



## Méthodes d'implantation d'un algorithme d'annulation d'écho acoustique $\text{GMDF}\alpha$

Adel Baganne\*, André Gilloire\*\*, Eric Martin\*, Patrick Martineau\*, J.P. Thomas\*\*

\*LESTER (Laboratoire d'Electronique des Systèmes Temps-Réel),

10 rue Jean Zay, 56100 Lorient France, tél. 97 87 28 48 Fax.97 87 28 15

\*\*CNET LAA/TSS/CMC- Route de Trégastel BP 40-22301 Lannion Cédex France

ET GDR 134

### RESUME

Dans le but de réduire les erreurs de spécifications et de mise en oeuvre des algorithmes sur les architectures parallèles, et de limiter au maximum les tests matériels, il est important d'avoir des outils de CAO performants. Nous présentons deux méthodes d'implantation d'un algorithme de filtrage adaptatif ( $\text{GMDF}\alpha$ ) [1] pour l'annulation d'écho acoustique : premièrement en utilisant les outils SIGNAL [2] et SYNDEX [3] pour une implantation sur une machine multiprocesseurs à base de TMS320C40. Deuxièmement, en utilisant un outil de synthèse architecturale GAUT[4] pour la synthèse de circuits dédiés (ASICs). Des résultats d'applications en Téléconférence et en Radio Mobile sont proposés.

### I - INTRODUCTION

Les architectures parallèles sont une préoccupation actuelle du développement de l'informatique car elles offrent une solution à l'amélioration des performances. On peut utiliser le parallélisme au niveau des traitements comme au niveau des communications. Le temps de réponse imposé pour le traitement est un problème important auquel est confronté la majorité des algorithmes de traitement du signal évolués.

Ce problème peut être résolu de deux manières : soit on développe un circuit VLSI spécialisé, soit on utilise une machine parallèle. Ces solutions engendrent des difficultés d'implantation qu'il faut résoudre grâce au développement d'outils automatiques. Il devient aussi impossible de prévoir les performances de cette implantation car on ne peut plus se limiter au calcul des temps de cycle nécessaires à l'exécution, il faut prendre en compte le volume des communications et les relations de précedence entre processus. Notons finalement que pour un algorithme donné à implanter sur une machine donnée, le choix du placement des variables et des opérateurs, l'allocation des voies de communications aux transferts et l'ordonnancement des opérations ont un impact déterminant sur la performance finale.

### ABSTRACT

In the aim to reduce mistakes in the specification and in the implementation of algorithms on parallel architectures, and to minimize material tests, it becomes essential to use efficient CAD tools. In this paper, we present two implementation methods of the  $\text{GMDF}\alpha$  algorithm, an adaptive filter well suited to acoustic echo cancellation for teleconference applications. First method consists in a parallel implementation on multi DSP TMS320C40 board using CAD tools SIGNAL [2] and SYNDEX [3]. The second consists in a synthesis of dedicated circuit (ASIC) using a high level architectural synthesis tool called GAUT [5]. We show some results in acoustic echo cancellation in both Teleconferencing and Mobile Radio applications.

Dans cet article, on s'intéressera à l'implantation temps réel d'un algorithme de filtrage adaptatif  $\text{GMDF}\alpha$  [1]. Deux approches d'implantation ont été explorées, l'une ciblant des architectures multiprocesseurs, l'autre ciblant les ASICs. Dans le premier cas, nous exploiterons le parallélisme à gros grain (tâches de forte complexité) en utilisant les outils SIGNAL et SYNDEX. Dans l'autre approche, qui cible des circuits dédiés (ASICs), nous exploiterons le parallélisme à grain fin, qui sera traité automatiquement par l'outil de synthèse architecturale GAUT[4].

La partie II décrit brièvement l'algorithme  $\text{GMDF}\alpha$ . La partie III présente la méthode d'implantation de l'algorithme sur une machine multiprocesseurs à base de TMS320C40 en utilisant les outils SIGNAL et SYNDEX. La partie IV développe l'implantation matérielle de l'algorithme sous forme d'un circuit dédié (ASIC GAUT).

### II - DESCRIPTION DE L'ALGORITHME $\text{GMDF}$

L'algorithme de filtrage adaptatif  $\text{GMDF}\alpha$  peut être considéré comme une formulation par bloc du filtre NLMS (Normalised Least Mean Square) avec adaptation dans le



domaine fréquentiel. Il utilise une technique de convolution rapide appelée WOLA (Weighted Overlap-add) qui offre une bonne accélération de convergence.

