

Réalisation d'un Terminal Multi-Standards : Performances et Avantages de la Technique des Opérateurs Communs, illustrés par l'ER-LFSR.

L.ALAUS¹ – J. PALICOT² – D.NOGUET¹

¹CEA-LETI, Minatec
17 rue des Martyrs, 38054 Grenoble Cedex 9, France

²SUPELEC/IETR
Avenue de la Boulaie CS 47601, 35576 Cesson-Sévigné Cedex, France

¹laurent.alaus@cea.fr, dominique.nogueta@cea.fr

²jacques.palicot@supelec.fr

Résumé - Dans ce document, nous traitons la problématique des terminaux multi-standards reconfigurables. Plus précisément, nous nous intéressons à une nouvelle méthodologie de réalisation et aux performances de son application sur un terminal tri-standards capable de gérer alternativement le 802.11g, le 802.16 et le 3GPP LTE. Nous avons déjà proposé en [1] une méthode de paramétrisation intitulée la technique des opérateurs communs. Dans cet article, nous affinons cette méthodologie en spécifiant une nouvelle organisation en Bancs et en illustrant celle-ci par plusieurs implémentations. Nous prouvons la facilité d'implémentation de la technique et son adaptabilité vis-à-vis des standards considérés.

Abstract - In the context of Software Defined Radio (SDR), parametrization is becoming a keystone in the design of multi-standard terminals. We define in [1], the Common Operator Technique, an original method to define multi standard terminals, based on a limited set of Common Operators. The technique enhances the reconfigurability and the scalability of the Terminal but requires a new organization of these operators to let them be executed correctly by the Terminal. In the proposed study, we present a new organization in Common Operator Bank (COB) to apply the Common Operator Technique. The COB creates a scalable design, limits the scheduling issue and reduces the number of operators to address.

1 Introduction

Placé dans le cadre des réseaux de télécommunications sans fils, l'objectif de ces travaux de recherches est de proposer une architecture reconfigurable pour les terminaux cognitifs [1][3]. Dans le contexte de la Radio Cognitive, nous assistons actuellement à la multiplication des normes et des standards de télécommunications. Le nombre croissant des standards normalisés permet d'élargir l'éventail des offres et des services disponibles pour chaque consommateur. Si cette prolifération des normes est un plus non négligeable pour les utilisateurs, elle engendre une complexité accrue des Terminaux à considérer. En effet, différents par les services proposés, les standards se distinguent par les bandes fréquentielles utilisées, les modulations, les codages et un ensemble de paramètres qui est spécifique à chacun d'eux. De cette multiplication des normes, le besoin en terminaux multistandard est apparu.

La méthode classique pour réaliser un terminal multistandard est la méthode « Velcro », c'est à dire la coexistence de toutes les entités architecturales nécessaires et d'un simple « Switch » qui permet de passer d'un mode à l'autre. La méthode « Velcro » permet de traiter plusieurs standards par le biais d'un seul et même terminal. Néanmoins, son évolutivité est

limitée par les standards considérés et la complexité de sa mise en œuvre reste maximale.

Pour palier ces inconvénients, nos travaux s'attachent à définir des terminaux multistandard reconfigurables dans le contexte de la Radio Logicielle Restreinte (SDR) [2]. Un état de l'art minutieux nous a permis d'identifier les Techniques de Paramétrisation [5] comme des méthodes pertinentes pour la réalisation d'un tel Terminal.

Les Techniques de Paramétrisation [4][5] ont comme idée fondatrice d'identifier les éléments communs aux différents standards et de proposer des architectures reconfigurables par simple téléchargement de paramètres. En respectant ces prérogatives, nous définissons en [1] ; « La Technique des Opérateurs Communs (OC) ».

Cette technique veut définir des entités de bases tant Software que Hardware de niveau de granularité élevé. Ainsi, nous voulons proposer un design reconfigurable se basant sur des opérateurs plus complexes mais tout autant génériques qu'un MAC ou une bascule.

Les Opérateurs considérés ont pour objectif d'être commun à différents standards mais aussi à différentes fonctions d'un même standard. Construits sur la base d'une architecture générique, les OC sont définis comme indépendants des fonctions et par conséquent des normes qui les utilisent. La technique des Opérateurs Communs aspire à trois objectifs : (1) un

design évolutif qui peut dans sa globalité s'adapter à n'importe quel standard (2) un design optimisé qui en se focalisant sur un set limité d'OC permet d'obtenir un gain en complexité vis-à-vis de la méthode Velcro Classique, (3) un design reconfigurable qui part simple téléchargement de paramètres permet de passer d'un mode d'exécution à l'autre.

