

CHAPITRE 5

LE PROCESSEUR DÉDIÉ AU TRAITEMENT NUMÉRIQUE

Notre application du rehaussement de la parole est destinée à être mise en œuvre sur un DSP pour toute application embarquée ou dédiée en temps réel, et notre choix s'est fixé sur le produit de Texas Instruments qui est un des principaux chefs de file dans ce domaine, mais aussi parce que ses produits de développement sont relativement conviviaux et performants et les ressources d'assistance sur internet sont utiles et accessibles. De plus, notre référence [23] nous permettait une introduction rapide sur le produit choisi, ce qui contribue à son utilisation aisée.

Le produit choisi de Texas Instruments est le DSK du TMS320C6711 qui vient avec comme matériel une platine de circuit imprimé avec ses composants et interfaces nécessaires pour la programmation, le développement et la simulation et le déverminage du DSP TMS320C6711. Son logiciel associé est le CCS qui possède une configuration IDE pour ordinateur de table ou portatif, il incorpore les principaux utilitaires de développement usuels, c'est-à-dire, éditeur pour les codes sources en langage C et en langage assembleur, le compilateur avec éditeur de liens, et les outils de déverminage inclus. Il s'insère très bien dans l'environnement Windows et autre système d'opération d'ordinateur.

Le processeur de traitement numérique TMS320C6711 est très performant pour la présente application. Travaillant avec une cadence d'horloge à 150 MHz, il peut, lorsqu'il est mené vers une optimisation maximale, accomplir selon [23] jusqu'à 900 MFLOPS ce qui se traduit par 1200 instructions par seconde sur des registres de 32 bits. L'adressage de 32 bits également permet la gestion d'une mémoire de 4 Go disposée en différents bancs.

L'entrée et la sortie des signaux se font avec un taux d'échantillonnage de 8 KHz sur des convertisseurs A/D et D/A des 16 bits. De plus il existe des interfaces en option pour se

relier à la carte mère qui ajoutent des possibilités accrues comme entrées et sorties stéréo avec possibilité d'un taux d'échantillonnage variable.

La seule limitation importante pour notre application est que nous n'avons pas deux canaux d'entrée à notre disposition, il faut donc prévoir une alternative pour simuler cette condition exigée. Dans un premier temps nous pensons faire usage de fichiers représentant les deux canaux ou faire appel à des enregistrements séparés mais alors en temps différé.

CHAPITRE 6

MISE EN ŒUVRE DES ALGORITHMES EN LANGAGE C

Les simulations faites sur Matlab nous ont servi de départ pour mettre au point le programme source en langage C de standard ANSI. La référence [23] nous a été très aidante. Nous y avons puisé des éléments de code pour l'adapter à nos besoins en particulier l'algorithme et la mise en œuvre performante de la FFT qui est utilisée intensivement pour toutes les convolutions nécessaires tant dans le filtrage numérique que dans la transformée en ondelette continue où le nombre de coefficients est important, mais également pour la transformée de Hilbert qui a été nécessaire pour définir l'enveloppe des signaux analysés.

Le programme peut être retrouvé à l'annexe II, c'est une version de déverminage développée sous environnement Borland. Cette phase de développement intermédiaire a été nécessaire compte tenu que nous ne pouvions pas avoir les interfaces dédiées aux deux canaux de traitement sur l'outil de développement DSP de Texas Instruments. Il a donc fallu simuler la saisie des deux canaux par la lecture de deux fichiers contenant l'information des signaux à traiter.

La méthode est aussi plus propice pour évaluer les performances de l'application dans sa phase de développement puisqu'elle assure un moyen répétitif de reproduire un contexte précis en observation étudiée, ce qui est exclu ou plus difficile dans le cas de l'emploi des microphones en temps réel.

