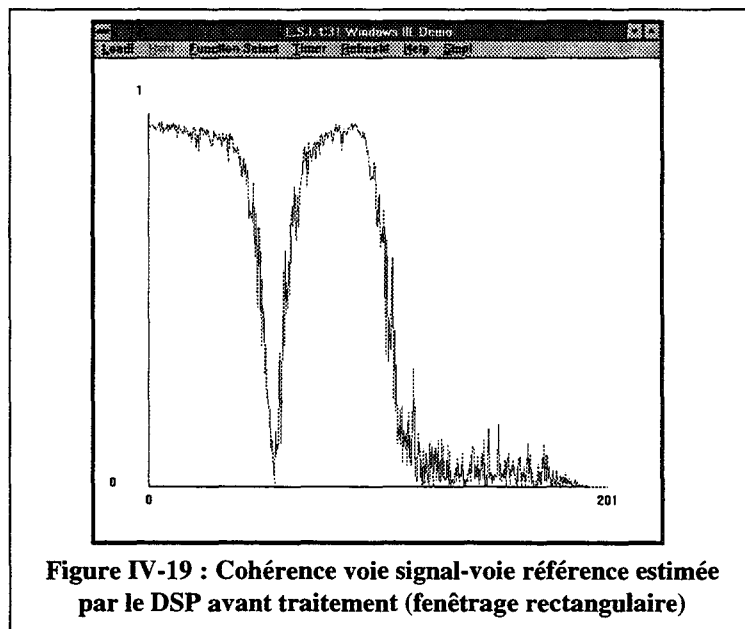
IV.3.1.2.2 Fenêtrage rectangulaire (naturel)

Nous allons tester les algorithmes de soustraction de bruit avec les paramètres suivants:

- fréquence d'échantillonnage = 22 KHz ;
- taille des blocs (c'est-à dire des FFT/IFFT) = 1024 échantillons ;
- taux de recouvrement (spectrofiltre/gradient) = 0.4375/0.3125 ;
- facteur d'oubli = 0.9 ;
- fenêtrage rectangulaire ;
- pas de contrôle amont (seuil = 0.0) ;
- pas de contrainte du gradient ;
- contrôle aval (facteur de sécurité = 1.0).

Dans un premier temps, le contrôle amont n'est pas mis en oeuvre, et on choisit le fenêtrage rectangulaire (naturel).

Le spectrofiltre adaptatif et le gradient fréquentiel procèdent à une estimation en temps réel des autospectres de la voie signal et de la voie référence, ainsi que de la fonction de cohérence. Grâce à l'interface graphique que nous avons développé entre le PC et la carte DSP, on peut visualiser ces grandeurs. La fonction de cohérence calculée par le DSP (voir Figure IV-19) est semblable à celle mesurée par l'analyseur de spectre ; néanmoins, la cohérence est comparativement nettement plus dégradée au voisinage de 3KHz, à cause du fenêtrage utilisé (Hanning pour l'analyseur de spectre et rectangulaire pour le DSP).



Les autospectres de la voie signal et de la voie référence calculés par le DSP sont identiques à ceux mesurés par l'analyseur, aux effets du fenêtrage près.

On peut maintenant analyser les résultats du spectrofiltre adaptatif : autospectre de la

voie signal (voir Figure IV-20), cohérence entre la voie signal et la voie référence (voir Figure IV-21), et signal temporel (voir Figure IV-22) avant et après traitement de soustraction de bruit. On constate un gain de traitement positif au voisinage de la raie signal (voir Figure IV-20), associé à une chute significative de la cohérence entre la voie signal et la voie référence (voir Figure IV-21) : ceci traduit clairement une soustraction de bruit.

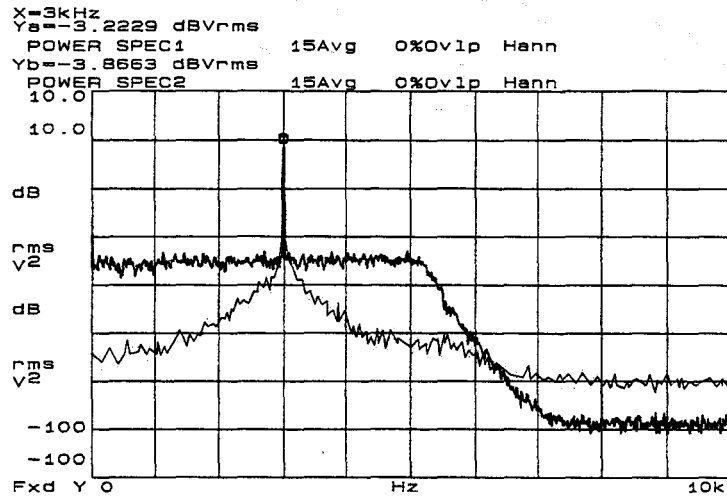


Figure IV-20 : Autospectre voie signal avant (courbe noire) et après traitement (courbe grise) par le spectrofiltre avec fenêtrage rectangulaire.

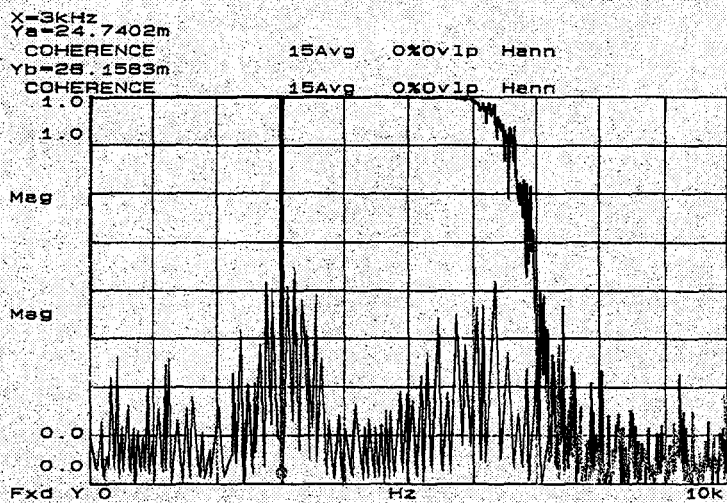
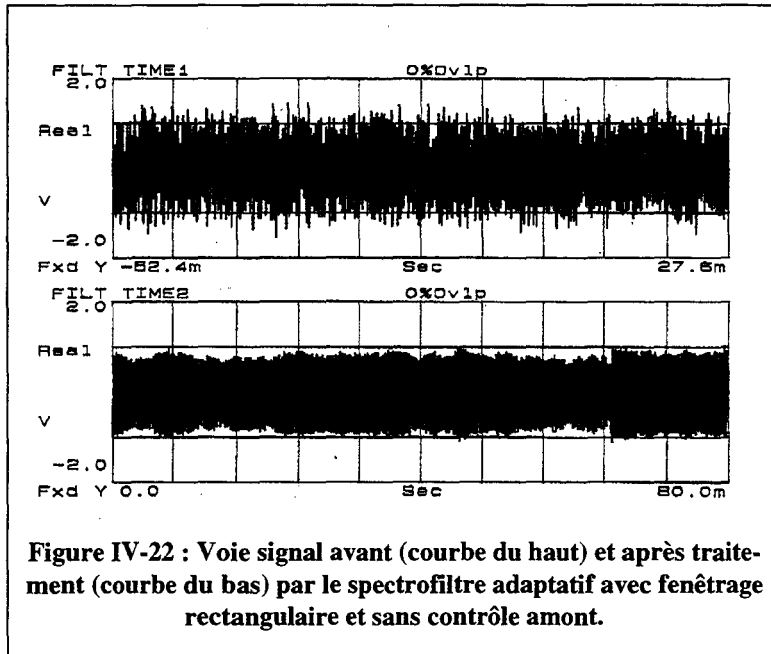
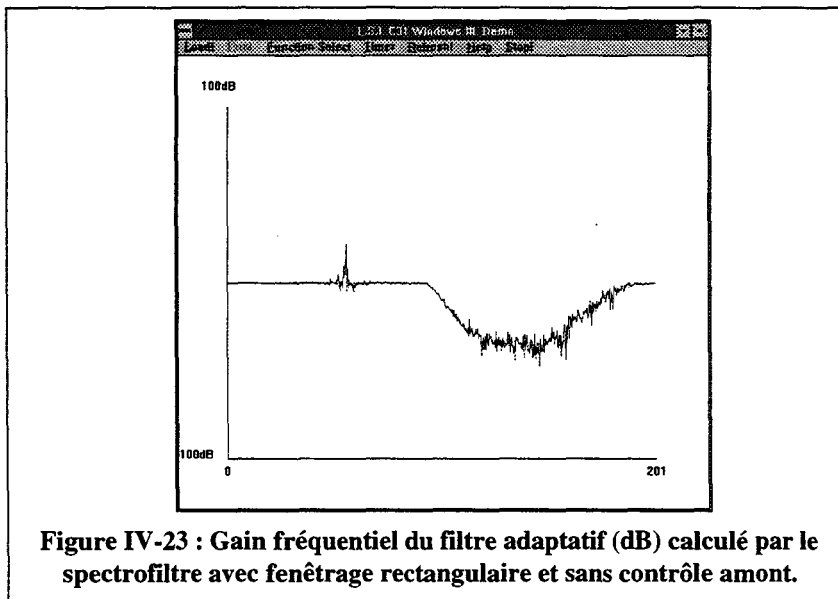


Figure IV-21 : Cohérence avant (courbe noire) et après traitement (courbe grise) par le spectrofiltre adaptatif avec fenêtrage rectangulaire.

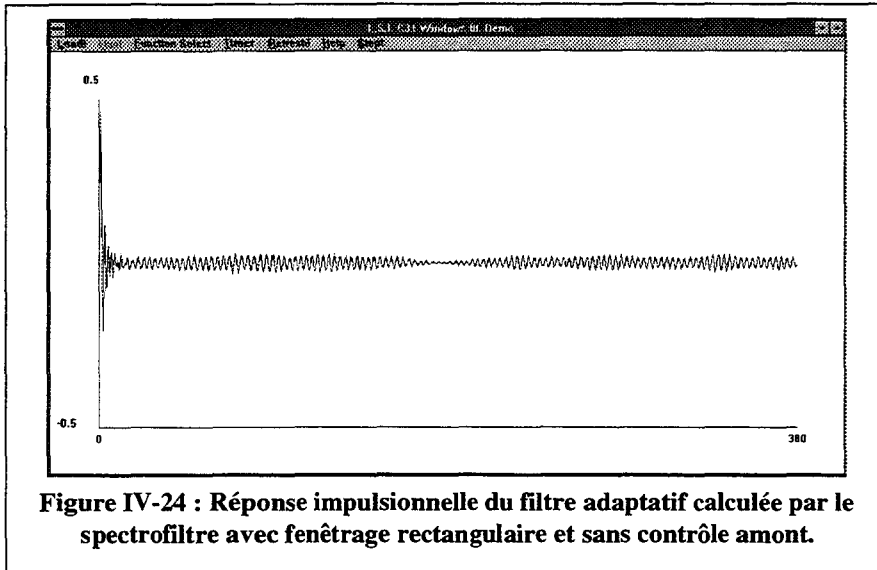


En revanche, le traitement ajoute environ 15 dBVrms de bruit pour les fréquences supérieures à 6.5KHz (voir Figure IV-20) et modifie l'amplitude de la raie signal (0.65 dBVrms d'écart mesuré par l'analyseur de spectre). En fait, le gain fréquentiel du filtre n'est pas nul sur ces canaux fréquentiels alors qu'il devrait l'être, étant donnée la valeur de la cohérence.

L'interface graphique permet de visualiser le gain fréquentiel du filtre adaptatif estimé par le spectrofiltre (voir Figure IV-23). Comme on pouvait s'y attendre, ce gain est élevé au-dessus de 6.5KHz, d'où le gain de traitement négatif dans cette zone (voir Figure IV-20). Par ailleurs, le gain fréquentiel du filtre adaptatif est élevé au voisinage de la raie signal (3KHz) ; bien que la cohérence soit très faible sur ces canaux fréquentiels, l'algorithme identifie une corrélation entre la voie signal et la voie référence. Il s'agit d'un bruit de calcul lié au fait que les FFT/IFFT sont effectuées sur des durées finies.

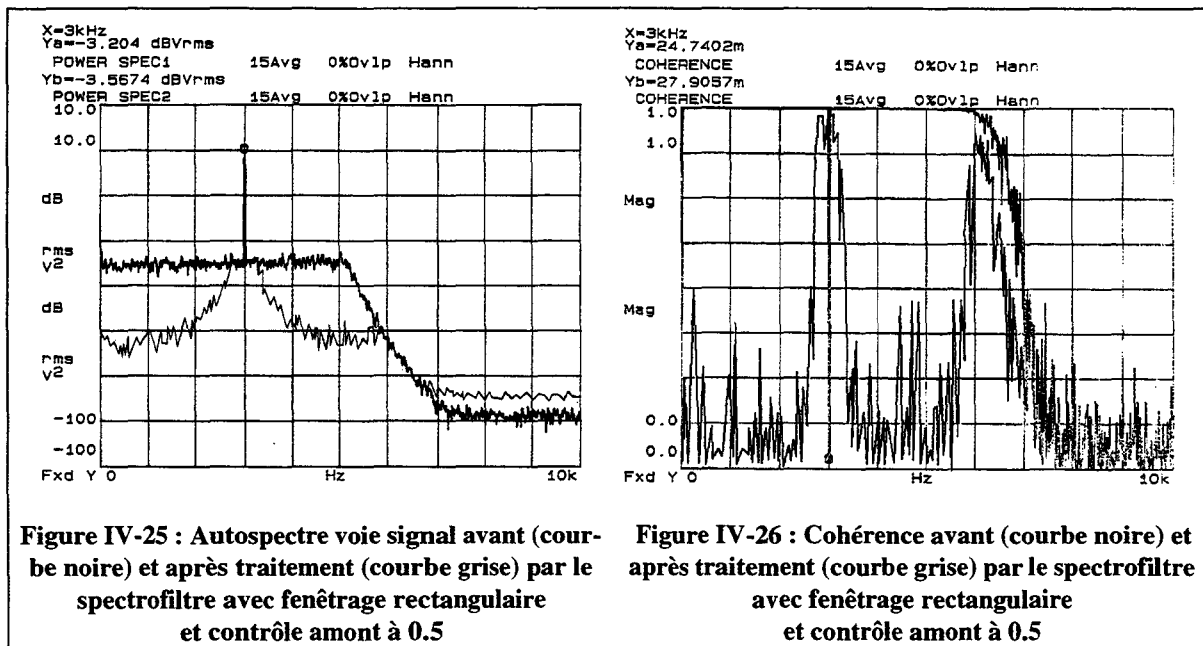


L'interface graphique permet également d'obtenir la réponse impulsionnelle du filtre adaptatif par simple IFFT de son gain fréquentiel (voir Figure IV-24). L'identification est bonne, mais la puissance du signal utile empêche une identification parfaite. En particulier, le signal utile donne une contribution qui allonge significativement la réponse impulsionnelle du filtre.