La Technique des Opérateurs Communs est une approche ambitieuse dans le design de terminaux multistandard mais nécessite une gestion complexe des opérateurs communs. Ce « Scheduling » des Opérateurs Communs affecte non seulement la capacité du terminal à gérer les opérateurs mais aussi la complexité résultante de leurs implémentations en introduisant le besoin de structures de contrôle. Cet aspect est le point faible de la méthode et constitue un frein réel à son développement. Dans cet article, nous respectons les trois objectifs précédents et solutionnons ce problème en introduisant une architecture en Banc d'Opérateurs Communs. Nous détaillons dans une première partie, la technique des opérateurs communs en explicitant les exigences de son implémentation et en présentant le concept des bancs d'opérateurs qui y répond. Dans une deuxième partie, nous appliquerons la technique aux opérations à base de registres à décalages afin de valider notre approche et de démontrer le gain engendré tant en termes de gestion des opérateurs que de complexité.

2 La Technique des Opérateurs Communs.

Notation : Un Opérateur Common (OC) p est noté ${}_p O$. Un standard j : S^j , une fonction et un opérateur i de S^j sont notés : F_i^j et ${}_p O_i^j$. Un groupe du même ${}_p O$, dupliqué est noté ${}_p OG$.

2.1 Etat de l'art.

Les travaux existants relatifs aux Techniques de Paramétrisation ont aboutis à la création d'une première technique, appelée la technique des « Fonctions Communes » (FC) [4][5]. Comme expliqué en [4], la Technique des Fonctions Communes recherche des similarités fonctionnelles parmi les différents standards et crée des entités communes reconfigurables appelées Fonctions Communes. Les Fonctions Communes permettent d'obtenir un design reconfigurable localement par bloc. En outre, en se focalisant sur des fonctionnalités similaires, la Technique des Fonctions Communes permet de limiter la redondance d'opérations similaires pour plusieurs standards et de permettre un gain en complexité.

Néanmoins, dans le contexte d'un terminal multistandard, les Fonctions Communes apparaissent comme dépendantes des standards considérés. En se référant au concept initial et aux premiers exemples proposés en [4][5], la Technique s'apparente à une approche Velcro appliquée pour chaque groupe

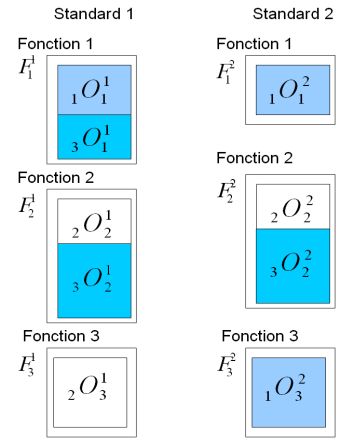


Figure 1 : Allocation des Opérateurs Communs.

fonctionnel où les différentes architectures requises sont juxtaposées pour donner la Fonction Commune. En conséquence de cette dépendance, le gain en complexité comme l'évolutivité de ces techniques sont faibles.

Tab 1 : Allocation des Opérateurs Communs

	S^1	S^2	S^3
F_1	${}_1 O_{1+3}^1 \quad {}_1 O_1^1$	${}_1 O_{1/3}^2 \quad {}_1 O_1^2$	${}_3 O_3^3$
F_2	${}_1 O_{2+2}^1 \quad {}_2 O_{1/2}^1 \quad {}_2 O_1^1$	${}_2 O_{2+3}^2 \quad {}_2 O_2^2$	${}_3 O_2^3$
F_3	${}_2 O_3^1$	${}_1 O_3^2$	${}_3 O_3^3$

2.2 La Technique des Opérateurs Communs.

Le travail réalisé lors de nos recherches, nous a permis d'identifier l'opportunité de passer outre cette dépendance aux standards et de proposer une technique de paramétrisation plus évolutive et performante en termes de complexité. Nous définissons en [1], la Techniques des Opérateurs Communs. Dans cette approche, nous ne cherchons plus à définir des similarités fonctionnelles entre les standards mais des similarités de granularité inférieure. Nous identifions ou désignons dès lors des Opérateurs Communs basés sur des Correspondances structurelles que ce soit d'un point de vue « Signal » comme « Architectural ». Un OC, avec le paramétrage adéquat, pourra être utilisé par plusieurs fonctions différentes d'une même chaîne d'un terminal. Ce dernier aspect est le point clé dans l'implémentation de la technique des opérateurs communs. En effet, étant donné qu'un opérateur commun peut être utilisé par plusieurs fonction d'un même standard simultanément, la gestion du nombre d'OC, en alternance par plusieurs fonctions à la fois devient un problème à résoudre.