Cet algorithme bénéficie d'un parallélisme potentiel important (Figure 1), dépendant des paramètres de la décomposition de la réponse impulsionnelle et de la taille du filtre [2] : le filtre adaptatif de longueur  $L$  est découpé en  $K$  segments de longueur  $N$  ( $L = K.N$ ), ce qui réduit le temps de traitement. De plus, un suréchantillonnage du signal d'entrée augmente le taux de rafraîchissement des coefficients du filtre dans un rapport  $\alpha$  ce qui améliore à la fois la vitesse de convergence la capacité de poursuite du filtre.

La Figure 1 représente le diagramme bloc de l'algorithme GMDF $\alpha$ .

Dans une application d'annulation d'écho acoustique en téléconférence, cet algorithme nécessite une mise en oeuvre parallèle pour réduire le temps de traitement et satisfaire les contraintes de temps de réponse [1].

### III - IMPLANTATION DE L'ALGORITHME SUR MACHINES PARALLELES

Cette méthode s'appuie sur la spécification et la validation de l'algorithme au moyen du langage synchrone SIGNAL et sur la génération de l'exécutif temps réel distribué par l'outil SYNDEX (Figure 2).

#### III - 1 -Spécification et validation de l'algorithme en SIGNAL:

SIGNAL [3] est un langage Synchrone à flot de données. Il permet une spécification des algorithmes sans tenir compte de toute implantation matérielle et des temps d'exécution. La description en SIGNAL permet de décrire des relations entre des suites d'événements valués, les signaux. Un Signal est une suite ordonnée, finie ou infinie, de valeurs auxquelles est associée une Horloge. Cette dernière représente la suite des instants de ses occurrences. Elle constitue un repère (dans une référence temporelle virtuelle) permettant le calcul des relations temporelles entre divers signaux.

L'algorithme GMDF $\alpha$  est représenté par un graphe flot de données conditionné (GFDC) dans lequel :

- les sommets (ou noeuds) sont caractérisés par des ports d'entrée ou de sortie auxquels est associé un processus de calcul ou de contrôle. Chaque processus peut être entièrement spécifié en SIGNAL ou faire appel à une fonction externe. Notons qu'un processus SIGNAL définit des relations temporelles et fonctionnelles entre les signaux d'entrée et de sortie.
- Les arcs représentent les liens de communications et de dépendances entre ports conditionnés.

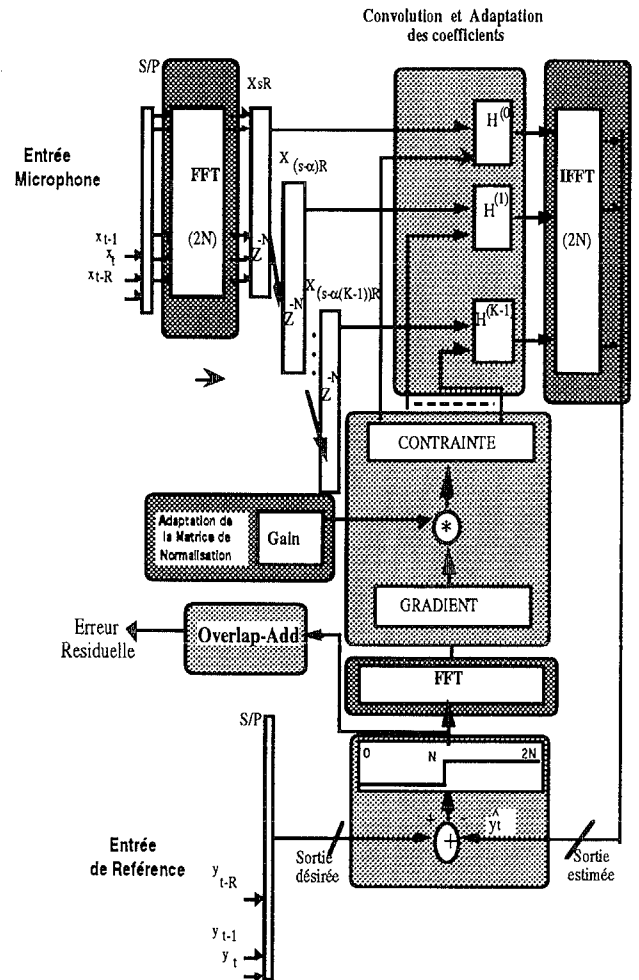


Figure 1 : Diagramme bloc de l'algorithme GMDF $\alpha$ .

