Adéquation Algorithme Architecture - Approches matérielles et logicielles pour les applications de communications numériques

Bertrand Le Gal

▶ **To cite this version:**

Bertrand Le Gal. Adéquation Algorithme Architecture - Approches matérielles et logicielles pour les applications de communications numériques. Architectures Matérielles [cs.AR]. Université de Bordeaux, 2023. tel-04025353

HAL Id: tel-04025353
https://theses.hal.science/tel-04025353

Submitted on 12 Mar 2023

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

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

université de BORDEAUX

Habilitation à Diriger des Recherches

présentée devant l'Université de Bordeaux

École Doctorale des Sciences de l'Ingénieur
Spécialité : Électronique

par **Bertrand LE GAL**

Adéquation Algorithme Architecture – Approches matérielles et logicielles pour les applications de communications numériques

préparée au Laboratoire IMS (UMR 5218)
Cours de la libération, Talence, France

Soutenue le 2 février 2023

Jury :

Catherine DOUILLARD	–	Professeur	–	IMT Atlantique	*Rapporteur*
Lionel LACASSAGNE	–	Professeur des Universités	–	Sorbonne Université	*Rapporteur*
Jean-Marie GORCE	–	Professeur des Universités	–	INSA-Lyon / Inria	*Rapporteur*
Lilian BOSSUET	–	Professeur des Universités	–	Université Jean Monnet	*Examinateur*
Daniel CHILLET	–	Professeur des Universités	–	Université de Rennes 1	*Examinateur*
Christophe JEGO	–	Professeur des Universités	–	ENSEIRB-MATMECA	*Examinateur*

1441 • 1896 • 2014
université de Bordeaux

Table des Matières

Remerciements 3

Avant propos 5

I Présentation des travaux de recherche 7

1 Conception automatisée d'architectures dédiées 13
1.1 Introduction . . . 13
1.2 Synthèse HLS sous contraintes de performances . . . 14
1.3 Intégration des contraintes de sécurité dans la HLS . . . 18
1.4 Intégration de contraintes non fonctionnelles dans la HLS . . . 21
1.5 Conception de décodeurs ECC pour le standard 5G . . . 25
1.6 Diversification des domaines applicatifs . . . 32
1.7 Bilan des travaux . . . 34

2 Conception et utilisation d'architectures programmables 37
2.1 Introduction . . . 37
2.2 Conception d'un système programmable pour l'analyse protocolaire . . . 38
2.3 Architectures programmables et flexibles pour le décodage des codes LDPC . . . 43
2.4 Extension de l'approche aux architectures dédiées . . . 47
2.5 Implémentation de décodeurs ECC sur architectures HPC . . . 48
2.6 Extension des travaux aux autres familles d'ECC . . . 53
2.7 Extension du jeu d'instructions des architectures programmables pour les ECC . . . 58
2.8 Bilan des travaux . . . 59

3 Algorithmes de décodage et prototypage matériel d'ECC 63
3.1 Introduction . . . 63
3.2 Amélioration des performances de décodage des turbo décodeurs . . . 64
3.3 Étude et implantation de décodeurs LDPC matériels à très haut débit . . . 70
3.4 Étude, évaluation et implantation d'algorithmes de décodage innovants . . . 78
3.5 Étude et prototypage de récepteurs pour l'IoT . . . 85
3.6 Bilan des travaux . . . 86

4 Bilan et perspectives de recherche 89
4.1 Bilan scientifique . . . 89
4.1.1 Activités de recherche . . . 89
4.1.2 Bilan personnel . . . 91

4.2 Perspectives de recherche . . . 92
4.2.1 Activités de recherche . . . 92
4.2.2 Perspectives à court terme . . . 92
4.2.3 Perspectives à moyen terme . . . 93