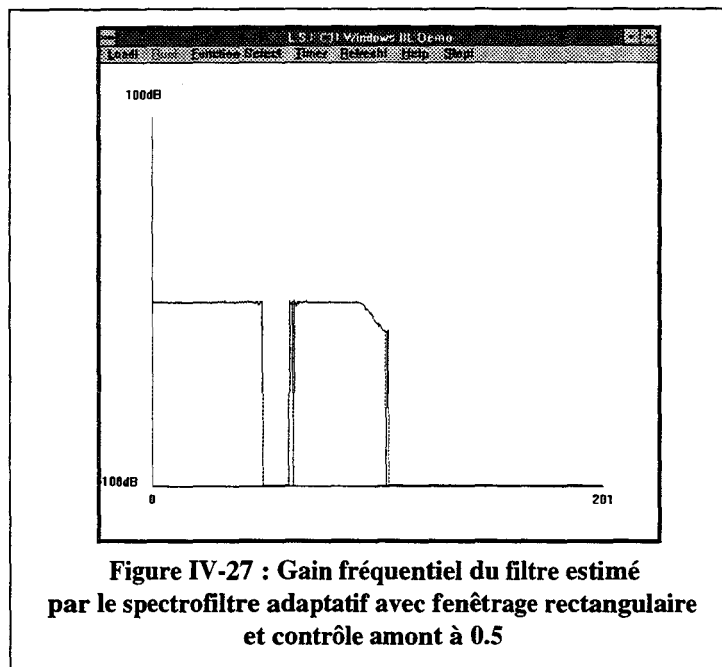


Les résultats du gradient fréquentiel étant similaires à ceux du spectrofiltre, ils ne figurent pas ici. Nous allons maintenant voir que l'utilisation du contrôle amont sur la fonction de cohérence permet d'améliorer nettement les résultats de soustraction de bruit.

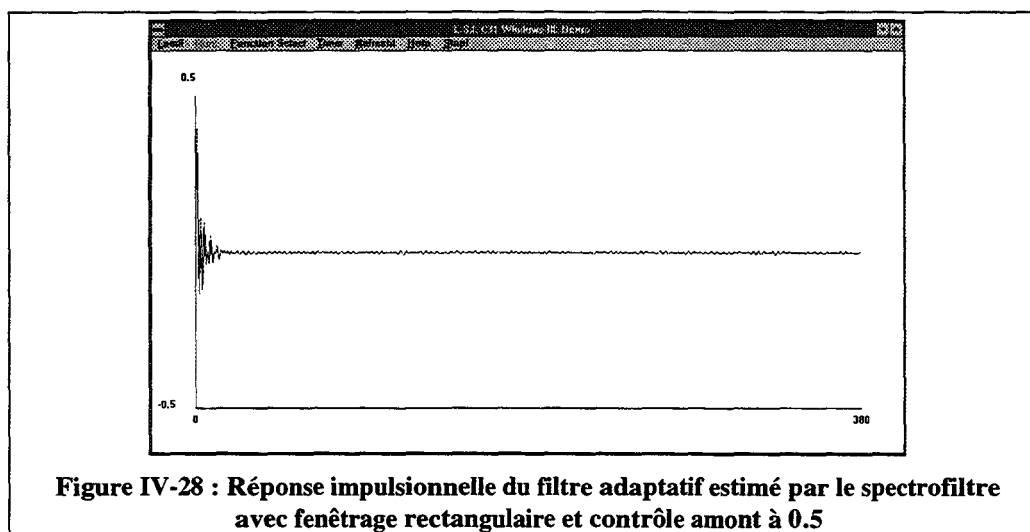
On garde les mêmes paramètres ; le seuil utilisé sur la fonction de cohérence est maintenant fixé à 0.5. Avec ce nouveau paramètre, le gain de traitement (voir Figure IV-25) est nul dans une large zone au voisinage de la raie signal, et on rajoute moins de bruit au-dessus de 6.5KHz (comparer Figures IV-25 et IV-20). A cause du seuil fixé à 0.5 sur la fonction de cohérence, la chute de cohérence par soustraction de bruit (voir Figure IV-26) est quasiment nulle au voisinage de la raie signal et au-dessus de 6.5KHz.



Dans le domaine temporel, on ne note pas d'amélioration par rapport au traitement précédent sans contrôle amont ; ceci est dû à l'utilisation du fenêtrage rectangulaire. En effet, la cohérence estimée par le DSP présente un creux qui n'est pas assez abrupt au voisinage de la raie signal (voir Figure IV-19) ; le seuillage à 0.5 entraîne donc une large zone de non-soustraction de bruit autour de la raie signal, comme on peut le voir sur le gain fréquentiel du filtre adaptatif fourni par l'interface graphique (voir Figure IV-27).



Il est également intéressant de constater qu'en interdisant le traitement au voisinage de la raie signal, on identifie nettement mieux la réponse impulsionnelle du filtre passe-bas (voir Figure IV-28). La contribution du signal utile disparaît (comparer Figures IV-24 et IV-28)



Le contrôle amont permet donc d'améliorer l'identification du filtre, notamment dans les situations où la puissance du signal utile est élevée. On vient également de mettre en évidence l'importance du fenêtrage. Pour la configuration de signaux que nous avons à traiter, le fenêtrage rectangulaire est un mauvais choix car le rapport signal à bruit est élevé dans le canal fréquentiel

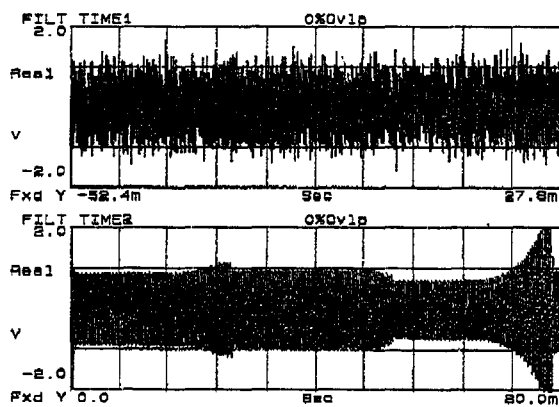
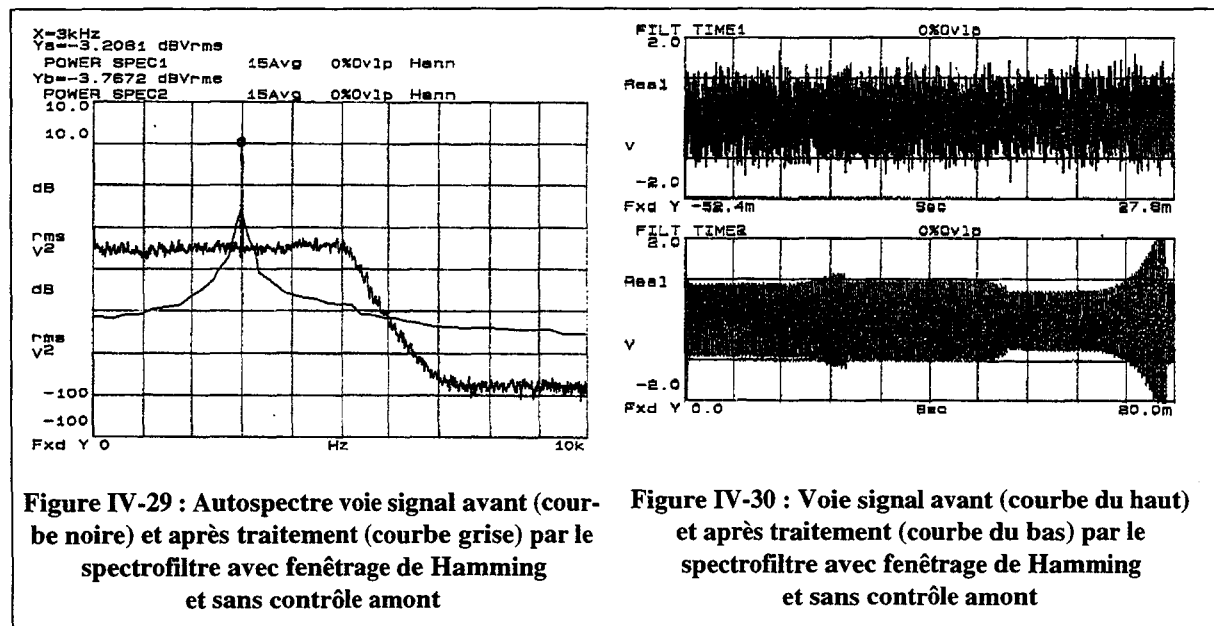
de la raie signal. Etudions maintenant les performances de la soustraction de bruit avec un fenêtrage de Hamming.

IV.3.1.2.3 Fenêtrage de Hamming

Nous allons tester le spectrofiltre adaptatif avec le fenêtrage de Hamming. Dans un premier temps, nous n'utilisons pas le contrôle amont. Rappelons les paramètres utilisés :

- fréquence d'échantillonnage = 22 KHz ;
- taille des blocs (et des FFT/IFFT) = 1024 échantillons ;
- taux de recouvrement (spectrofiltre/gradient) = 0.4375/0.3125 ;
- facteur d'oubli = 0.9 ;
- fenêtrage de Hamming;
- pas de contrôle amont (seuil = 0.0) ;
- pas de contrainte du gradient ;
- contrôle aval (facteur de sécurité = 1.0).

Avec ces paramètres, le spectrofiltre adaptatif présente des résultats plutôt médiocres: le traitement ajoute du bruit autour de la raie signal et au-dessus de 6.5KHz (environ 25 dBVrms : voir Figure IV-29). Le signal temporel estimé (voir Figure IV-30) présente des battements importants, notamment en fin de bloc. Ces battements sont dûs au fait que le gain fréquentiel du filtre estimé (voir Figure IV-31) n'est pas nul sur le canal de la raie signal et pour les fréquences supérieures à 6.5KHz ; cet effet est encore accentué par le fenêtrage utilisé (les données de fin de bloc étant pondérées très faiblement par la fenêtre de Hamming, elles sont moins prises en compte dans le traitement de soustraction de bruit, et le fenêtrage inverse utilisé pour revenir dans le domaine temporel amplifie les erreurs de calcul sur ces données).



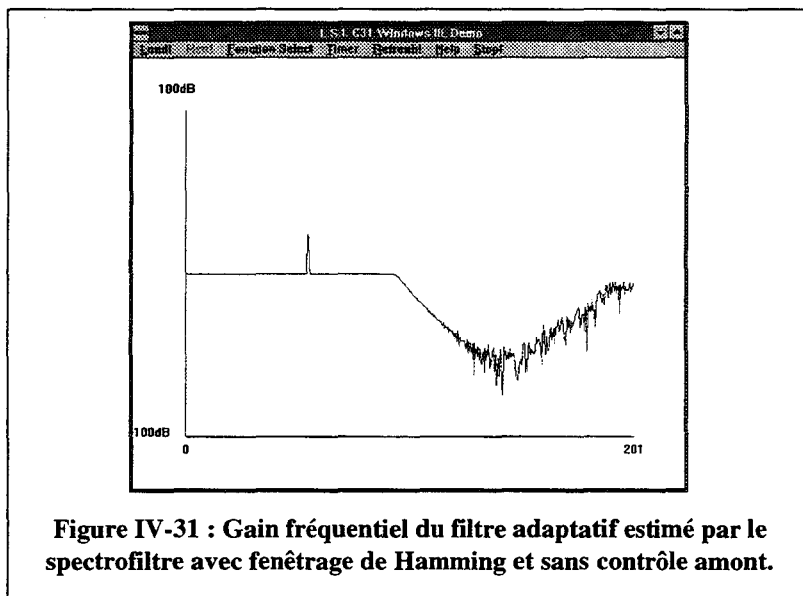


Figure IV-31 : Gain fréquentiel du filtre adaptatif estimé par le spectrofiltre avec fenêtrage de Hamming et sans contrôle amont.

Les résultats du gradient fréquentiel avec ces mêmes paramètres sont similaires. On voit donc que le fenêtrage ne permet pas de résoudre tous les problèmes. Il peut même dégrader les performances s'il est utilisé sans précautions (comparer Figures IV-29 et IV-20). Il convient de l'utiliser en parallèle avec le contrôle amont, comme nous allons le montrer ci-après.

Avec ce même fenêtrage, nous mettons en oeuvre le contrôle amont avec un seuil à 0.5 sur la fonction de cohérence estimée par le DSP (voir Figure IV-32).