Afin d'illustrer cette problématiques, nous pouvons nous référer au Tab I. Celui-ci illustre l'utilisation de plusieurs opérateurs par des Fonctions pour un même standard (Figure 1). Ainsi, ${}_1 O$ peut être utilisé par F_1^1 et F_2^1 , quand S^1 est sélectionné ou par F_1^2 et F_3^2 , quand S^2 est à son tour sélectionné. Deplus, F_2^1 peut être réalisée par ${}_1 O$ et ${}_2 O$ ou ${}_2 O$ seul. Dans ce cas précis,

$_2O$ peut réaliser les opérations de $_1O$ dans le cas de F_2^1 . La gestion des opérateurs (scheduling) au sein d'un même standard devient une problématique majeure à prendre en compte.

2.3 Le Banc d'Opérateurs Communs (BOC).

Afin de résoudre cette problématique, nous proposons une implémentation spécifique des opérateurs communs en banc. Pour se faire nous distinguons deux organisations des OC ; « locale » et « Globale ». L'organisation locale correspond à l'agencement d'un opérateur par une fonction donnée. L'exemple de l'opérateur « papillon » pour la FFT en est une parfaite illustration.

L'organisation globale est spécifique à la technique des opérateurs communs et qualifie l'utilisation en parallèle d'un même OC par des fonctions réalisées en simultanées ou à la suite.

Cette dernière organisation globale peut devenir ardue dans l'implémentation de notre Technique. Pour y remédier, nous proposons d'implémenter les OC en Bancs. Pour chaque type OC, nous implémentons une quantité fixe du même $_pO$ pour former un groupe $_pOG$.

Un BOC sera la concaténation de plusieurs $_pOG$.

Pour chaque $_pOG$, nous implémentons le nombre suffisant d' $_pO$ pour que chaque fonction (F_i^j) qui requiert leurs utilisations, possède des $_pO$ dédiés.

Ainsi, pour un standard j , une F_i^j n'aura pas à recourir à l'utilisation d'opérateurs réservés pour une autre fonction (F_i^j). Nous évitons ainsi, l'organisation globale des OCs. La Figure 2 illustre ce propos.

La définition des BOC respecte les prérogatives de la technique des Opérateurs Communs. Le BOC évite de considérer la gestion des opérateurs et les structures de contrôle requises. Néanmoins, en dupliquant des OC, nous augmentons potentiellement la complexité hardware du design, nous ne garantissons plus un gain

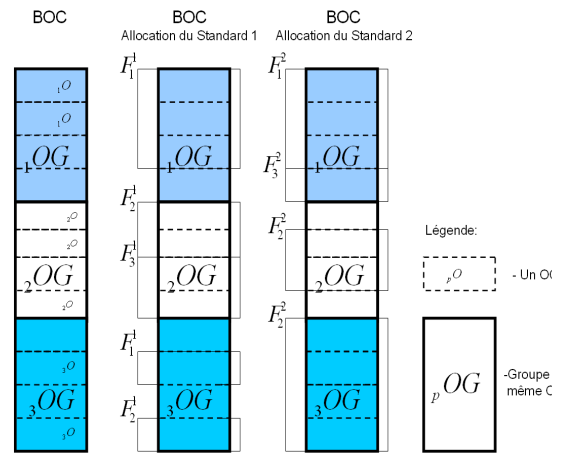


Figure 2 : Principe du Banc d'Opérateurs Communs

en complexité attendue avec la techniques des OC.

3 Application

3.1 Opérateurs Commun Considérés.

En nous focalisant sur les standards actuels que sont le 802.11g (WIFI), le 802.16 (Wimax) ou le 3GPP LTE [7][8][9][10], nous avons identifié que les registres à décalages (Shift Register – SR) et les registres à décalages à rétroactions linéaires (Linear Feedback Shift Registers – LFSR) étaient couramment utilisés à de nombreuses reprises tout au long de la chaîne de traitement. En effet, ces deux structures peuvent être employées pour des opérations aussi diversifiées que l'embrouillage, la génération de séquences aléatoires, le codage convolutif, le turbo codage, la génération des éléments du corps de Galois ou encore des étapes du codage et décodage Reed Solomon.