II Parcours professionel **97**

5 Parcours Professionnel **99**
5.1 Parcours universitaire et professionnel . . . 99
5.1.1 Parcours universitaire . . . 99
5.1.2 Parcours professionnel . . . 100
5.2 Activités d'enseignements . . . 100
5.2.1 Activités passées (10/2002 → 08/2006) . . . 100
5.2.2 Enseignement à l'ENSEIRB-MATMECA (Sept. 2006 - ...) . . . 101
5.2.3 Responsabilités au sein de l'ENSEIRB-MATMECA . . . 103
5.3 Thématiques de recherche . . . 105
5.4 Production scientifique . . . 106
5.5 Encadrement doctoral et scientifique . . . 107
5.5.1 Encadrement de doctorants . . . 107
5.5.2 Encadrements de doctorants invités et de post-doctorants . . . 113
5.5.3 Encadrement d'étudiants de niveau master 1 & 2 . . . 114
5.5.4 Encadrement d'ingénieurs d'études . . . 117
5.6 Rayonnement . . . 118
5.6.1 Participation à des jurys de thèse . . . 118
5.6.2 Participation à des comités de suivi de thèse . . . 119
5.6.3 Participation à des comités de sélection . . . 120
5.7 Collaborations scientifiques et projets industriels . . . 123
5.7.1 Contrats publics de recherche . . . 123
5.7.2 Contrats privés de recherche . . . 125
5.7.3 Collaborations nationales et internationales . . . 129

Publications personnelles **131**

Bibliographie **149**

Abbreviations

ADMM Alternating Direction Method of Multipliers
ALM Adaptive Logic Module
ASIC Application-Specific Integrated Circuit
AWGN Additive White Gaussian Noise
BER Bit Error Rate
BNF Backus-Naur form
C-RAN Cloud Radio Access Network
CCSK Cyclic Code Shift Keying
CPU Central Proccessing Unit
CRC Cyclic Redundancy Check
DSL Domain-Specific Language
DSP Digital Signal Processing element
ECC Error Correction Code
FER Frame Error Rate
FFT Fast Fourier Transform
FIR Finite Impulse Response (filter)
FNC Flip and Check
FPGA Field Programmable Gate Array
GPU Graphics Processing Unit
HPC High Performance Computing
LDPC-CC LDPC Convolutional Codes
LDPC Low-Density Parity-Check
LP Linear Programming
LSB Least Significant Bit
LUT Look-Up Table
MSB Most Significant Bit
NISC No Instruction Set Computer
NMS Normalized Min-Sum
OMS Offset Min-Sum

OSI Open Systems Interconnection
PU Processing Unit
RTL Register Transfer Level
SIMD Single Instruction Multiple Data
SoC System On Chip
SoPC System On Programmable Chip
VHDL VHIC Hardware Description Language
PCIe Peripheral Component Interconnect Express

Remerciements

Un homme seul n'est rien, je voudrais avant tout remercier Dominique DALLET de m'avoir fait confiance en 2006, de m'avoir intégré dans l'équipe Circuits et Systèmes Numériques et d'avoir guidé mes premiers pas du côté obscur de la force. Je souhaite aussi remercier sincèrement Christophe JEGO, qui depuis son recrutement en 2010, pendant une période difficile, m'a permis au travers de nos multiples interactions de trouver ma voie et de m'ouvrir à la force.

Je voudrais ensuite remercier Catherine DOUILLARD, Jean-Marie GORCE et Lionel LACASSAGNE d'avoir accepté de rapporter sur ces 20 années de travaux de recherche. Leurs suggestions ont notablement contribué à améliorer ce document. Je remercie également, Lilian BOSSUET, Daniel CHILLET et Christophe JEGO, pour leurs remarques, questions et encouragements.

Ce mémoire serait moins épais s'il ne résumait que mon travail : on y trouve aussi celui des 11 doctorants qui ont quitté le nid et des 9 qui me supportent encore et que je ne peux pas tous citer ici pour des raisons évidentes de place...J'ai passé et je passe beaucoup de bon temps à échanger avec eux et surtout à essayer de les suivre. Je les remercie de tout ce qu'ils m'ont apporté scientifiquement et humainement. Ces remerciements ciblent aussi l'ensemble des étudiants de master 1 et 2 qui ont choisi de partager un moment de vie avec moi et réaliser les tâches de développement souvent ingrates, mais nécessaires.

L'équipe CSN a toujours été une équipe dynamique au sein de laquelle, chacun m'a, à un moment ou à l'autre, donné un coup de main. Je ne résiste donc pas à la facilité de remercier tous ses membres présents ou passés avec qui j'ai eu la chance de collaborer ou avec qui je collabore toujours. Je remercie en particulier, dans l'ordre alphabétique : Lilian BOSSUET, Jérémie CRENNE et Camille LEROUX. Ces remerciements s'étendent au personnel administratif et technique sans lequel nous ne pourrions pas travailler au quotidien.