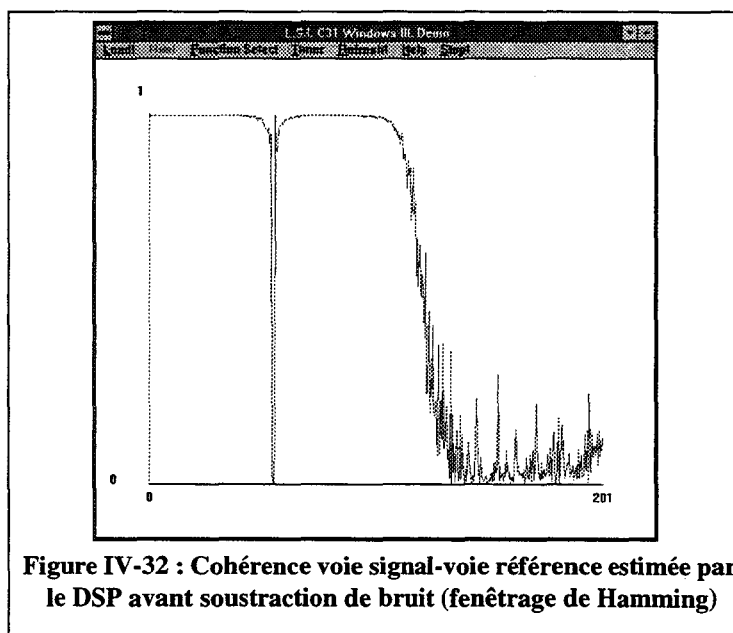


Figure IV-32 : Cohérence voie signal-voie référence estimée par le DSP avant soustraction de bruit (fenêtrage de Hamming)

On constate que la cohérence estimée par le DSP est nettement plus abrupte autour de la raie signal (comparer Figures IV-32 et IV-19). On peut donc s'attendre à un meilleur fonctionnement du contrôle amont ; en effet, on pourra utiliser un seuil assez élevé tout en ayant une bande fréquentielle de non-traitement très réduite, ce que ne permettait pas le fenêtrage rectangulaire. Effectivement, le spectrofiltre adaptatif donne des résultats très intéressants en termes de gain de

traitement (voir Figure IV-33). Au niveau du signal temporel estimé (Figure IV-34), on note la disparition presque complète des battements en fin de bloc (comparer Figures IV-34 et IV-30).

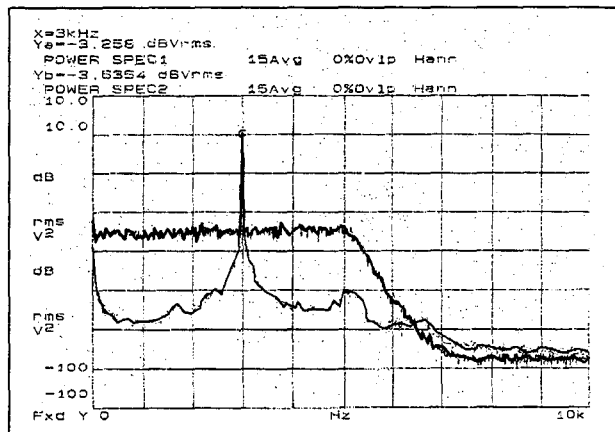


Figure IV-33 : Autospectre voie signal avant (courbe noire) et après traitement (courbe grise) par le spectrofiltre avec fenêtrage de Hamming et contrôle amont à 0.5

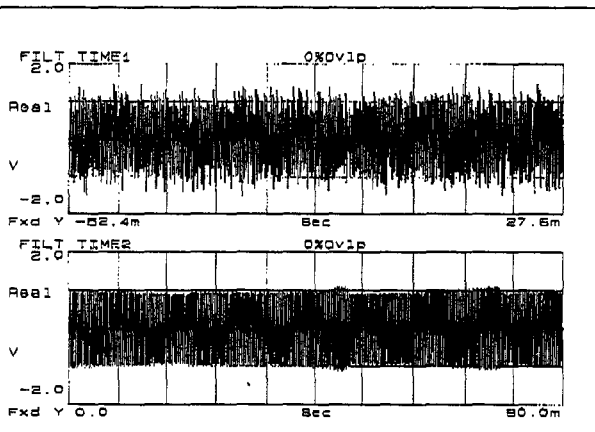


Figure IV-34 : Voie signal avant (courbe du haut) et après traitement (courbe du bas) par le spectrofiltre avec fenêtrage de Hamming et contrôle amont à 0.5

Le gain fréquentiel du filtre adaptatif (Figure IV-35) caractérise mieux le filtre à identifier. La zone fréquentielle de non-traitement au voisinage de la raie signal est nettement plus réduite ; de plus, un nombre plus important de canaux fréquentiels correspondant à la transition du filtre passe-bas est pris en compte dans le traitement (comparer Figures IV-35 et IV-27).

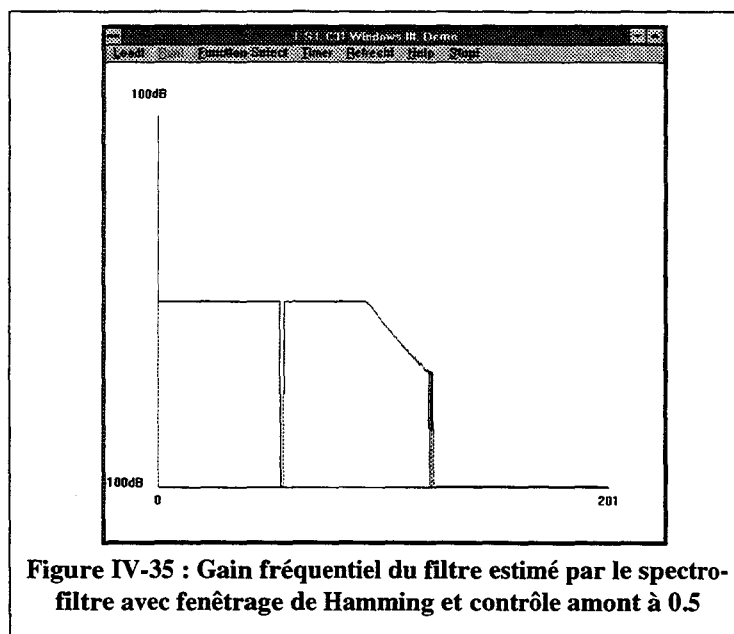


Figure IV-35 : Gain fréquentiel du filtre estimé par le spectrofiltre avec fenêtrage de Hamming et contrôle amont à 0.5

Le contrôle amont a clairement prouvé son utilité (voir Figure IV-36 : comparatif entre les figures IV-29 et IV-33). Nous avons également montré qu'il convient de choisir convenablement le fenêtrage pour éviter d'avoir une large zone de non-traitement sur les canaux fréquentiels de la voie signal pour lesquels la puissance du signal utile est élevée. Pour les signaux à traiter, le fenêtrage de Hamming permet d'utiliser une valeur plus élevée pour le seuil et améliore le fenêtrage rectangulaire, qui nécessitait un compromis sur la valeur du seuil (pas trop fai-

ble pour ne pas prendre en compte les canaux fréquentiels au-dessus de 6.5KHz, et pas trop élevée pour ne pas avoir une trop large zone de non-traitement).

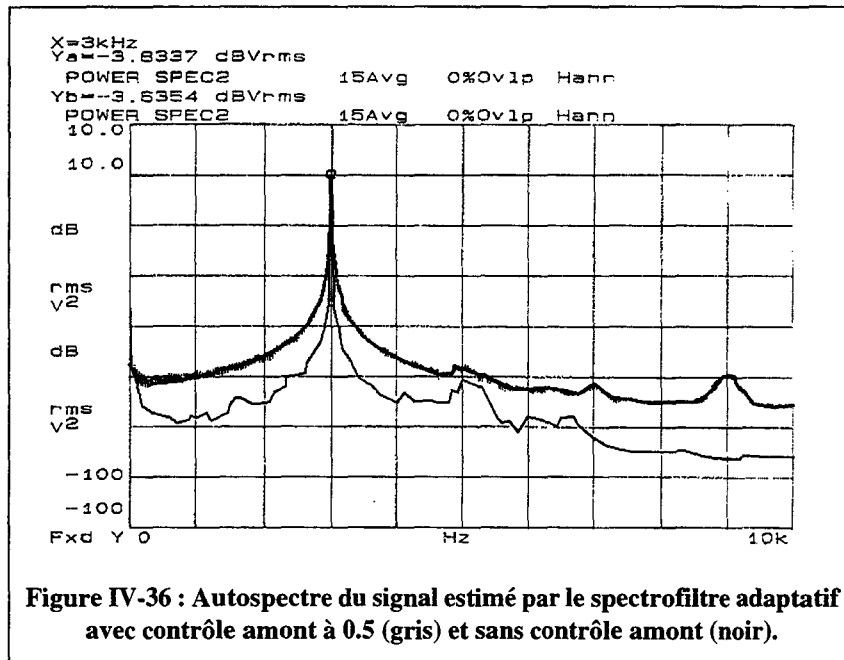


Figure IV-36 : Autospectre du signal estimé par le spectrofiltre adaptatif avec contrôle amont à 0.5 (gris) et sans contrôle amont (noir).

Enfin, on peut remarquer que dans cette situation, il existe une valeur minimale du seuil sous laquelle on risque de prendre en compte les valeurs de la cohérence résultant d'un bruit de calcul (ici pour les fréquences supérieures à 6.5KHz). Cette valeur minimale du seuil se situe autour de 0.4 dans cette application. Il convient également de ne pas choisir la valeur du seuil trop élevée afin de ne pas trop restreindre le support fréquentiel de la soustraction de bruit (par exemple : si on choisit une valeur de 0.9, les canaux fréquentiels correspondant à la transition du filtre passe-bas ne seront pas pris en compte dans le traitement). La détermination de la valeur optimale du seuil n'est donc pas immédiate ; elle dépend de plusieurs paramètres :

- type de fenêtrage utilisé ;
- rapport signal à bruit sur la voie signal (nous le verrons dans le paragraphe suivant).

Pour cette application, une valeur de 0.5 avec le fenêtrage de Hamming donne des résultats satisfaisants. Nous allons maintenant vérifier que la puissance du signal utile joue un rôle dans la détermination de la valeur du seuil à mettre en oeuvre dans le contrôle amont.

IV.3.1.3. INFLUENCE DE LA PUISSANCE DU SIGNAL UTILE

Nous nous sommes placés dans une situation où le rapport signal à bruit sur la voie signal est élevé dans le canal fréquentiel de la raie signal. Nous avons vérifié expérimentalement le phénomène de pollution par la puissance du signal utile et la nécessité dans ce cas d'utiliser un contrôle amont avec un fenêtrage approprié des données. On peut imaginer une autre situation, où le rapport signal à bruit sur la voie signal est faible dans le canal fréquentiel de la raie signal. Dans ce cas, nous serons moins touchés par la pollution due à la puissance du signal utile. Malgré tout, nous verrons que la détermination de la valeur du seuil n'est pas immédiate.

IV.3.1.3.1 Analyse des signaux

La nouvelle configuration de signaux est désormais la suivante : le signal utile est une sinusoïde d'amplitude 25mV et de fréquence 2.975KHz ; le bruit est un bruit blanc d'amplitude maximale 3Volts dans la bande 0-10KHz et le filtre à identifier est le même que précédemment (voir Figures IV-37 et IV-38).

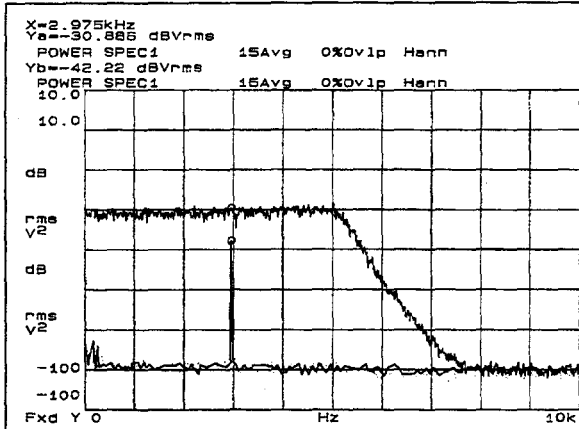


Figure IV-37 : Autospectre du signal bruité (courbe noire) et du signal utile (courbe grise)

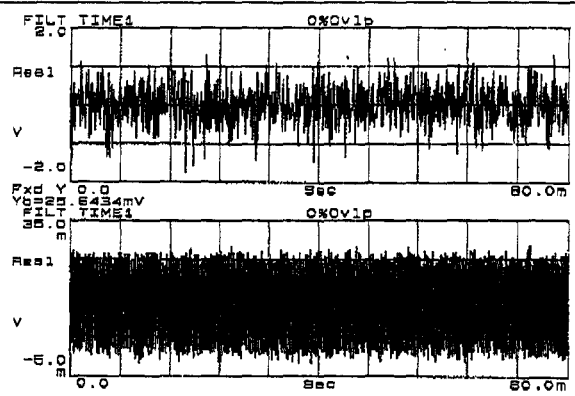


Figure IV-38 : Signal bruité (courbe du haut) et signal utile (courbe du bas)

La cohérence entre les deux voies (voir Figure IV-39) est nettement moins dégradée que précédemment dans le canal fréquentiel de la raie signal, comme on pouvait s'y attendre.

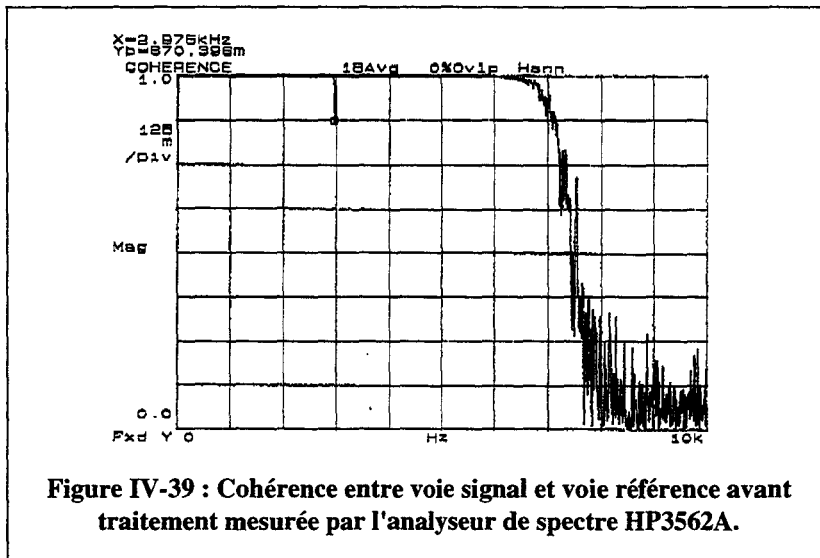


Figure IV-39 : Cohérence entre voie signal et voie référence avant traitement mesurée par l'analyseur de spectre HP3562A.