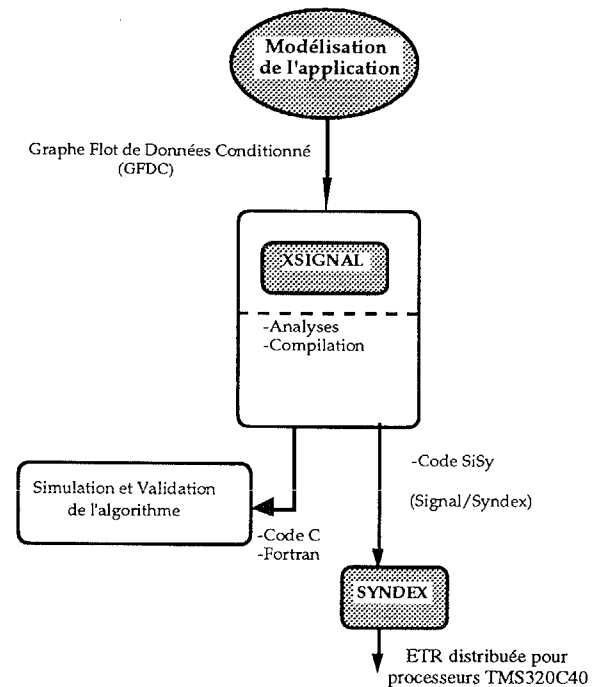


Figure 2 : Environnement SIGNAL / SYNDEX.

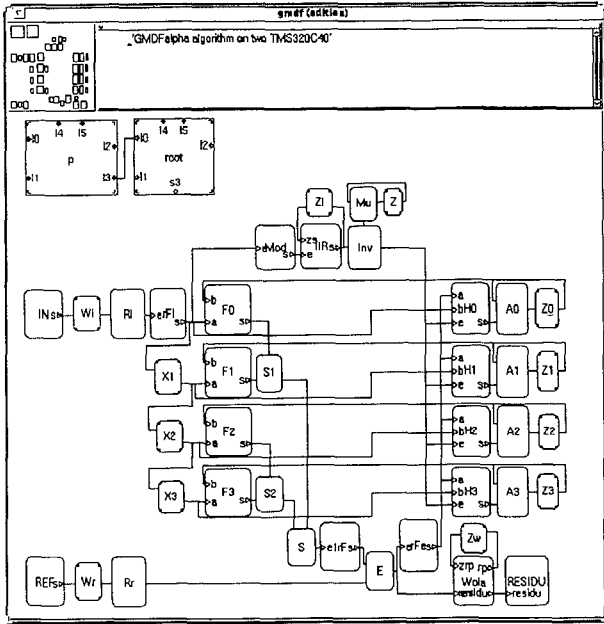


Figure 3 : Diagramme flot de données conditionné de l'algorithme GMDFA édité sous SYNDEX.

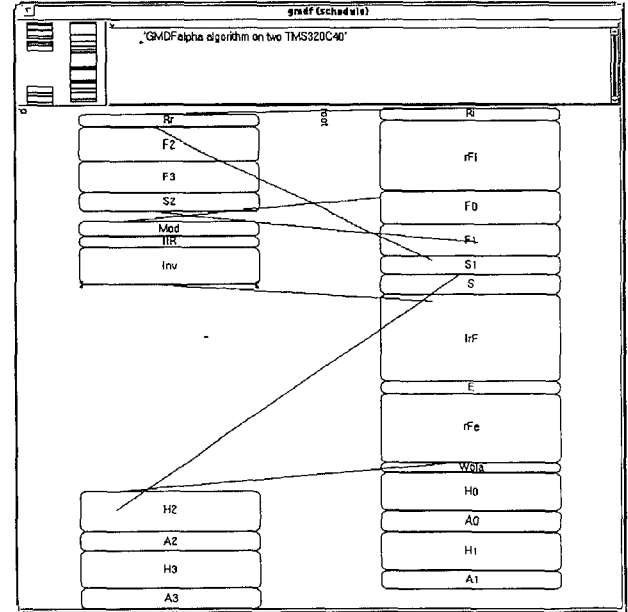


Figure 4 : Placement / Ordonnancement fourni par SYNDEX.