Mon parcours et mon investissement dans l'univers de la recherche n'auraient pas été possibles sans bon nombre de discussions riches et animées 20 ans plus tôt avec les membres du laboratoire LESTER qui ont développé ma vocation. Pour cela, je salue amicalement tous mes collègues qui ont volontairement ou pas influé sur mes choix : Emmanuel CASSEAU, Guy GOGNIAT, Éric MARTIN, Jean Phillipe DIGUET et tous les permanents et doctorants de cette époque antédiluvienne que je n'ai pas la place de nommer...

Le métier d'enseignant chercheur incluant aussi un investissement dans les activités pédagogiques et administratives, je souhaite aussi remercier aussi l'ensemble des collègues de l'ENSEIRB-MATMECA avec lesquels j'échange régulièrement sur nos expériences et notre métier.

Enfin et surtout, je remercie mon épouse Rachel et mes fils Aymeric et Naël d'avoir juste été là !

Avant propos

Ce manuscrit présente une synthèse des activités de recherche, d'enseignement et administratives que j'ai menées depuis octobre 2002, date de début de ma thèse. Mes travaux de recherche ont été effectués au laboratoire LESTER jusqu'en aout 2005, puis se sont poursuivis au dans l'équipe R2D2 du laboratoire IRISA/INRIA à Lannion durant mon année d'ATER et enfin se poursuivent au laboratoire IMS depuis septembre 2006, date de ma nomination en tant que Maitre de Conférences à l'ENSEIRB-MATMECA. Au sein du laboratoire IMS, je suis membre de l'équipe *Circuits et Systèmes Numériques*.

Depuis déjà maintenant 20 ans, mes travaux de recherche s'intéressent au domaine de la conception de systèmes numériques efficaces en lien avec leurs domaines applicatifs. Ces travaux pluridisciplinaires mêlant informatique, électronique, traitement du signal et télécommunications m'ont permis au fil des années d'élargir mes domaines d'expertises. Depuis une dizaine d'années, mes activités se focalisent principalement sur la conception de systèmes temps réel, flexibles et efficaces pour les systèmes de communications numériques.

En parallèle de ces activités de recherche, je me suis impliqué de manière continue dans mon établissement d'enseignement, proposant des enseignements dérivés de mes activités de recherche pour répondre aux besoins de forte qualification requis par le monde industriel. Au-delà de l'aspect purement pédagogique, au travers des responsabilités assurées depuis 2006, j'ai participé à l'évolution de l'offre de formation de la filière Électronique.

Ce document est organisé en deux parties distinctes :

Partie 1 : Présentation des travaux de recherche

Cette première partie présente une synthèse des travaux que j'ai menés depuis l'obtention de mon doctorat. Elle propose tout d'abord une introduction qui positionne les trois axes de recherche autour desquels s'articulent mes travaux. Chaque axe est ensuite détaillé et une sélection de certains travaux est proposée afin d'illustrer l'activité menée. Enfin, une conclusion et des perspectives dressent une liste des actions envisagées dans l'avenir.

Partie 2 : Parcours professionnel

Cette seconde partie présente de façon complète l'ensemble de mon parcours professionnel. Les différentes contributions au niveau des activités d'enseignement, de recherche et administratives sont récapitulées. Les publications scientifiques sont également données afin d'illustrer la production scientifique de mes activités de recherche.

Partie I

Présentation des travaux de recherche

Cette première partie présente de façon approfondie les différents travaux de recherche que j'ai menés. Elle propose tout d'abord une introduction afin de positionner les différentes contributions suivant les trois axes de recherche autour desquels s'articulent mes travaux. Chaque axe est ensuite détaillé et une sélection de certaines activités est proposée afin d'illustrer l'activité menée. Enfin une conclusion et des perspectives sont proposées afin de préciser les actions envisagées dans l'avenir.

Introduction

Mes travaux de recherche, depuis 2002 et le début de ma thèse, se situent aux frontières de l'électronique, de l'informatique et des télécommunications, domaines qui ont connu des (r)évolutions majeures depuis lors. Dans cette partie, je présente une synthèse des activités de recherche auxquelles j'ai contribué successivement au sein des laboratoires :

— LESTER (actuellement Lab-STICC) de 2002 à 2005, en tant que doctorant ;
— dans l'équipe R2D2 du laboratoire IRISA/INRIA de 2005 à 2006, en tant qu'ATER ;
— dans l'équipe CSN du laboratoire IMS de 2006 à aujourd'hui, en tant que maître de conférences.