IV.3.1.3.2 Résultats de la soustraction de bruit

Nous allons maintenant étudier les performances du spectrofiltre et du gradient fréquentiel dans cette situation, ainsi que l'influence de la valeur du seuil choisie pour le contrôle amont. Les paramètres choisis pour les deux algorithmes sont les suivants :

- fréquence d'échantillonnage = 22 KHz ;
- taille des blocs (et des FFT/IFFT) = 1024 échantillons ;

- taux de recouvrement (spectrofiltre/gradient) = $0.4375/0.3125$;
- facteur d'oubli = 0.9 ;
- fenêtrage rectangulaire ;
- contrôle amont (seuil = 0.5) ;
- pas de contrainte du gradient ;
- contrôle aval (facteur de sécurité = 1.0).

Nous avons constaté que dans cette nouvelle situation, le fenêtrage rectangulaire donne de meilleurs résultats que le fenêtrage de Hamming. La fonction de cohérence estimée par le DSP (voir Figure IV-40) est identique à celle mesurée par l'analyseur de spectre.

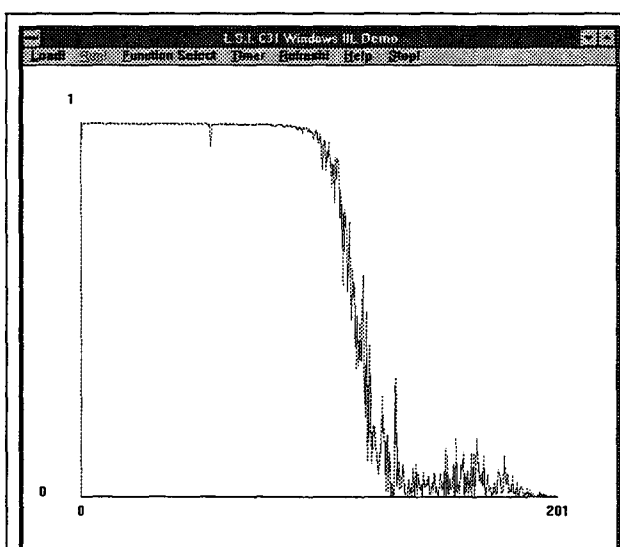


Figure IV-40 : Cohérence estimée par le DSP avant traitement avec fenêtrage rectangulaire

Avec le seuil à 0.5, le spectrofiltre donne de bons résultats en termes de gain de traitement (voir Figure IV-41) et de chute de cohérence (voir Figure IV-42).

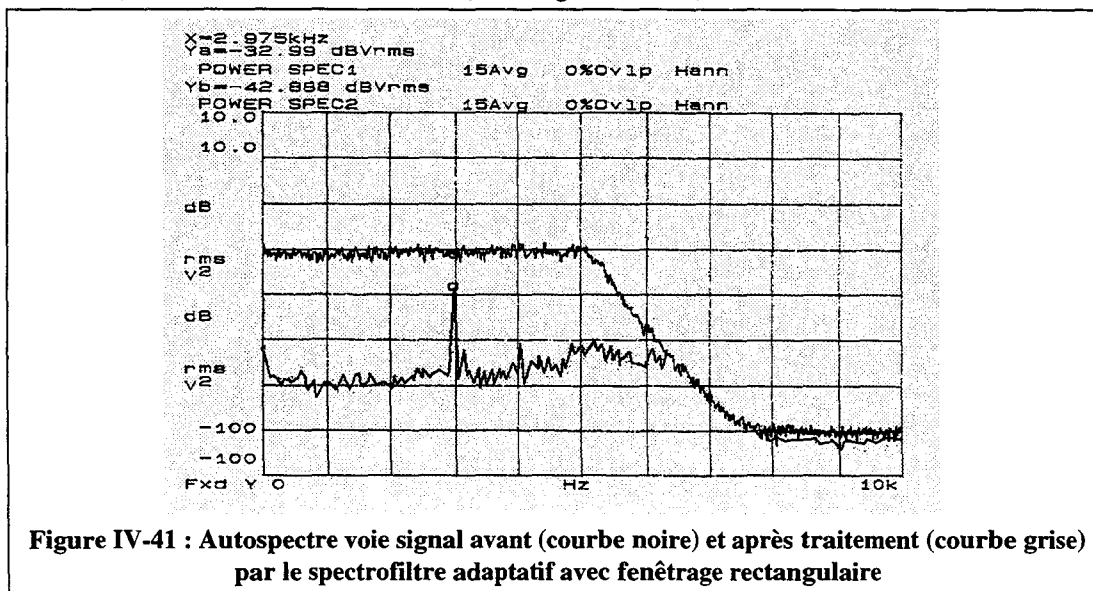
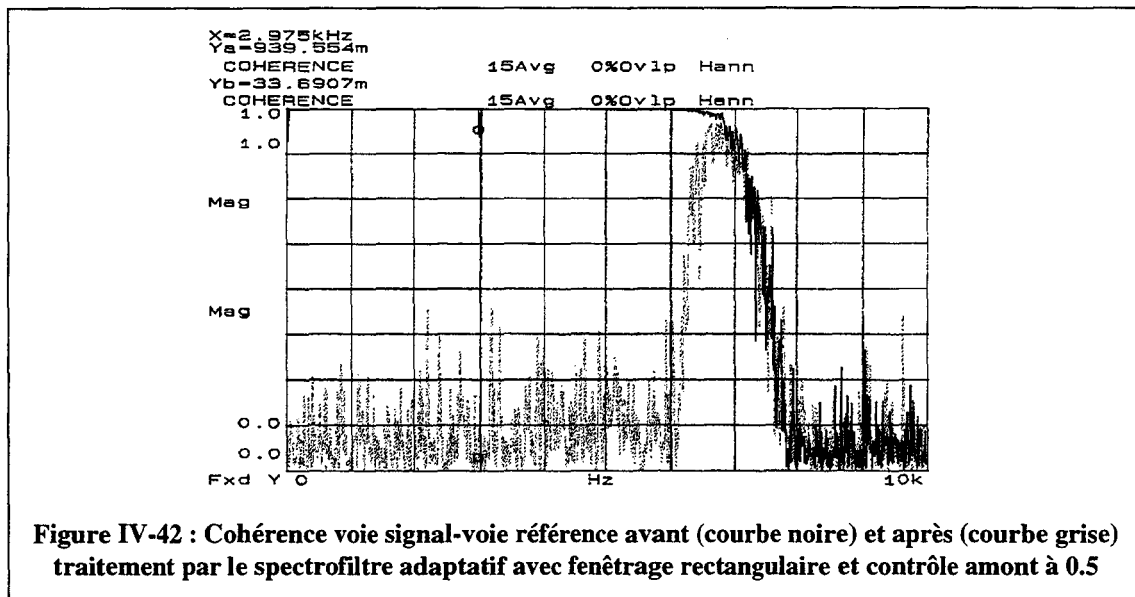
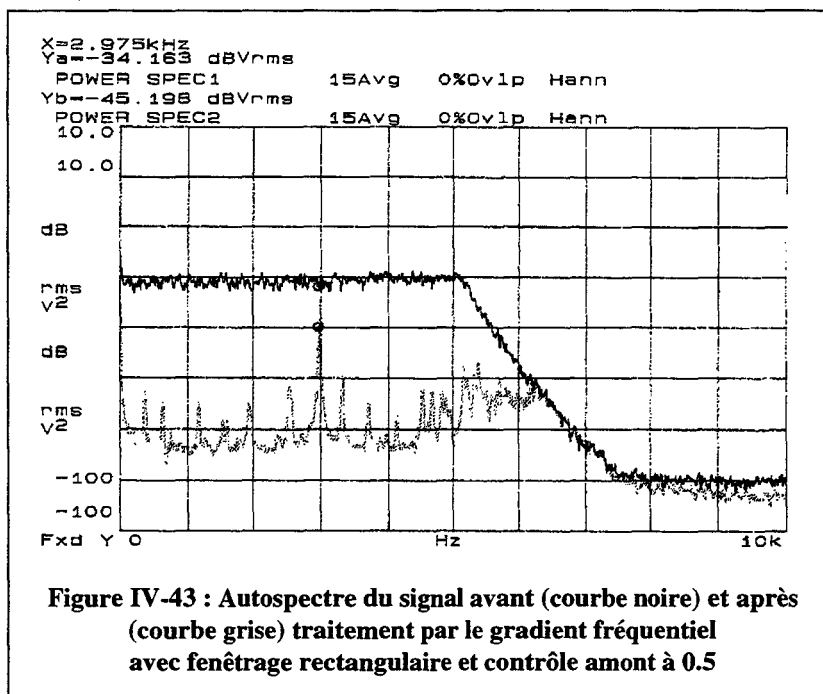


Figure IV-41 : Autospectre voie signal avant (courbe noire) et après traitement (courbe grise) par le spectrofiltre adaptatif avec fenêtrage rectangulaire



On constate que l'autospectre du signal estimé par le gradient fréquentiel est pollué par des raies (voir Figure IV-43).



Nous justifierons la présence de ces raies ultérieurement. Pour le gradient comme pour le spectrofiltre, le choix du seuil à 0.5 permet de forcer le gain fréquentiel du filtre à zéro pour les fréquences supérieures à 6.5KHz, tout en effectuant la soustraction de bruit sur la raie signal. Il convient donc dans cette situation de ne pas choisir une valeur trop élevée du seuil ; dans le cas contraire, la soustraction de bruit n'aura plus lieu sur le canal fréquentiel de la raie signal, et les performances seront nettement dégradées.

Enfin, on peut imaginer une situation intermédiaire, où le rapport signal à bruit dans le canal fréquentiel de la raie signal est proche de 0dB. Dans ce cas, la dégradation de la cohérence

est légèrement plus importante que précédemment. Vu l'analyse théorique réalisée au paragraphe IV.2.5.1, il vaut mieux interdire dans ce cas la soustraction de bruit sur le canal fréquentiel de la raie signal, de manière à ne pas rajouter de bruit de calcul. Il faut donc dans ce cas que la valeur du seuil soit assez élevée pour détecter la dégradation de la cohérence.

En conclusion, le contrôle amont permet d'améliorer nettement les performances de la soustraction de bruit. Encore faut-il choisir convenablement le type de fenêtrage en fonction des signaux et la valeur du seuil en fonction du rapport signal à bruit présent sur la voie signal. Evidemment, l'analyse qui vient d'être faite concerne un seul canal fréquentiel ; en situation réelle, le rapport signal à bruit peut varier d'un canal fréquentiel à l'autre, ce qui complique encore la détermination de la valeur du seuil. On peut quand même tirer qualitativement la conclusion suivante : le seuil doit être assez élevé pour que les valeurs de la cohérence résultant d'un bruit de calcul ne soient pas prises en compte dans le traitement (typiquement : seuil > 0.4) ; mais le seuil ne doit pas être trop élevé pour deux raisons :

- il faut que la soustraction de bruit soit autorisée sur les canaux fréquentiels de la voie signal pour lesquels le rapport signal à bruit est faible (typiquement : seuil < 0.8) ;
- il ne faut pas trop restreindre le support fréquentiel de la soustraction de bruit.

En pratique, on choisira empiriquement la valeur du seuil entre 0.4 et 0.8 en fonction de la configuration expérimentale.

IV.3.1.4. INFLUENCE DU CONTROLE AVAL SUR LA SOUSTRACTION DE BRUIT

Le contrôle aval permet de garantir que la densité spectrale de puissance du signal estimé n'est jamais supérieure à la densité spectrale de puissance de la voie signal. Néanmoins, nous avons constaté que le simple fait d'utiliser ce contrôle peut dégrader les performances de la soustraction de bruit. En particulier, c'est le contrôle aval qui est responsable des raies observées dans l'autospectre du signal estimé par le gradient fréquentiel (voir Figure IV-43).

En effet, le contrôle aval compare pour chaque bloc de signal estimé sa densité spectrale de puissance avec celle du bloc correspondant de la voie signal. Or chaque bloc estimé contient les transitoires dûs à la convolution circulaire, qui se manifestent dans le domaine fréquentiel par des harmoniques multiples de la fréquence de répétition des blocs. Pour ces fréquences, le contrôle aval va forcer l'autospectre du signal estimé à la valeur de l'autospectre de la voie signal, d'où les raies observées dans l'autospectre du signal estimé par le gradient fréquentiel (voir Figure IV-43). Le spectrofiltre n'est pas sensible à cet effet car ce n'est pas un algorithme récursif, à la différence du gradient fréquentiel.

On se place dans la configuration de signaux du paragraphe IV.3.1.3.1 ; la fréquence de la sinusoïde a légèrement glissé et vaut maintenant 3KHz. On garde les mêmes paramètres, mais le contrôle aval est interdit. On obtient l'autospectre du signal estimé par le spectrofiltre

(voir Figure IV-44) et par le gradient fréquentiel (voir Figure IV-45).