La phase de compilation vérifie la correction temporelle du programme et génère du code C et SiSy (SIGNAL pour SYNDEX). Le code C permet au programmeur de valider la description de l'algorithme. Le code SiSy représente un graphe flot de données conditionné (Graphe Logiciel) destiné à l'outil SYNDEX.

**III - 2 - L'outil SYNDEX**

SYNDEX [4] est un environnement de programmation graphique et textuel qui offre les services d'interfaçage avec SIGNAL, permet de spécifier et dimensionner une machine cible multiprocesseurs TMS320C40. Cet outil d'aide à l'implantation optimise la distribution et l'ordonnancement des tâches de l'algorithme. Il génère le code C pour chaque processeur, ainsi que l'exécutif temps réel qui gère les échéances et transferts de données entre tâches.

L'implantation de l'algorithme sur l'architecture consiste à réduire le parallélisme potentiel de l'algorithme au parallélisme matériel disponible. Les modèles de graphes utilisés pour spécifier l'algorithme et l'architecture conduisent à formaliser l'implantation en terme de transformations de graphes. La réduction est obtenue en transformant progressivement le graphe flot de données modélisant l'algorithme jusqu'à ce qu'il corresponde au graphe matériel modélisant l'architecture. Ces transformations représentent une distribution (allocation spatiale) et un ordonnancement (allocation temporelle) d'une part, des actions sur les processeurs et d'autre part, des communications interprocesseurs sur les liaisons physiques. A partir de cette distribution et de cet ordonnancement, un exécutif distribué temps réel, permettant l'exécution de l'algorithme sur l'architecture, est généré automatiquement.

La Figure 4 présente le placement / ordonnancement de l'algorithme GMDFA sur deux processeurs (root et p) fourni par SYNDEX. Le diagramme temporel obtenu fait apparaître les communications interprocesseurs (traits obliques), et pour chaque processeur, l'ordonnancement des processus qui y sont placés (hauteur proportionnelle à la durée d'exécution). Ce diagramme permet de mettre en évidence l'importance des communications.

**IV - IMPLANTATION DE L'ALGORITHME EN ASIC**

Une autre voie pour mettre en oeuvre des algorithmes de traitement du signal sur des architectures parallèles dédiées est celle de la synthèse d'architectures. L'outil de synthèse architecturale GAUT [5] génère une description architecturale du circuit ASIC à partir d'une description comportementale d'un algorithme en VHDL [6]. GAUT intègre une méthode de conception en deux étapes. La première étape transforme et parallélise le code VHDL qui décrit l'application à un niveau comportemental. Cette étape aboutit à une représentation interne de traitement, à parallélisme maximum, sous la forme d'un graphe flot de données. La seconde étape de la conception calcule le nombre de tranches de pipeline de l'architecture et sélectionne le jeu optimal d'opérateurs à mettre en oeuvre. Les opérations sont ensuite ordonnancées, puis les chemins de données (registres, bus, etc.) sont optimisés.

Le résultat de la synthèse est fourni sous la forme d'un fichier VHDL qui décrit la structure des composants mis en oeuvre, ainsi que la machine à états finis qui contrôle sur cette structure l'exécution de l'algorithme. Dans cette approche, il s'agit d'exploiter le parallélisme à grain fin, de manière à optimiser l'implantation parallèle de l'algorithme. Ce résultat en VHDL forme le point d'entrée d'un outil de synthèse logique

du commerce afin d'obtenir le plan de masse du circuit.

A ce stade, il est possible de connaître la complexité de l'architecture conçue pour une application considérée. L'évaluation de la complexité des algorithmes est mesurée par le nombre de multiplications, ou par une somme pondérée des multiplications, additions et soustractions. Nous présentons les résultats obtenus par la synthèse architecturale dans les deux domaines d'applications: Téléconférence et Radio Mobile sous la forme d'un tableau (Table 1).

Domaine d'Application du Filtrage	Librairie COMPASS CMOS 1 $\mu$ m
<b>Application Radio Mobile (8 KHZ)</b> L = 128, K = 8, N=16 $\alpha = 4$	- 1 additionneur- soustracteur - 1 multiplieur - 7 registres - 5 multiplexeurs
<b>Application Téléconférence (16 KHZ)</b> L = 1024, K = 128, N=8 $\alpha = 2$	- 3 additionneurs - 2 soustracteurs - 5 multiplieurs - 36 registres - 29 multiplexeurs

Table 1. Complexité de l'architecture du GMDF $\alpha$  implanté sous forme VLSI.