Ma première expérience en matière de recherche s'est déroulée au laboratoire LESTER sous la direction d'Éric MARTIN. Mon thème de recherche portait sur l'Adéquation Algorithme Architecture pour les systèmes embarqués. Plus précisément, mes travaux de recherche ciblaient le développement de méthodes et d'algorithmes permettant d'accélérer la conception de circuits numériques de niveau RTL sous contraintes à partir de descriptions comportementales. Cette approche de conception automatisée est actuellement connue sous le nom de synthèse d'architectures ou synthèse HLS. Mes activités de recherche se sont déroulées au sein d'un projet RNTL nommé ALIPTA. Le cadre expérimental de ces travaux était principalement lié au domaine des systèmes de communications numériques.

Pour faire suite à ces travaux, durant mon année d'ATER au sein de l'équipe R2D2 du laboratoire IRISA/INRIA, je me suis intéressé à étendre les travaux réalisés précédemment afin de généraliser les méthodes et les algorithmes développés en intégrant d'autres contraintes de conception. L'objectif sous-jacent était de minimiser la complexité matérielle et la consommation d'énergie des circuits numériques ainsi conçus.

Depuis mon arrivée à l'ENSEIRB-MATMECA, j'effectue mon activité de recherche au sein du laboratoire IMS (UMR 5218). Au sein du laboratoire IMS, mes recherches se déroulent au sein de l'équipe *Circuits et Systèmes Numériques (CSN)* dirigée actuellement par Jean-Baptiste BEGUERET. Mes activités dans l'équipe CSN se concentrent sur les problématiques d'Adéquation Algorithme Architecture principalement dans le domaine des systèmes de communications numériques. Ces activités sont à la frontière entre les sections CNU 27, 61 et 63, comme l'a mis en exergue mon recrutement sur un support de MCF en 63ème section alors que j'étais qualifié au sortir de ma thèse dans les sections 27 et 61. Indépendamment de cela, dans l'ensemble de ces domaines applicatifs, l'augmentation constante de la complexité calculatoire des applications nécessite une adéquation forte entre les algorithmes à intégrer et les architectures numériques servant de support à

leurs exécutions afin de respecter des contraintes d'intégrations toujours plus exigeantes. Afin d'aboutir au développement de solutions efficaces, l'intégration de l'algorithme original est souvent inopportune. En effet, ce dernier nécessite des adaptations ou des transformations plus ou moins conséquentes pour tirer pleinement parti des fonctionnalités offertes par les architectures numériques de type FPGA, CPU ou GPU. De manière analogue, les architectures numériques doivent elles aussi être repensées afin de s'adapter au mieux aux besoins applicatifs.

Mon activité de recherche couvre principalement trois thèmes :

1. La modélisation et la conception automatisée d'architectures numériques ;
2. La conception et l'utilisation d'architectures programmables ;
3. Les algorithmes et le prototypage matériel de codes correcteurs d'erreurs.

Les travaux réalisés dans chacun de ces thèmes sont décrits dans les trois prochains chapitres. Chaque partie contient par un bilan d'encadrement, de valorisation au travers de publications et de projets. D'autre part, des perspectives de recherche concluent le document.

J'ai rédigé ce manuscrit comme un document de synthèse. Ainsi les aspects techniques ne sont la plupart du temps pas abordés dans le détail. Seuls le contexte, les principes et les résultats obtenus sont décrits. Les détails pourront être consultés dans les articles référencés, dont certains sont reproduits en annexe.

Je vous souhaite une agréable lecture.

1 Conception automatisée d'architectures dédiées

1.1 Introduction

Ce premier chapitre dresse un bilan des projets que j'ai menés dans l'axe de recherche sur la *conception automatisée d'architectures dédiées*. Je décris brièvement le contexte de cet axe, introduis les travaux menés et donne son organisation depuis le début de ma carrière. La figure 1.1 synthétise les étudiants encadrés lors de ces travaux. Afin d'illustrer l'activité de recherche, la suite de ce chapitre présente différentes études qui ont suivi les activités initiées dans le cadre de mes travaux de thèse. La première section présente une extension de mes travaux de thèse permettant de mieux prendre en considération les caractéristiques matérielles des cibles ASIC/FPGA durant le processus de synthèse d'architectures. Dans un second temps, les travaux de recherche traitant de la prise en considération de contraintes non fonctionnelles telles que les problématiques de sécurité et de fiabilité sont détaillés. Les travaux ciblant l'exploitation d'outils de synthèse HLS pour la conception de décodeurs de codes correcteurs d'erreurs pour les systèmes de communicant de 5$^{\text{ème}}$ génération à partir de modèles comportementaux constituent le thème de la troisième section. Enfin, la dernière section résume brièvement les actions en cours avant qu'une synthèse des activités soit réalisée.