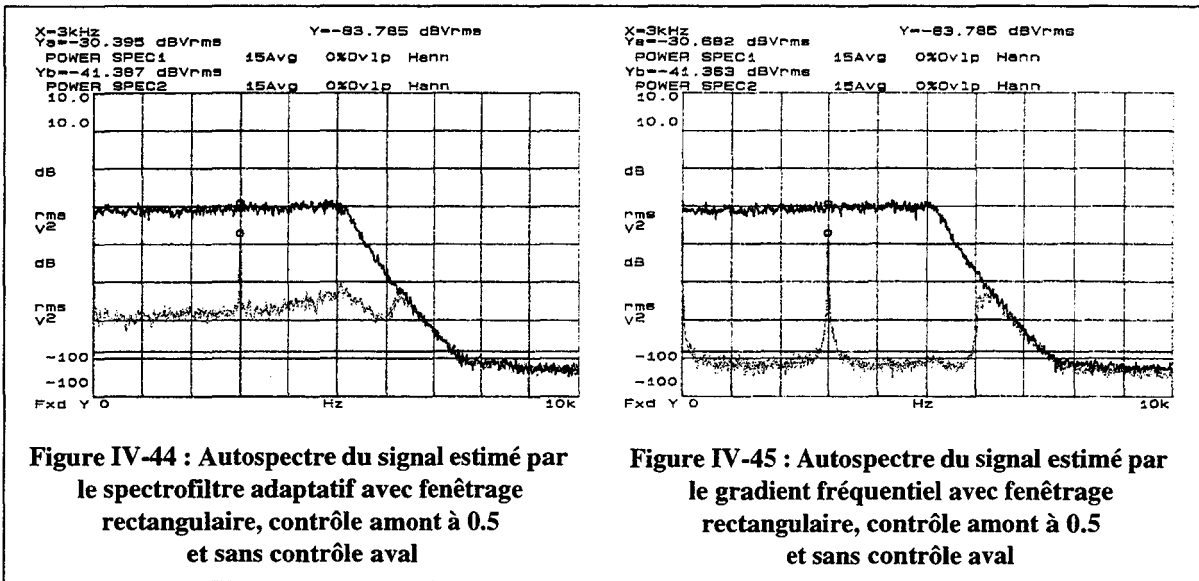


Figure IV-44 : Autospectre du signal estimé par le spectrofiltre adaptatif avec fenêtrage rectangulaire, contrôle amont à 0.5 et sans contrôle aval

Figure IV-45 : Autospectre du signal estimé par le gradient fréquentiel avec fenêtrage rectangulaire, contrôle amont à 0.5 et sans contrôle aval

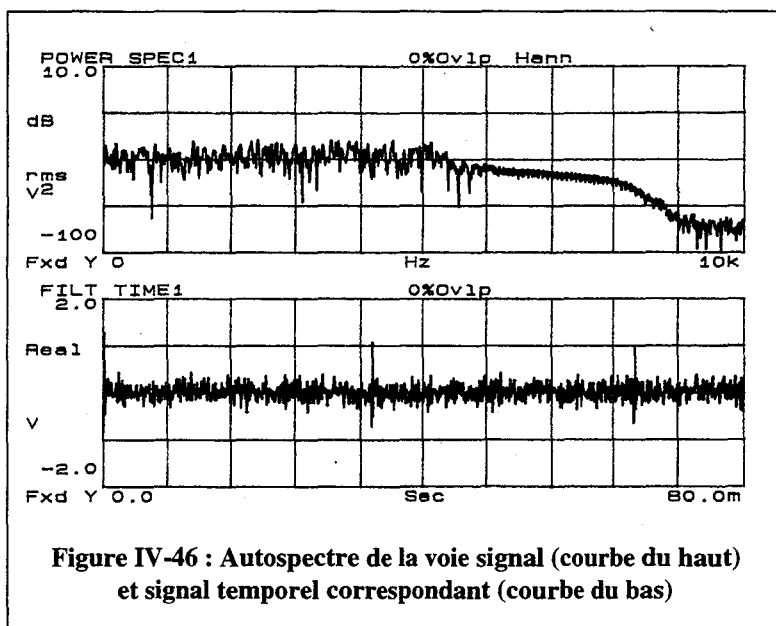
On constate une nette amélioration du gain de traitement pour le gradient fréquentiel (comparer les figures IV-45 et IV-43). Pour le spectrofiltre, l'amélioration est moins visible. Le gradient fréquentiel présente sans le contrôle aval des résultats meilleurs que le spectrofiltre pour cette configuration de signaux.

Le contrôle aval est très utile en théorie, mais si on veut le mettre en oeuvre correctement il faudrait reconstituer chaque bloc du signal estimé sans le transitoire de début de bloc et refaire la FFT au préalable. Cette opération est trop coûteuse à mettre en oeuvre en temps réel, nous n'utiliserons donc pas le contrôle aval.

IV.3.1.5. RECONSTITUTION DU SIGNAL DANS LE DOMAINE TEMPOREL

Au cours de la première expérimentation en cuve acoustique, nous n'étions pas parvenus à retrouver le signal utile dans le domaine temporel, qui était un burst sinusoïdal émis périodiquement. En fait, le traitement de soustraction de bruit déteriorait le signal utile à cause de la puissance de ce signal et par le fait qu'on n'utilisait aucun contrôle amont sur la cohérence. L'objectif de ce paragraphe est de montrer que les algorithmes de soustraction de bruit permettent désormais de retrouver exactement le signal utile dans le domaine temporel.

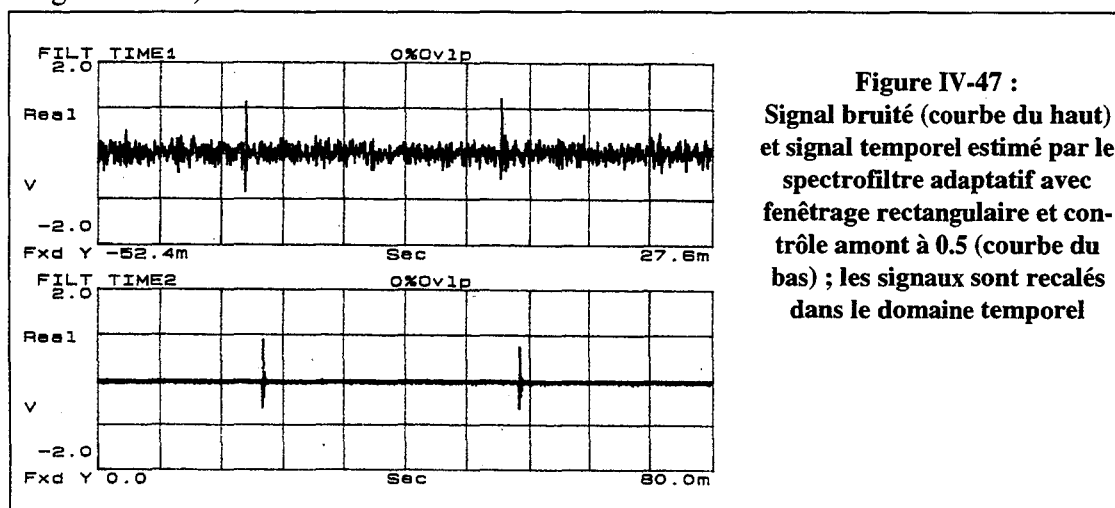
Nous avons choisi comme signal utile un burst sinusoïdal (1 cycle) d'amplitude 1Volt, de fréquence 5KHz (fréquence de répétition = 30Hz). Le bruit perturbateur et le filtre à identifier sont les mêmes que précédemment (amplitude maximale du bruit blanc = 1Volt). On peut visualiser l'autospectre de la voie signal (voir Figure IV-46).



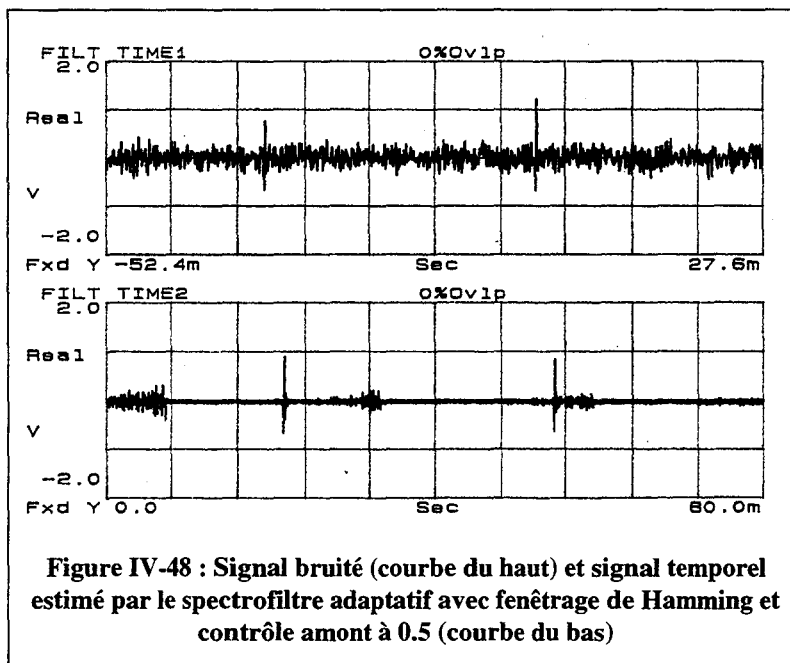
Les paramètres utilisés par les algorithmes de soustraction de bruit sont les suivants :

- fréquence d'échantillonnage = 22 KHz ;
- taille des blocs (et des FFT/IFFT) = 1024 échantillons ;
- taux de recouvrement (spectrofiltre/gradient) = 0.4375/0.3125 ;
- facteur d'oubli = 0.9 ;
- fenêtrage rectangulaire ;
- contrôle amont (seuil = 0.5) ;
- pas de contrainte du gradient ;
- contrôle aval (facteur de sécurité = 1.0).

Le signal estimé par le spectrofiltre et le gradient fréquentiel est presque identique au signal utile (voir Figure IV-47).



L'utilisation du fenêtrage de Hamming avec les mêmes paramètres pour cette configuration de signaux dégrade le traitement pour les données de fin de bloc (voir Figure IV-48), que ce soit avec l'algorithme du spectrofiltre ou celui du gradient.



Cette dernière validation montre que le contrôle amont sur la fonction de cohérence permet de retrouver le signal utile dans le domaine temporel. Elle nous permet également de préciser les idées concernant le fenêtrage des données : le fenêtrage de Hamming présente des avantages indéniables par rapport au fenêtrage naturel ; en revanche, les échantillons de fin de bloc subissent une dégradation du traitement due à la pondération plus faible (la fenêtre de Hamming vaut 0.08 sur les bords), et au fenêtrage inverse en fin de traitement qui amplifie les erreurs de calculs sur ces échantillons.

IV.3.2. SECONDE VALIDATION EN BASSIN D'ESSAIS ACOUSTIQUES

IV.3.2.1. DISPOSITIF EXPERIMENTAL

Le dispositif expérimental est exactement le même que pour la première validation en bassin d'essais acoustiques. Nous avons simplement choisi un projecteur acoustique dont la fréquence de résonance est nettement plus basse (3KHz), de manière à pouvoir utiliser une fréquence d'échantillonnage compatible avec les modifications algorithmiques envisagées au paragraphe IV.2.5.2. De plus, une fréquence d'échantillonnage plus basse nous permet de nous rapprocher de la réalité physique des phénomènes.

IV.3.2.2. INSTRUMENTATION UTILISEE

Pour l'émission du signal utile sur le projecteur acoustique, l'instrumentation utilisée est la même que lors de la première validation (voir paragraphe IV.2.2).

Pour l'émission du bruit sur le pot vibrant, nous avons utilisé le générateur de bruit

blanc pseudo-aléatoire de l'analyseur de spectre HP3562A. Le signal est un bruit blanc stationnaire dans la bande 0-10KHz. Avant d'attaquer le pot vibrant (qui est le même que lors de la première expérimentation), ce signal passe dans un filtre passe-haut (multichannel filter 3905A). En parallèle avec le générateur de bruit blanc, nous utilisons cette fois le générateur de fonctions HP33120A pour injecter des non-linéarités. Le signal fourni par ce générateur de fonctions est une sinusoïde de fréquence 500Hz. Les deux signaux sont sommés et passent ensuite dans un amplificateur de puissance avant d'attaquer le pot vibrant.

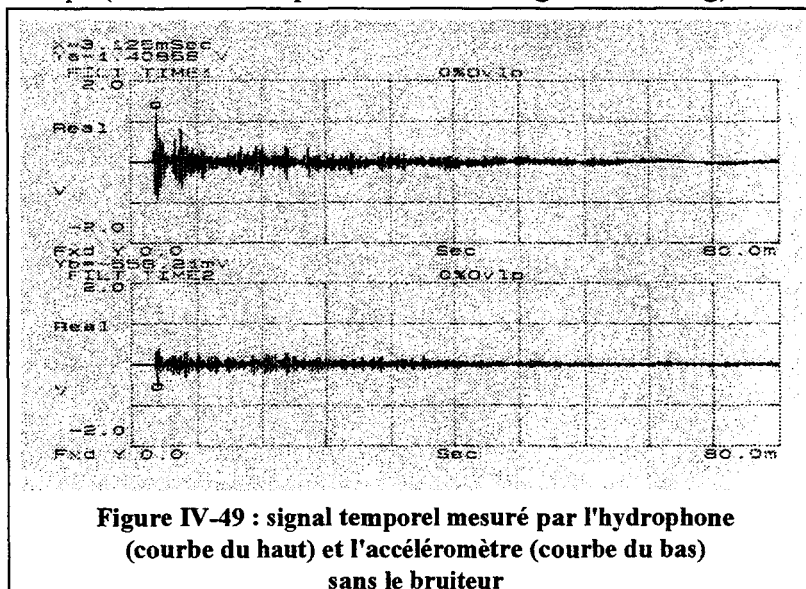
Concernant les voies de réception, l'instrumentation utilisée est la même que lors de la première expérimentation (voir Figures IV-4 et IV-5).