## V - CONCLUSION ET PERSPECTIVES

Nous avons présenté deux méthodes d'implantation de l'algorithme adaptatif GMDF $\alpha$  exploitant respectivement un partitionnement fin et moyen. La granularité du partitionnement est un élément important : un partitionnement à grain fin est adapté aux machines SIMD (Single Instruction Multiple Data stream) et aux implantations VLSI car l'unité de calcul est faiblement chargée et les communications sont réduites et locales. Cette finesse permet de mettre en évidence tous les parallélismes potentiels de l'algorithme et des communications. Le partitionnement à grain moyen ou à gros grain peut être utilisé lorsque la machine cible comporte peu de processeurs [2]. Ces deux niveaux de parallélisme et partitionnement peuvent être utilisés simultanément dans une stratégie de Co-Conception matérielle et logicielle, aussi appelée CO Design.

Les deux méthodes d'implantations présentées illustrent bien la contribution des outils automatiques (CAD) de haut niveau dans l'implantation d'applications complexes. De tels outils aident le concepteur à envisager un nombre de solutions important dans le but de faciliter la mise en oeuvre parallèle des applications et d'optimiser leurs implantations.

S'il s'avère difficile au premier abord d'appréhender un nouveau langage de description spécifique à un outil, l'aspect générique de cette description doit permettre de rentabiliser en terme de temps de développement et de fiabilité la mise en oeuvre de cette démarche. Le Projet ASAR [7], actuellement en cours doit permettre d'unifier la spécification des applications indépendamment de l'architecture cible.

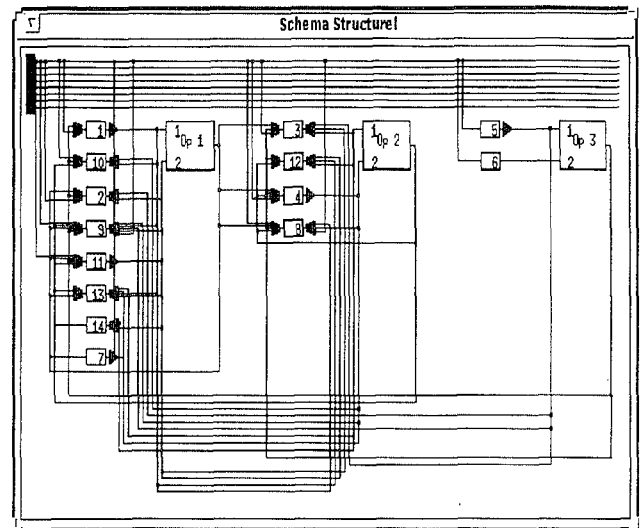


Figure 5. Architecture de l'Unité de Traitement dédiée à l'algorithme GMDF $\alpha$

## VI - BIBLIOGRAPHIE

- [1] O. Amrane, "Identification des systèmes à réponse impulsionnelle longue par filtrage adaptatif en fréquence : Application à l'annulation d'écho acoustique". Thèse de doctorat, ENST, 1992.
- [2] J. Sanchez "Prédiction et évaluation de performances des algorithmes adaptatifs implantés sur machines parallèles" Thèse de doctorat ENST, 2 novembre 1993.
- [3] P. Bournai, P. Le Guernic, "Un environnement graphique pour le langage SIGNAL". Rapport de recherche n° 2040, INRIA, 1993.
- [4] C. Lavarenne, Y. Sorel, "Spécification, optimisation des performances et génération d'exécutif pour application temps réel embarquée multiprocesseur avec SYNDEX", CNES international symposium, novembre 1992.
- [5] E. Martin, O. Sentieys, J-L. Philippe, "Synthèse architecturale de coeur de processeurs de traitement du signal". TSI n° 2/1994, pp 251-279.
- [6] R. Airiau, J-M. Bergé, V. Olive, J. Rouillard, "VHDL, du langage à la modélisation". Presses Polytechniques et Universitaires Romandes, Collection CNET / ENST, 1990.
- [7] M. Auguin, M. Belhadj, J. Benzakki, C. Carriere, G. Durrieu, T. Gautier, M. Israël, P. le Guernic, M. Lemaitre, E. Martin, P. Quinton, L. Rideau, F. Rousseau & O. Sentieys "Towards a multi-formalism framework for architectural synthesis: the ASAR project" 3° International Workshop on Hardware - Software Co-design Code/CASHE' 94, 22-24 septembre 1994, Grenoble.