Les grandes lignes des fonctionnalités de l'application peuvent se résumer ainsi : pour chaque segment prélevé du signal sur chacun des deux canaux nous avons à accomplir :

- Transformée en ondelettes dyadique réalisée par FFT sur les trois échelles
- Détermination de la fréquence fondamentale f_0
- Construction du banc de filtres de type passe-bande selon la f_0 retenue

- Filtration du signal par FFT, la saisie et l'obtention des sous-bandes filtrées $r_{i,k}(t)$ sur chaque canal
- Relevé de l'enveloppe $\hat{A}_{i,k}(t)$ sur les signaux des bandes filtrées par transformée de Hilbert par l'emploi de la FFT
- Génération des signaux synthétisés $Z_{i,k}(t)$ selon f_0 ou kf_0 modulée par l'enveloppe
- Recouvrement de l'information de la phase contenue dans chaque sous-bande filtrée par filtre de Wiener avec l'approche LMS pour livrer les signaux synthétisés avec correction de phase $x_{i,k}(t)$
- Combinaison de chaque sous-bande synthétisée des deux canaux puis traitement sur filtre adaptatif de Wiener par la matrice de séparation et la prédiction linéaire du rehausseur
- Chaque résultat précédent est sommé pour retourner le signal rehaussé en sortie

Chaque bloc fonctionnel de l'ensemble de l'application, énuméré plus haut, a été développé et testé et comparé par rapport aux résultats de la simulation sous Matlab et leur intégration dans le programme principal est terminée. Un imprimé des différentes étapes d'exécution du programme exécutable est placé à la suite de l'imprimé du code source dans l'annexe II.

La figure 34 montre un résultat intermédiaire où l'on peut observer le signal synthétisé de la première sous-bande du canal 2 à partir du signal filtré de la même sous-bande. La figure 35 montre le signal estimé à la suite des perturbations de bruit convolutif et additif subi par le signal original de la source visée.

La figure 36 montre le résultat pour le son « Matlab » bruité avec un RSB de 3 dB et convolué. Il est intéressant de remarquer que le signal estimé ne contient plus de composantes continues puisque nos bancs de filtres sont du type passe-bande.

Comme le signalaient les auteurs [10] la périodicité du signal estimé permet la perception cependant de la parole lorsqu'on écoute ce signal sous l'environnement de Matlab avec la fonction « sound.m ». Mais pour améliorer le signal estimé nous avons ajouté un filtre passe-bas contigu à la limite inférieure du premier passe-bande contenant la f_0 comme fréquence centrale. Cela a donné le résultat amélioré de la figure 37.

Compte tenu de la portabilité du langage C, le programme source élaboré dans cette étude peut facilement s'adapter moyennant quelques ajustements pour tenir compte des particularités de la configuration du DSP choisi.