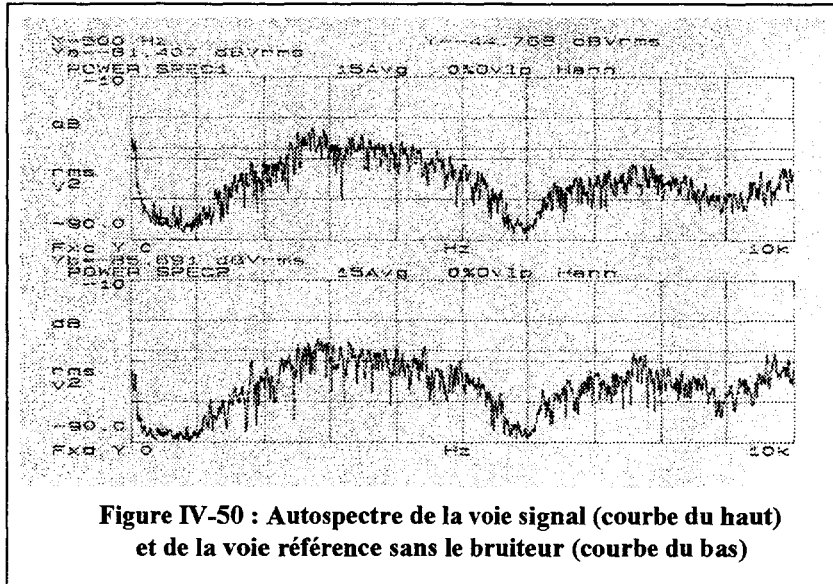
IV.3.2.3. RESULTATS OBTENUS EN SOUSTRATION DE BRUIT

Avec la configuration expérimentale présentée au paragraphe précédent, nous avons testé les algorithmes du spectrofiltre adaptatif, du gradient fréquentiel et du LMS temporel. Nous avons en particulier pu étudier l'influence du contrôle amont sur les résultats de soustraction de bruit obtenus par les méthodes fréquentielles. En ce qui concerne l'algorithme LMS temporel, l'utilisation d'une fréquence d'échantillonnage plus faible (22KHz) nous a permis notamment de mettre en oeuvre un filtre adaptatif de taille plus importante en temps réel, et de mettre en évidence une amélioration significative des résultats par rapport à la première expérimentation.

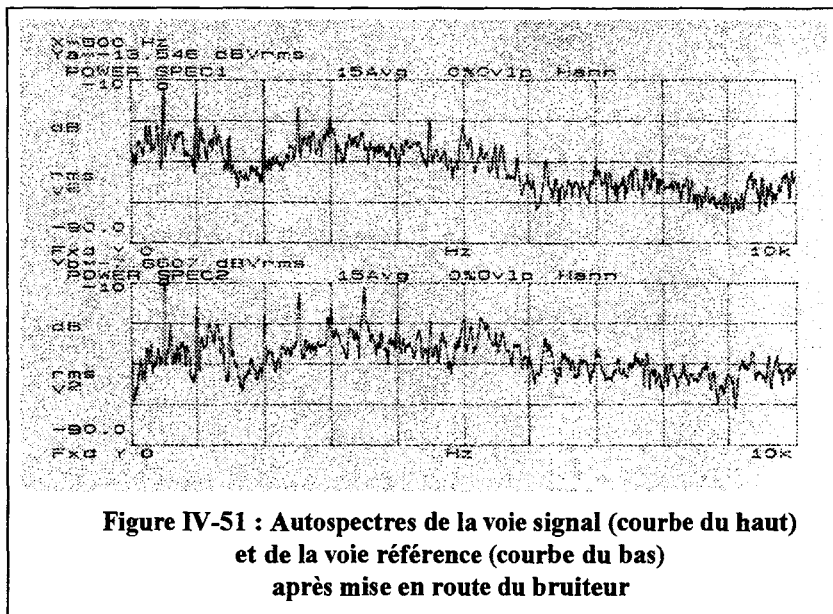
IV.3.2.3.1 Analyse des signaux

Le signal utile injecté sur le projecteur acoustique est un burst sinusoïdal de fréquence 3KHz. En l'absence du bruiteur, ce signal est bien détecté par l'hydrophone et l'accéléromètre (voir Figure IV-49). On remarquera les échos sur les parois du bassin. A la différence de la première expérimentation (voir paragraphe IV.2.3), le support fréquentiel du signal utile se situe désormais entre 1 et 9KHz (voir Figure IV-50). L'analyse spectrale est faite par périodogramme moyenné sur 15 coups (FFT sur 1024 points avec fenêtrage de Hanning).

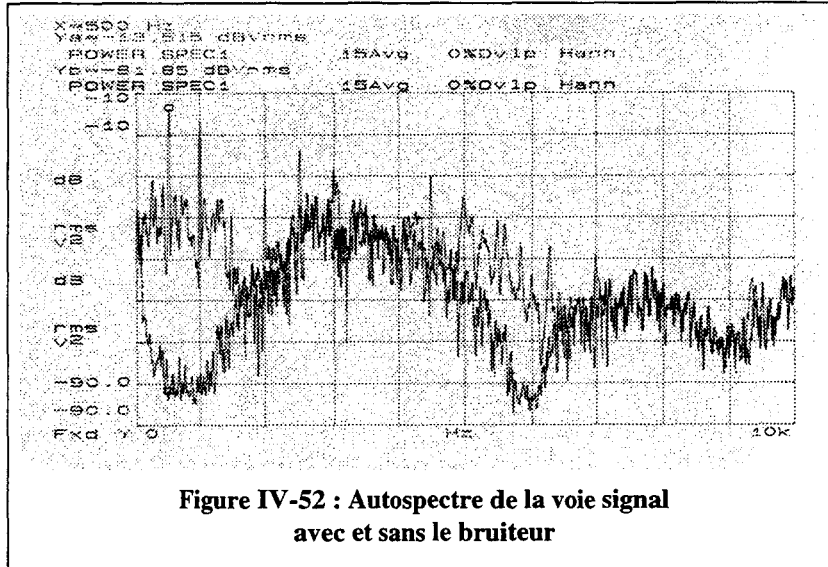




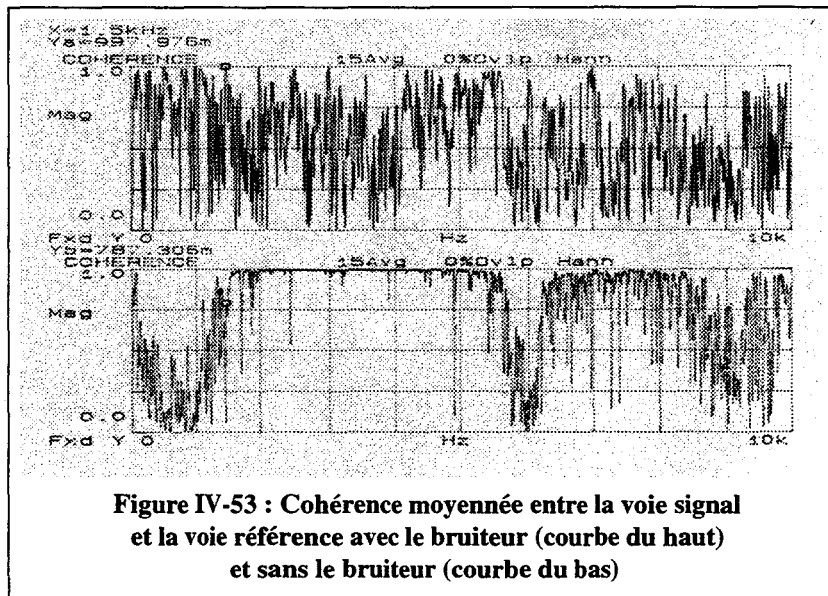
Lors de la mise en route du bruiteur, le signal utile est noyé dans un bruit qui se manifeste essentiellement sous forme de raies dans l'autospectre de la voie signal et de la référence bruit (voir Figure IV-51). On remarque dans ces spectres les harmoniques de la fréquence fondamentale injectée par le générateur de fonctions HP33120A sur le pot vibrant (500Hz).



Ces raies ont une amplitude nettement supérieure au niveau spectral du signal utile (voir Figure IV-52). On se retrouve dans une situation analogue à la première expérimentation : le rapport signal à bruit sur la voie signal est proche de 0dB pour certains canaux fréquentiels.



A l'examen de la fonction de cohérence entre la voie signal et la voie référence sans le bruiteur (voir Figure IV-53), on constate de nouveau que l'hydrophone et l'accéléromètre sont très corrélés sur le signal utile. Néanmoins, la cohérence est également très élevée sur les raies de bruit (harmoniques du fondamental à 500Hz). On constate une nette dégradation de cette cohérence au voisinage des canaux fréquentiels pour lesquels le rapport signal à bruit est élevé sur la voie signal, d'où l'intérêt de mettre en oeuvre le contrôle amont pour éviter de dégrader le signal utile comme dans la première expérimentation.



IV.3.2.3.2 Algorithme du spectrofiltre adaptatif

Dans cette seconde expérimentation, l'algorithme du spectrofiltre adaptatif a été mis en oeuvre avec toutes les fonctionnalités décrites au chapitre II, notamment le contrôle amont du traitement. La fréquence d'échantillonnage utilisée est de 22KHz, la taille des tranches est de 1024 points avec un taux de recouvrement de 50%. Le facteur d'oubli vaut 0.9 ; nous avons mis

en oeuvre un fenêtrage naturel (rectangulaire) sur les tranches. Dans un premier temps, le contrôle amont est mis en oeuvre avec un seuil à 0.25 sur la fonction de cohérence.

On constate un gain de traitement positif sur les raies de bruit (harmoniques du fondamental à 500Hz), associé à une diminution significative de la cohérence entre la voie signal et la référence bruit sur ces raies (voir Figures IV-54 et IV-55).

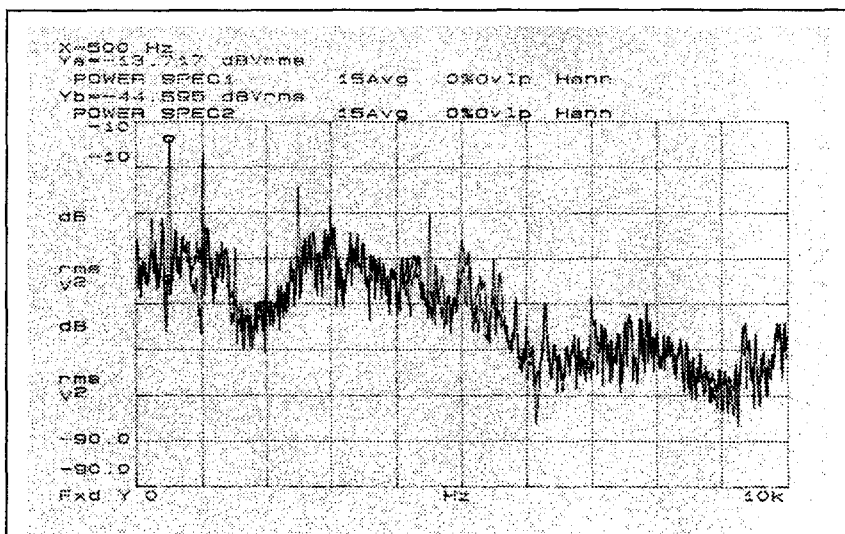


Figure IV-54 : Autospectre de la voie signal avant et après traitement de soustraction de bruit par le spectrofiltre adaptatif

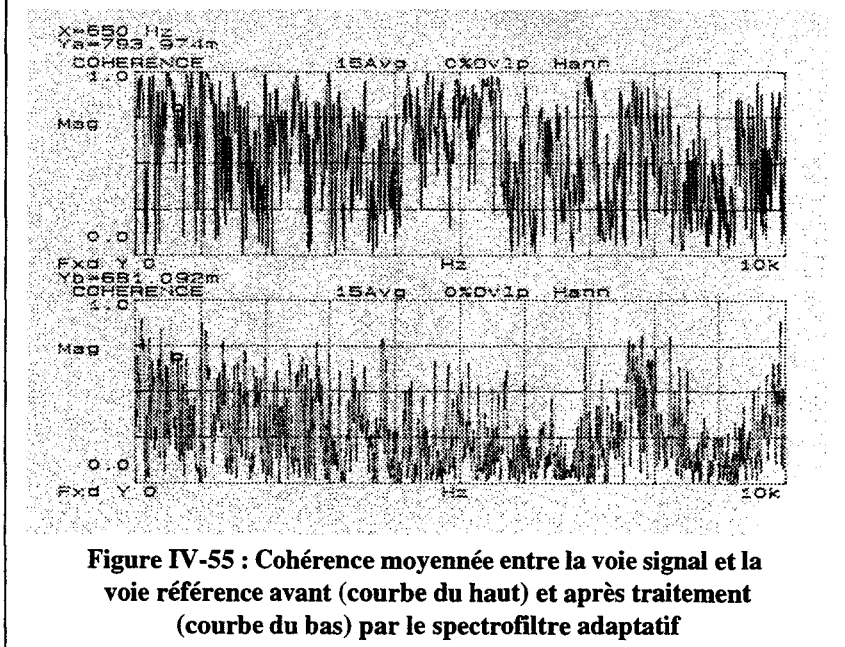


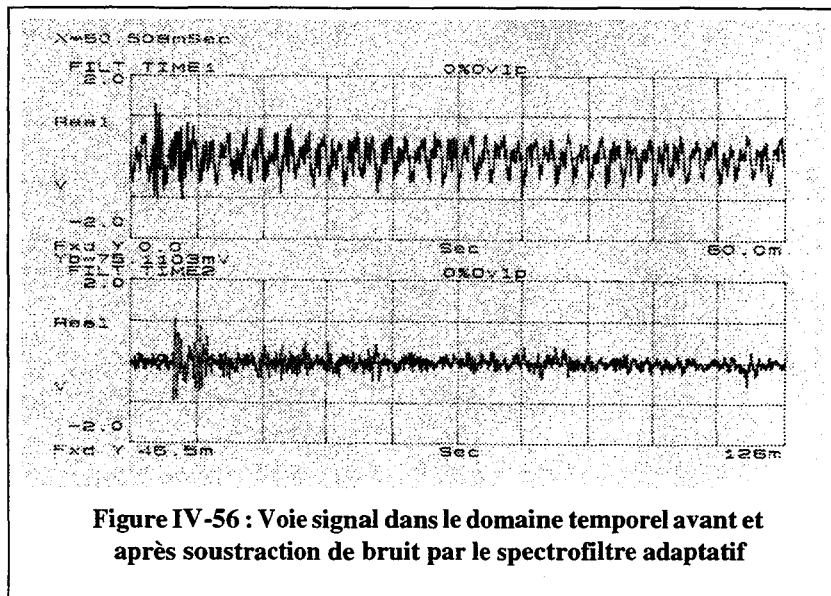
Figure IV-55 : Cohérence moyennée entre la voie signal et la voie référence avant (courbe du haut) et après traitement (courbe du bas) par le spectrofiltre adaptatif

Le tableau suivant met en évidence le gain de traitement et la chute de cohérence associée sur les principales raies de bruit identifiées dans l'autospectre de la voie signal.