A partir de ce constat, nous avons développé quatre opérateurs communs à base de LFSR : (1) Reconfigurable Fibonacci LFSR (RF-LFSR), (2) Reconfigurable Galois LFSR (RG-LFSR), (3) Reconfigurable LFSR (R-LFSR) et (4) Extended Reconfigurable LFSR (ER-LFSR) (Figure 3). Les trois premiers opérateurs [11] sont respectivement des versions évoluées d'un LFSR de Galois, de Fibonacci et d'un Filter IIR. L'ER-LFSR, quant à lui peut être appréhendé comme la concaténation d'un filtre IIR, d'un filtre FIR et d'un LFSR de Fibonacci.

Comme décrit en [11] et [12] et illustré en Figure 3 et Figure 4, les opérateurs présentent des périmètres fonctionnels et des architectures gigognes. C'est-à-dire que seul l'ER-LFSR peut remplacer 100% des besoins en LFSR d'un standard basé sur le 802.11g, 802.16e et le 3GPP LTE. Comme l'architecture de l'ER-LFSR est complexe par rapport à un LFSR de base [11], nous avons développé trois autres LFSRs de complexité moindre mais limités à certaines fonctionnalités. Le tableau en Figure 3 représente ces fonctionnalités et les périmètres spécifiques de chaque OC.

3.2 Implémentation.

Dans cette publication, notre objectif est de prouver la viabilité de la technique des opérateurs communs et la simplicité de mise en œuvre du banc d'opérateurs. Nous

		RF-LFSR	RG-LFSR	R-LFSR	ER-LFSR	
Structures Génériques	Generateur de Séquences Aléatoires	X	X	X	X	Périmètre Spécifique du RF-LFSR
	Scrambleur	X	X	X	X	
	CRC Codeur		X	X	X	Périmètre Spécifique du RG-LFSR
	CRC Decodeur		X	X	X	
	Code Cyclique Correcteur d'Erreur		X	X	X	
	Codeur Convolutionnel Non Systematique			X	X	Périmètre Spécifique du R-LFSR
	Codeur Convolutionnel Recursif Systematique			X	X	
	Turbo Codeur			X	X	
	Code Cyclique Correcteur d'Erreur			X	X	
	Generateur Champs de Galois			X	X	
Codeur Reed Solomon			X	X	Périmètre Spécifique du ER-LFSR	
Decodeur Reed Solomon			X	X		
802.11g: Codeur Convolutionnel ERP-PBCC 22/33 Mbits/s				X		
Structures Spécifiques	802.16: Turbo Codeur (Rate 1/2 et 1/3)				X	Périmètre Spécifique du ER-LFSR
	3 GPP LTE: Scrambleur Code Generateur (Uplink et Downlink)				X	

Figure 3 : Fonctionnalités des Opérateurs Communs LFSR.