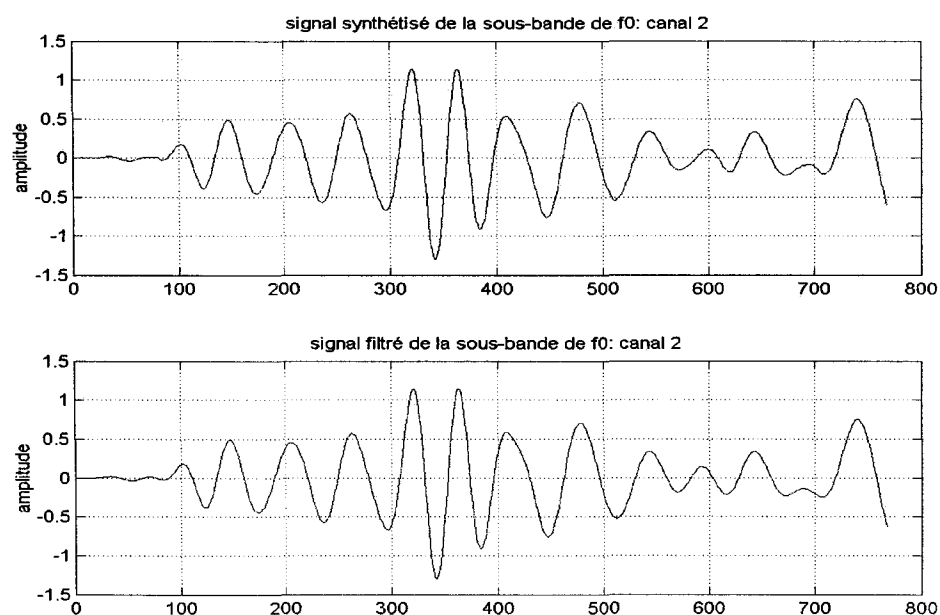


Figure 34 Signal synthétisé à partir du signal filtré sur la première sous-bande

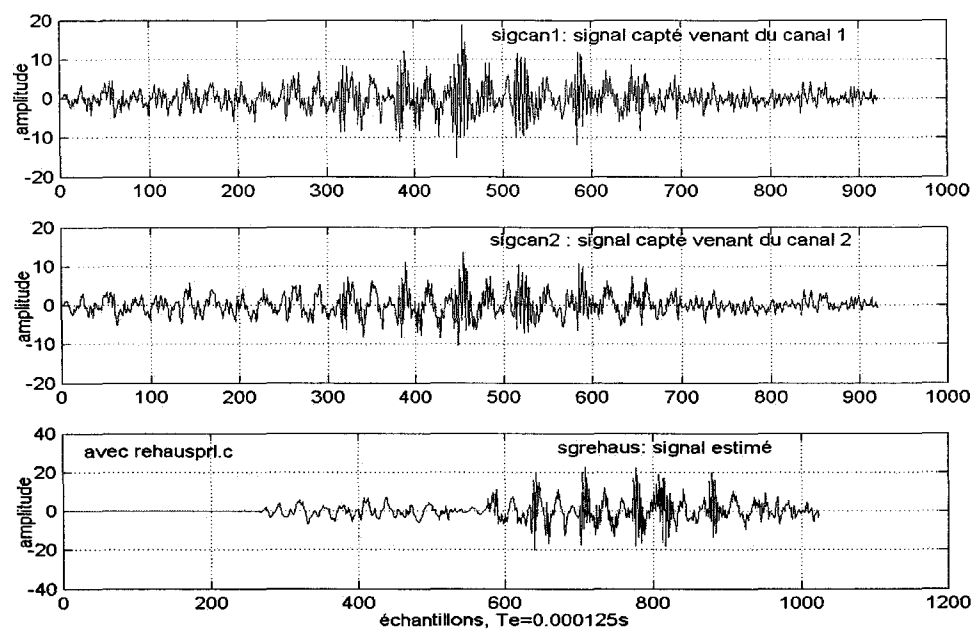


Figure 35 Signal estimé à partir des signaux captés

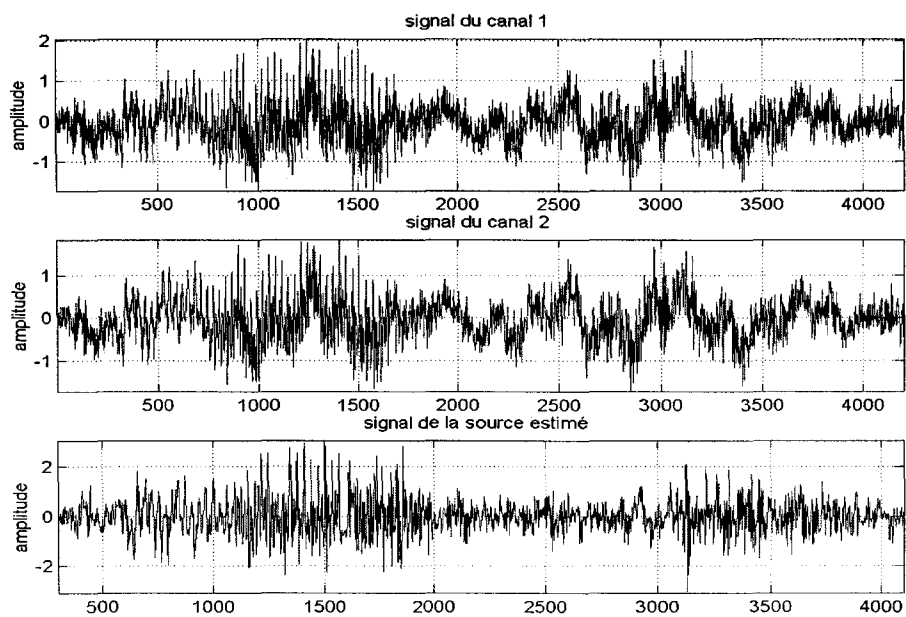


Figure 36 Estimation du son "Matlab" sans composantes continues

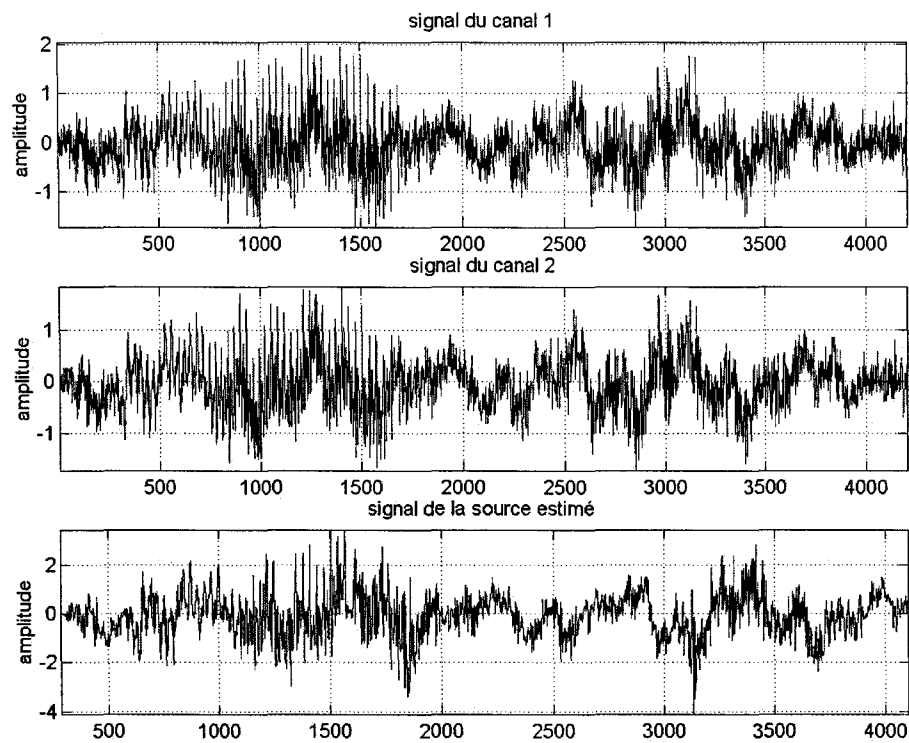


Figure 37 Estimation du son "Matlab" avec composantes continues

DISCUSSION ET INTERPRÉTATION DES RÉSULTATS

Les figures 28 à 32 donnent les composantes fréquentielles extraites et estimées de la source principale venant des deux canaux et la figure 33 donne la sommation de toutes les contributions pour aboutir au signal estimé qui est mis en comparaison par rapport au signal original. La figure 35 donne le résultat obtenu sous l'application développée en C à partir des signaux captés venant des deux canaux.

On observe autant dans la simulation que dans l'application en langage C que les composantes fréquentielles élevées du signal original sont atténuées ou absentes sur le signal estimé, cependant la signature du signal sur la portion recouverte paraît préservée en dépit des bruits additifs et convolutifs subis.

Le délai observé sur la réponse du signal estimé est inhérent au traitement de la filtration de l'analyse fréquentielle et adaptative.

Le rehaussement du signal, ici la parole est évidemment un processus d'optimisation qui mène à l'impossibilité de la restitution fidèle idéale de la source originale. Il faut alors mettre en œuvre certaines méthodes objectives ou subjectives pour évaluer le degré de rehaussement apporté.