Table 11: Gain de traitement et chute de cohérence sur les principales raies de bruit pour les algorithmes mis en oeuvre lors de la 2ème expérimentation : (1) = Spectrofiltre adaptatif avec seuil = 0.25 (contrôle amont) ; (2) = Spectrofiltre adaptatif avec seuil = 0.75 ; (3) = Gradient fréquentiel avec seuil = 0.25 ; (4) = LMS temporel (M = 170, $\mu = 0.001$)

Fréquence (Hz)	Gain de traitement (dB)				Chûte de cohérence			
	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)
500	30.9	25.4	32.8	45	0.37	0.15	0.88	0.94
1000	19.4	18.8	31.8	35.3	0.125	0.08	0.87	0.45
1500	11.2	5.8	20.6	14.1	0.37	0.05	0.25	0.36
2000	11	16.4	11.8	15.9	0.33	0.75	0.25	0.62
2500	14.1	12.3	14.7	20.6	0.39	0.39	0.33	0.37
3000	14	16.5	20	16	0.28	0.59	0.42	0.31
4500	20.6	15.9	16.5	16	0.81	0.34	0.72	0.30
5000	7	3	8.2	2.3	0.62	0.64	0.58	0.05
5500	8.2	7	8.5	5	0.47	0.25	0.72	0.02
5600	10	10.6	10.6	6	0.53	0.70	0.70	0.08
5800	1.2	0	1	0	0.14	0.16	0.47	-0.19
6000	2.4	0	3	0	0.39	0.15	0.47	0.34
6270	0	0	0	0	0.51	0.10	0.47	0.23
7000	14.7	11.8	20.6	17	0.61	0.16	0.90	0.08

Dans le domaine temporel, cela se traduit par une soustraction de bruit. On retrouve le signal utile (burst sinusoïdal) après prise en compte du retard introduit par le traitement par blocs (voir Figure IV-56).



Le passage du seuil sur la fonction de cohérence de 0.25 à 0.75 a très peu d'influence sur les résultats (voir le tableau précédent). En effet, une valeur de 0.25 est ici suffisante pour détecter les canaux fréquentiels où le rapport signal à bruit est élevé et interdire le traitement de soustraction de bruit sur ces canaux. Le signal utile est ainsi préservé.

IV.3.2.3.3 Algorithme du gradient fréquentiel

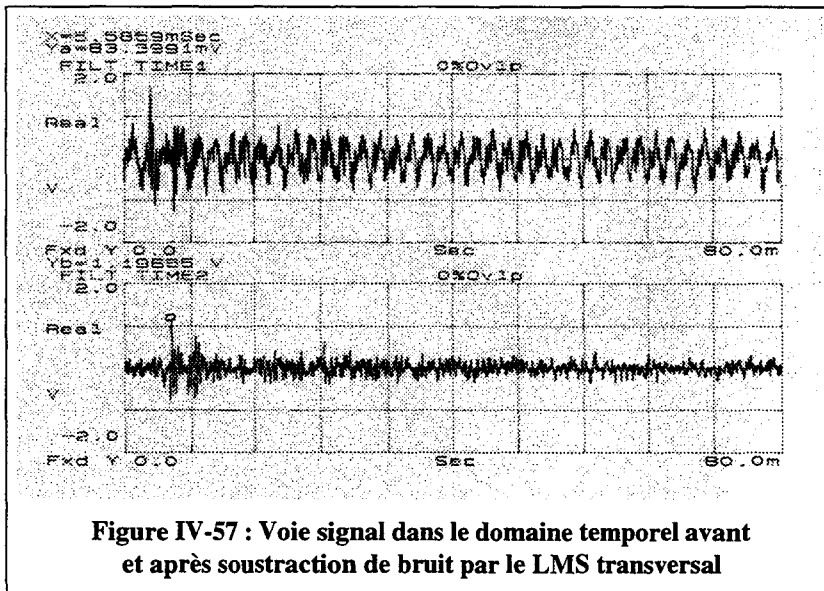
Les paramètres utilisés pour le gradient fréquentiel sont identiques à ceux du spectrofiltre adaptatif, excepté pour le taux de recouvrement, qui est ici de 40.625% (taux maximal autorisé avec utilisation de l'interface graphique et sans mise en oeuvre de la contrainte du gradient). Le seuil utilisé pour le contrôle amont est de 0.25 : nous avons vu au paragraphe précédent que cette valeur est suffisante pour préserver le signal utile.

Les résultats en termes de gain de traitement et de chute de cohérence sont comparables à ceux atteints par le spectrofiltre (voir Tableau 11) sur les raies de bruit. Les résultats sur le fondamental et le premier harmonique (1KHz) sont meilleurs que pour le spectrofiltre adaptatif.

IV.3.2.3.4 Algorithme LMS temporel

Grâce à l'utilisation d'une fréquence d'échantillonnage plus basse par rapport à la première expérimentation, nous avons pu mettre en oeuvre en temps réel un filtre adaptatif transversal plus long (170 coefficients) dans l'algorithme LMS temporel.

Les résultats obtenus par cet algorithme sont très bons (voir Tableau 11), et permettent notamment de retrouver le signal utile dans le domaine temporel (voir Figure IV-57). Le pas d'adaptation μ a été fixé dans un premier temps à 0.001.



Le gain de traitement sur les principales raies de bruit est significatif (voir Tableau 11). Les résultats obtenus par cet algorithme sont meilleurs que les résultats atteints par les méthodes fréquentielles sur les deux premières raies de bruit (500Hz et 1KHz).

Le passage du pas d'adaptation de 0.001 à 0.01 crée un phénomène de creusement du spectre de bruit large bande sur la raie du fondamental à 500Hz. On observe également un phénomène de creusement du spectre de signal utile (voir Figure IV-58) autour des raies correspondant aux harmoniques 4 et 6 (2.5KHz et 3.5KHz).

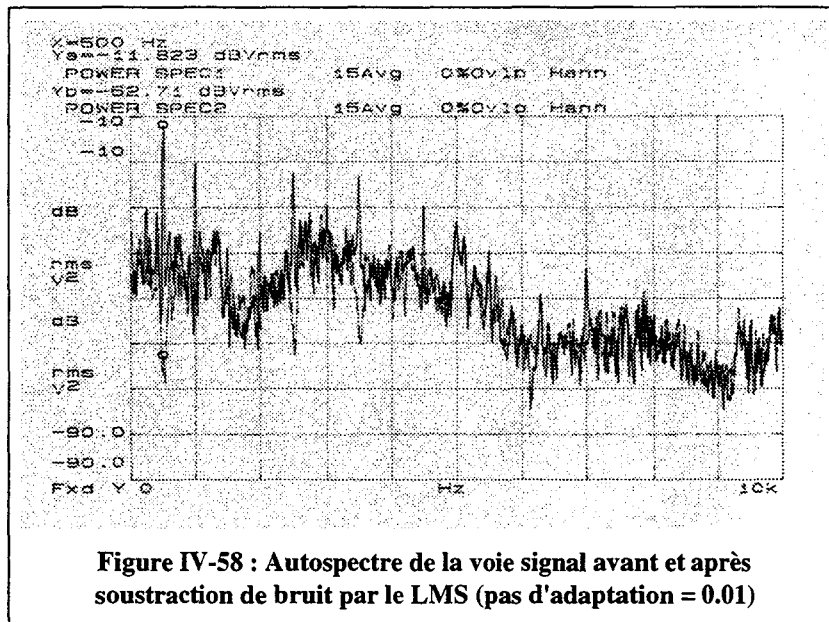


Figure IV-58 : Autospectre de la voie signal avant et après soustraction de bruit par le LMS (pas d'adaptation = 0.01)

Ce phénomène de creusement n'induit pas de dégradation trop sérieuse du signal utile. Néanmoins, nous venons de mettre une fois de plus en évidence la sensibilité du LMS temporel par rapport à ses paramètres, notamment le pas d'adaptation (ou facteur de convergence).

IV.3.2.3.5 Influence de la puissance du signal utile

Pour étudier l'influence de la puissance du signal utile sur la soustraction de bruit, nous avons fait varier l'atténuation de l'amplificateur de puissance Bruel&Kjaer 2706. De cette manière, nous parvenons à injecter des niveaux de bruit différents sur le pot vibrant.

L'algorithme testé est le spectrofiltre adaptatif ; ses paramètres sont les mêmes que dans le paragraphe IV.3.2.3.2. Le contrôle amont n'est pas mis en oeuvre (i.e. le seuil utilisé sur la fonction de cohérence vaut 0). Pour trois valeurs différentes de l'atténuation du 2706 (10dB, 20dB et 40dB), nous pouvons examiner la voie signal dans le domaine temporel avant et après soustraction de bruit par le spectrofiltre (voir Figures IV-59, IV-60 et IV-61).

En améliorant progressivement le rapport signal à bruit sur la voie signal, on arrive à un seuil à partir duquel on soustrait du signal utile. Ceci prouve que les estimateurs spectraux utilisés ne sont pas parfaits et met en évidence de manière indiscutable l'intérêt du contrôle amont sur la fonction de cohérence.

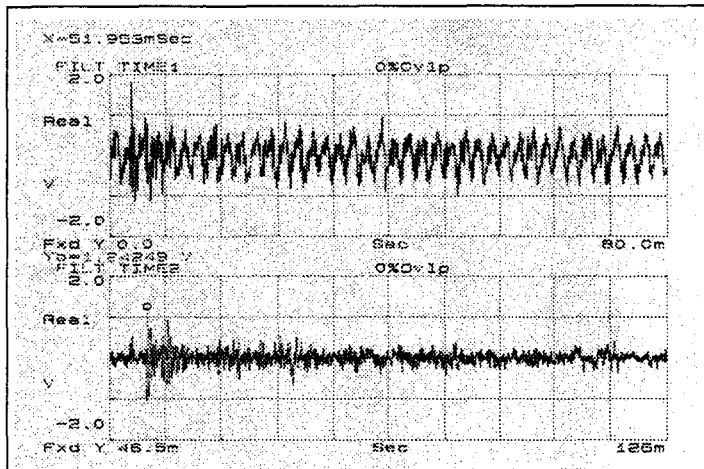


Figure IV-59 : Voie signal dans le domaine temporel avant et après traitement par le spectrofiltre adaptatif (Atténuation = 10dB)

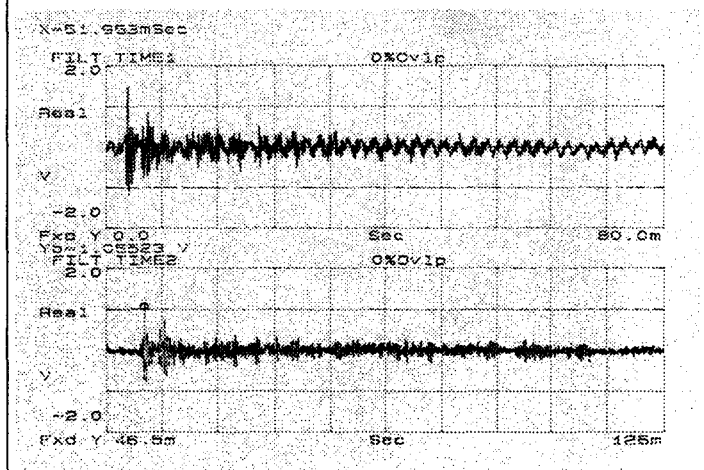


Figure IV-60 : Voie signal dans le domaine temporel avant et après traitement par le spectrofiltre adaptatif (Atténuation = 20dB)

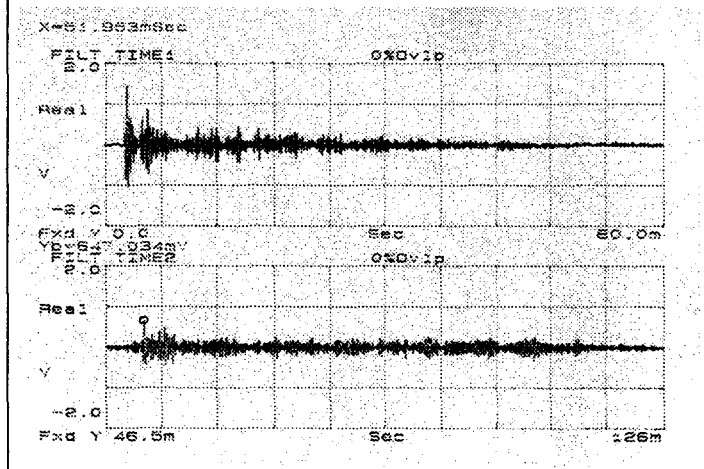


Figure IV-61 : Voie signal dans le domaine temporel avant et après traitement par le spectrofiltre adaptatif (Atténuation = 40dB)

IV.4. CONCLUSION

Une validation expérimentale du fonctionnement des algorithmes de soustraction de bruit LMS et spectrofiltre/gradient fréquentiel a été menée en temps réel sur une structure mécanique relativement proche de la réalité physique. Les résultats obtenus par ces algorithmes sont tout à fait satisfaisants en termes de gain de traitement et de chute de cohérence entre la

voie signal et la voie référence. Nous avons montré que la mise en oeuvre du contrôle amont sur la fonction de cohérence est importante pour pouvoir traiter les situations où le rapport signal à bruit sur la voie signal est élevé sans dégrader le signal utile.