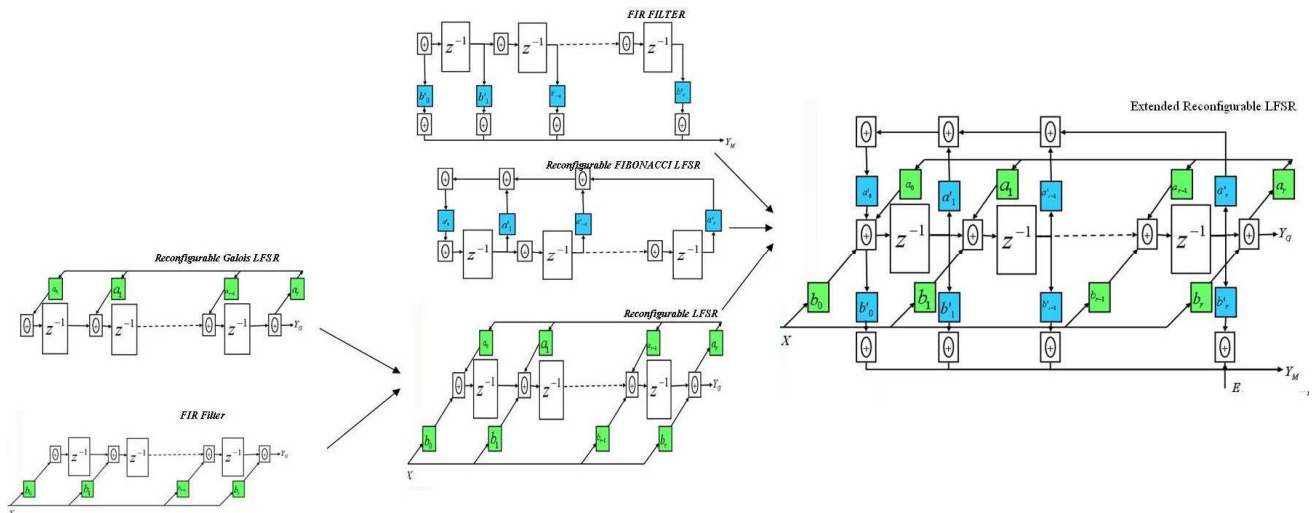


Figure 4 : Opérateurs Communs Considérés

ciblons un terminal tri standard (802.11g, 802.16 et 3 GPP LTE) où les structures substituables par l'ER-LFSR sont aux nombres de 29 ; 24 pour le R-LFSR, 19 pour le RG-LFSR et 9 par le RF-LFSR. Du fait des quatre différents opérateurs cités antérieurement, nous pouvons tester le concept des bancs d'opérateurs en implémentant plusieurs versions d'un même BOC qui doit remplacer les 29 structures mentionnées précédemment. Nous construisons trois BOC distincts. Le premier (BOC 1) se concentre sur les quatre opérateurs. Chacun développe un p OG par rapport aux fonctionnalités de son périmètre spécifique.

Le BOC 2 quant à lui est constitué du R-LFSR et de l'ER-LFSR. Le premier forme un p OG dédié aux « structures génériques » et le second aux « structures spécifiques ».

Le troisième BOC ne sera constitué que d'un seul p OG, construit à partir de l'ER-LFSR.

Nous présentons à titre d'exemple un tableau de résultats :

Tab 2 : Complexité des Différents BOC

Type d'Implémentation	Velcro	BOC 1	BOC 2	BOC 3
Complexité en %	100	81	79	60

Nous représentons en Tab 2, la complexité en termes de Logic Cells, réalisées sur une cible ALTERA/CYCLONE II en utilisant les outils Quartus. Quelque soit le nombre d'opérateurs différents considérés, chaque BOC obtient un gain de complexité significatif par rapport à l'équivalent Velcro.

4 Conclusions

Cet article a permis de détailler la technique des opérateurs communs, ces avantages mais aussi les inconvénients inhérents à son implémentation notamment en termes de gestion des opérateurs communs. Nous avons définis l'architecture en BOC qui

répond à cette problématique. Le BOC résout le problème du scheduling en dupliquant les opérateurs et en augmentant la complexité attendue de la technique des opérateurs communs. Néanmoins, l'application aux opérations à base de registres à décalage démontre un gain proche de 40% vis-à-vis d'une approche Velcro. Ainsi, l'organisation en Banc respecte bien les trois objectifs de reconfigurabilité, d'évolutivité et d'optimisation définis par la Technique des Opérateurs Communs.

5 Références.

- [1] Alaus, L., Palicot, J., Roland, C., Louet, Y., Noguét, D. (2009). Promising technique of parameterization for reconfigurable radio, the Common Operators Technique : fundamentals and examples. *Journal of Signal Processing Systems (Accepted)*
- [2] Mitola, J. (1995). The software Radio Architecture. *IEEE Communications Magazine*, (pp. 26-38).
- [3] SDR Forum - Software Defined Radio Forum, <http://www.sdrforum.org/>.
- [4] Rhiemeier, A. (2002). Benefits and Limits of Parameterized Channel Coding for Software Radio. 2nd Karlsruhe Workshop on Software Radios. Germany.
- [5] Jondral, F. (2002). Parameterization-a technique for SDR Implementation. In W Tuttlebee, *Software Defined Radio Enabling Technologies*, (pp. 233-256). Wiley.
- [6] Klapper, A., Goresky, M., (2002). Fibonacci and Galois Representations of Feedback-With-Carry Shift Registers. *IEEE Transactions on Information Theory*, 48(11).
- [7] ARIB STD-T63-25.212 V4.5.0, Multiplexing and channel coding, (FDD), (Release 4).
- [8] IEEE Std 802.11b-1999/Cor 1-2001
- [9] IEEE Std 802.11g-2003
- [10] IEEE Std 802.16 - Part 16
- [11] Alaus, L., Palicot, Noguét, D. (2008). A Reconfigurable Linear Feedback Shift Register for Software Defined Radio Terminal. ISWPC'08, Santorin, Grèce, 6-9 Mai 2008.
- [12] Alaus, L., Palicot, Noguét, D. (2008). A Extended Reconfigurable Linear Feedback Shift Register fir SDR Terminal. ISSSTA'08, Bologne, Italie, 6-9 Aout 2008.