Généralement en traitement de signal, on a recours à la mesure du rapport signal sur bruit, cependant selon plusieurs avis d'experts du signal de la parole ce critère n'est pas toujours révélateur dans ce domaine; il arrive qu'un bon RSB d'un signal de la parole aboutisse à une intelligibilité médiocre de cette dernière et d'autres fois, un signal avec un faible RSB donne une bonne intelligibilité.

Une approche pseudo-objective est souvent préconisée. Elle se réfère à l'évaluation de la qualité de l'information de la parole et recommandée par la norme G.107 de l'ITU. C'est

la méthode qui met en place une échelle pondérée de notes subjectives comprises entre 1 et 5 donnant une appréciation de la qualité perçue par l'auditeur. Elle est connue sous l'acronyme MOS (pour Mean Opinion Score). Le tableau II présente ce pointage.

Tableau II

Échelle du MOS

Échelle du MOS	Valeur qualitative
5	Excellent
4	Bon
3	Moyen
2	Dégradé
1	Mauvais

La récupération des résultats pour les différents signaux estimés a été faite à l'aide de la fonction « sound » de Matlab. En excluant l'effet de troncature du signal à la fin de l'écoute, on peut dire que les signaux nous semblent bien reconnaissables aux échelles 3 à 4 du MOS et ceci malgré le fait que le bruit convolutif ait été ajouté sur les sources originales.

On devrait donc avoir un résultat amélioré avec le traitement en temps réel dans la mise en œuvre du DSP, car selon les auteurs [10] l'analyse spectrale à court terme des bancs de filtres devrait réduire voir annuler l'effet convolutif, ce qui n'a pas été le cas dans notre simulation ou dans nos essais avec l'application en C. Dans ces deux cas l'effet convolutif a été gardé.

Une autre approche plus quantitative cette fois-ci mais tout aussi utilisée se base sur le critère de la grandeur de l'erreur estimée présente dans les processus impliqués amenant au rehaussement de la parole.

Sur les figures 28 à 32 l'erreur se trouve sur le dernier graphique de chaque tranche fréquentielle liée à la fondamentale f_0 ou à une de ses harmoniques. On constate que l'erreur absolue croît au fur et à mesure que la bande fréquentielle augmente dans le domaine spectral où la vitesse de convergence ne peut suivre les changements rapides des signaux. Ce qui se démontre bien dans le résultat global obtenu sur la figure 33 et plus prononcé sur la figure 35.

Une tentative d'explication peut être ici soumise. Deux éléments des algorithmes mis en œuvre par notre application sont, selon nous, imputables à ce résultat.

Lors de la conception des filtres passe-bande de chaque banc de filtre, pour respecter d'une part, la limite imposée par la fréquence de Nyquist et d'autre part, pour conserver constant le facteur de qualité de chaque passe-bande nous avons plus ou moins délibérément omis la partie haute du spectre selon la f_0 trouvée voir à titre d'exemple la figure 18 sur ce point.

Nous avons constaté que l'ajout d'un filtre passe-bas avant le premier passe-bande sur la f_0 améliorerait le rehaussement en introduisant la composante continue du signal, comme le montre les résultats sur la figure 37 par rapport à la figure 36.

L'autre cause vient du prédicteur linéaire dans la partie de l'ACI où il est vu comme un rehausseur adaptatif de ligne dont le nombre de coefficients est proportionnel à la période de la f_0 ou d'une de ses harmoniques ainsi donc la traçabilité des variations rapides sur les composantes de fréquence élevée ne peut être garantie ou apprise.

De plus, comme nous l'avons vu précédemment, toute l'approche adaptative basée sur les algorithmes du LMS repose sur l'hypothèse de la stationnarité des signaux traités.

Nous savons cependant que dans notre cas, le signal de la parole ne possède pas toujours cette particularité.

[MCours.com](https://www.mycours.com)