Il est à noter que le dispositif expérimental a rendu plus difficile la tâche de l'algorithme LMS temporel, car la réponse impulsionnelle du filtre à identifier a été allongée par les échos sur les parois du bassin.

IV.5. BIBLIOGRAPHIE

[Servièrè 89]

SERVIÈRE C., Eléments de comparaison entre différentes méthodes de soustraction de bruit, Thèse de doctorat INPG, Mai 1989.

CONCLUSION

L'objectif de ce travail était d'étudier la faisabilité d'un dispositif permettant de réduire en temps réel le bruit propre d'origine mécanique sur les antennes de réception d'un sonar passif. L'aspect temps réel de la réduction de bruit est très important, car dans le cas des systèmes sonar il faut détecter le plus tôt possible un éventuel ennemi. La solution envisagée dans cette étude consiste à utiliser des capteurs intelligents permettant une réduction locale du bruit propre, c'est-à-dire permettant d'épurer directement le signal mesuré par un hydrophone de sa composante de bruit propre.

Après avoir mis au point un modèle simple de plaque en vibration permettant de représenter la structure mécanique supportant l'antenne sonar, nous avons réalisé une synthèse des différentes méthodes de soustraction de bruit existant dans la littérature. En particulier, nous avons tenté de trouver une démarche unifiée permettant de comprendre l'obtention des algorithmes adaptatifs.

Ces algorithmes ont ensuite été simulés dans un langage haut niveau (C) sur des signaux expérimentaux, ainsi que sur les signaux issus de la modélisation effectuée dans le premier chapitre. Nous avons alors opéré une première sélection parmi ces algorithmes, sur la base de leurs performances obtenues en simulation, et de la faisabilité de leur mise en oeuvre en temps réel sur un processeur de traitement du signal standard à faible coût.

Le choix du processeur de traitement du signal sur lequel nous implémentons ces algorithmes est ensuite justifié, et nous présentons la mise en oeuvre sur ce processeur des algorithmes sélectionnés respectant la contrainte temps réel de notre application (LMS dans le domaine temporel et Spectrofiltre dans le domaine fréquentiel).

Pour terminer, une validation en temps réel des algorithmes a été menée en bassin d'essais acoustiques sur une structure mécanique relativement proche de la réalité physique. Les résultats des algorithmes sont présentés en termes de gain de traitement et de chute de cohérence entre la voie signal et la voie référence. Les modifications algorithmiques à apporter en vue d'une amélioration des résultats sont présentées. Enfin, une seconde validation des algorithmes incorporant les modifications précédentes est menée en bassin d'essais acoustiques, sur la même structure physique. Les résultats obtenus en soustraction de bruit propre lors de cette seconde validation s'avèrent très satisfaisants, et nous permettent de conclure sur la faisabilité de la mise en oeuvre en temps réel des algorithmes de soustraction de bruit LMS et Spectrofiltre, appliqués au bruit propre d'origine mécanique sur les capteurs d'antennes sonar.

Dans une étape suivante, il serait intéressant d'étudier les problèmes associés à l'application de ces méthodes de soustraction de bruit pour des signaux et des systèmes plus proches de la réalité, en procédant par exemple à des essais en lac sur un tronçon de coque de sous-marin immergé et excité en vibration.

Par ailleurs, nous nous sommes intéressés dans cette étude au traitement d'un seul doublet de capteurs hydrophone-accéléromètre ; nous avons montré que le traitement de soustraction de bruit sur ce dispositif de capteurs nécessite un DSP avec une puissance de calculs minimale de 40 MFLOPS. L'évolution très rapide de la puissance de calcul des DSP permet aujourd'hui d'ouvrir des perspectives intéressantes sur la mise en oeuvre en temps réel d'algorithmes de soustraction de bruit plus complexes (tels que RLS-rapides ou Treillis), ou encore sur la mise en oeuvre en temps réel d'un système de soustraction de bruit multi-références [Guerre-Chaley 90]. En effet, on trouve maintenant des DSP à virgule flottante présentant un temps de cycle de 5-6 nsec, avec une architecture permettant une puissance de calculs de 1 GFLOPS (valeur pic), tels que le TMS320C67x de Texas Instruments.

Ainsi, une validation en temps réel des algorithmes éliminés dans cette étude est aujourd'hui possible ; menée sur un système plus proche de la réalité (essais en lac), une telle validation permettrait notamment de trouver le compromis optimal entre performances et coût de calculs pour la solution envisagée dans cette étude.

On peut également envisager la mise en oeuvre en temps réel d'un système de soustraction de bruit multi-références [Guerre-Chaley 90], afin d'effectuer une étude paramétrique des performances de la soustraction de bruit en fonction du positionnement des capteurs de référence-bruit. Une telle étude, menée sur un système plus proche de la réalité (essais en lac), permettrait de comparer la solution envisagée dans cette thèse et une solution plus traditionnelle, qui consiste à disposer les capteurs de référence-bruit sur les machines les plus bruyantes. Pour une antenne opérationnelle donnée, on pourrait alors déterminer le dispositif de capteurs optimal nécessaire au traitement de soustraction de bruit, et le coût de calculs associé.

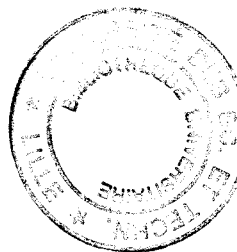
Enfin, l'utilisation d'un processeur tel que le TMS320C67x permettrait de mettre en oeuvre en temps réel un traitement de soustraction de bruit multi-voies, ce qui minimiserait la complexité -et donc le coût- du système de soustraction de bruit nécessaire au traitement d'une antenne sonar opérationnelle.

REFERENCES BIBLIOGRAPHIQUES

- [Arques 82] ARQUES P.Y., FAUCON G., Proposal and experimental evaluation of a combined structure correlofilter-adapter, ICASSP 1982, Paris.
- [Audoly et Giangreco 92] AUDOLY C., GIANGRECO C., A study of the output of a hydrophone fixed on a vibrating structure, Journal de Physique IV, Colloque C1, supplément au Journal de Physique III, Vol.2, Avril 1992.
- [Baudois 81] BAUDOIS D. et al., Etude théorique et expérimentale du corrélo-filtre, 8ème colloque GRETSI 1981.
- [Bellanger 89] BELLANGER M., Analyse des Signaux et Filtrage Numérique Adaptatif, Collection Technique et Scientifique des Télécommunications, 1989.
- [Benallal et Gilloire 88] BENALLAL A., GILLOIRE A., A new method to stabilize fast RLS algorithms based on a first-order model of the propagation of numerical errors, IEEE Trans. on ASSP, 1988.
- [Carayanis et al. 83] CARAYANIS et al., A fast sequential algorithm for LS filtering and prediction, IEEE Trans. on ASSP, vol.31, p.1394, Décembre 1983.
- [Cioffi et Kailath 84] CIOFFI J.M., KAILATH T., Fast, Recursive Least Squares transversal filters for adaptive filtering, IEEE Trans. on ASSP, vol.32, p.304, Avril 1984.
- [Cioffi et Kailath 85] CIOFFI J.M., KAILATH T., Windowed Fast Transversal Filters adaptive algorithms with normalization, IEEE Trans. on ASSP, vol.33, p.607, Juin 1985.
- [Clark 83] CLARK et al., A unified approach to time- and frequency domain realization of FIR adaptive digital filters, IEEE Trans. on ASSP, vol.31, no.5, Octobre 1983.
- [Feit et Liu 85] FEIT D., LIU Y.N., The nearfield response of a line-driven fluid-loaded plate, J. Acoustical Society of America, August 1985.
- [Friedlander 82] FRIEDLANDER B., Lattice filters for adaptive processing, Proc. IEEE, vol.70, no.8, Août 1982.
- [Givens 58] GIVENS W., Computation of plane unitary rotations transforming a general matrix to triangular form, J. Soc. Ind. Appl. Math, vol.6, pp.26-50, 1958.
- [Guerre-Chaley 90] GUERRE-CHALEY J.F., Etude de différentes structures de sous-

- tracteurs de bruit adaptatifs multiréférences, Thèse de doctorat INPG, Décembre 1990.
- [Harris 78] HARRIS, On the use of windows for harmonic analysis with the DFT, Proc. IEEE, vol.66, no.1, Janvier 1978.
- [Haykin 93] HAYKIN S., Adaptive filter theory, Prentice Hall, 2nd ed., 1993.
- [Haykin 96] HAYKIN S., Adaptive filter theory, Prentice Hall, 3rd ed., 1996.
- [Jin et al. 96] JIN Q. et al., Optimum filter banks for signal decomposition and its application in adaptive echo cancellation, IEEE Trans. on Signal Processing, vol.44, no.7, p.1669, Juillet 1996.
- [Junger et Feit 86] JUNGER M.C., FEIT D., Sound, Structures and their Interaction, 2nd Edition, 1986.
- [Kailath 74] KAILATH T., A view of three decades of linear filtering theory, IEEE Trans. on Information theory, vol.20, pp.145-181, March 1974.
- [Kalman 60] KALMAN R., On the general theory of control, Proc. IFAC Congress, London, 1960.
- [Kung et Leiserson 78] KUNG H.T., LEISERSON C.E., Systolic arrays (for VLSI), Soc. Ind. Appl. Math., pp.256-282, 1978.
- [Lin 84] LIN D.W., On digital implementation of the Fast Kalman Algorithms, IEEE Transactions on ASSP, vol.32, p.998, Octobre 1984.
- [Macchi 86] MACCHI, Advances in adaptive filtering, Digital Communications, Elsevier Publishing, North-Holland, p.41, 1986.
- [Michaut 92] MICHAUT F., Méthodes adaptatives pour le signal, Editions Hermès, 1992.
- [Morgan et Thi 95] MORGAN D.R., THI J.C., A delayless subband adaptive filter architecture, IEEE Trans. on Signal Processing, vol.43, no.8, p.1819, Août 1995.
- [Ogue et al. 83] OGUE et al., A fast convergence frequency domain adaptive filter, IEEE Trans. on ASSP, vol.31, no.5, Octobre 1983.
- [Oppenheim et Schaffer 75] OPPENHEIM A.V., SCHAFFER R.W., Digital Signal Processing, p.556, Prentice-Hall, 1975.
- [Papamichalis 90] PAPAMICHALIS P.E., Digital Signal Processing Applications with the TMS320 family, vol.3, 1990.
- [Servière 89] SERVIÈRE C., Éléments de comparaison entre différentes méthodes de soustraction de bruit, Thèse de doctorat INPG, Mai 1989.
- [Servière et al. 89] SERVIÈRE et al., Comparaison de deux méthodes d'estimation du gain complexe d'un filtre optimal au sens de Wiener, Traitement du Signal, vol.6, 1989.

- [Shynk 92] SHYNK J.J., Frequency domain and multirate adaptive filtering, IEEE Signal Processing Magazine, vol.9, pp.14-37, Janvier 1992.
- [Slock et Kailath 88] SLOCK D.T.M., KAILATH T., Numerically stable fast Recursive Least-Squares transversal filters, IEEE Trans. on ASSP, p.1365, 1988.
- [Thomson ASM 90] Thomson Sintra ASM, Sonar et Acoustique Sous-Marine, Tome1 (Le Milieu Marin), 1ère édition, Décembre 1990.
- [Widrow 75] WIDROW et al., Adaptive noise cancelling : principles and applications, Proc. IEEE, vol.63, no.12, Décembre 1975.
- [Widrow et Stearns 85] WIDROW B., STEARNS S.D., Adaptive signal processing, Prentice Hall, Signal Processing series, 1985.
- [Wiener 49] WIENER N., Extrapolation, Interpolation and Smoothing of Stationary Time Series, with Engineering applications, Wiley, 1949.



UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE
SCOLARITE - 3 ème Cycle
59655 VILLENEUVE D'ASCQ CEDEX
THESE DE DOCTORAT

PRESENTEE DEVANT L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

- VU la loi n° 84-52 du 26 janvier 1984 et notamment son article 17
- VU le décret n° 84-573 du 5 juillet 1984
- VU l'arrêté du 30 Mars 1992 relatif aux études de troisième cycle
- VU les rapports de

GALLAND M.A. - Maître de Conférences HDR - Ecole Centrale de Lyon

WAGSTAFF P. - Professeur - Univ. Techno. Compiègne

LE PRESIDENT DE L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE :

1) Autorise Mr VERRIEST ERIC

à soutenir une Thèse de Doctorat en :

ELECTRONIQUE

sur le sujet suivant :

**CONCEPTION ET MISE EN OEUVRE EN TEMPS REEL D'UN SYSTEME
DE SOUSTRACTION DE BRUIT ADAPTATIF POUR ANTENNE SONAR**

2) Désigne :

DECARPIGNY J.N. - Directeur - ISEN Lille

GALLAND M.A. - Maître de Conférences HDR - Ecole Centrale de Lyon

WAGSTAFF P. - Professeur - Univ. Techno. Compiègne

CAPRON J.M. - Ingénieur - ISEN Lille

GIANGRECCO C. - Ingénieur - DCN Ingénierie Six Fours

ROURE A. - Ingénieur de Recherches - Labo. de Méca. et d'Acoustique Marseille

SALMER G. - Professeur -

SERVIERE C. - CR CNRS - INP Grenoble

pour constituer le jury de soutenance de cette thèse.

3) La soutenance aura lieu le 18-06-1998 à 10 Heures Amphi. de l'I.E.M.N.

Villeneuve d'Ascq le 9/6/98

Pour le Président
Le Vice-Président
chargé des Formations


H. DUBOIS

IMPLIATION TRANSMISSION A

Monsieur Eric VERRIEST

Pour le Secrétaire Général,

Par déléation

Le Chef du Service Scolarité


J.C. CALBRE