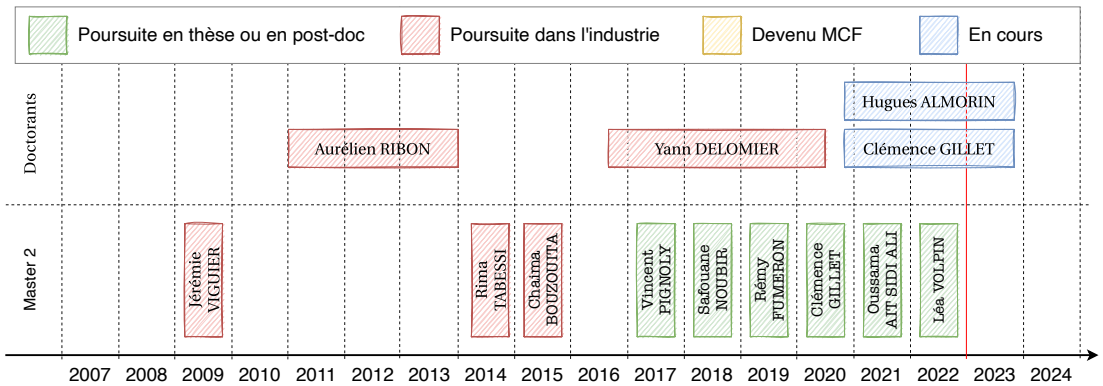


Figure 1.1 - Étudiants encadrés ayant participés aux études relatives à l'axe de recherche décrit dans le Chapitre 1.

1.2 Synthèse HLS sous contraintes de performances

Cet axe de recherche a débuté en octobre 2002 sous l'impulsion d'Éric MARTIN et d'Emmanuel CASSEAU avec le début de mes travaux de thèse [LeG05a]. Ces travaux de thèse m'ont permis d'étudier, de proposer et d'implanter des algorithmes au sein de l'outil de synthèse d'architectures GAUT [Cou08] développé au laboratoire LESTER. L'originalité de la méthodologie que j'ai proposée reposait sur trois axes complémentaires : la modélisation, l'ordonnancement des calculs et des accès mémoire et la génération matérielle (RTL) du contrôleur mémoire. J'ai proposé une méthode originale de modélisation de calculs indéterministes liés aux accès mémoires reposant sur une extension d'un graphe flot de données. La phase d'ordonnancement et d'assignation qui consistait à assurer la synthèse d'une solution architecturale fonctionnelle minimisant les transferts d'adresses entre la mémoire et les unités de calcul. Enfin, le développement d'une architecture matérielle efficace pour le contrôleur mémoire constituait le troisième axe de la méthodologie proposée. L'impact et la pertinence du processus de synthèse d'architectures proposé ont été vérifiés pour différentes applications issues des domaines du traitement du signal et des communications numériques. Ces travaux ont abouti à la publication d'un article dans une revue internationale [ri:LeG08] et neuf articles de conférence [ci:LeG04, ci:Cas04, ci:Hue04, ci:Cas05, ci:Abd05, ci:LeG05a, ci:LeG05b, ci:LeG05c, ci:LeG05d]. Ces travaux soutenus le 8 décembre 2005 m'ont permis d'appréhender les problématiques liées à la conception d'approches méthodologiques et d'outils logiciels pour la conception de circuits et de systèmes numériques. Les nombreux échanges enrichissants que j'ai pu avoir durant ces trois années de thèse avec d'autres doctorants et permanents du laboratoire (liste non exhaustive : Guy GOGNIAT, Jean Phillipe DIGUET, Philippe COUSSY, Gwenole CORRE, Sylvain HUET) m'ont motivé à essayer de rejoindre la famille de la recherche académique.

Les travaux entrepris dans le cadre de ma thèse de doctorat ont porté mon attention sur les limitations des méthodologies mise en œuvre dans les outils de HLS. Ainsi durant l'année où j'ai occupé un poste d'ATER à Lannion au sein de l'équipe R2D2 rattachée au laboratoire l'INRIA/IRISA, et ensuite lors de mes premières années au laboratoire IMS, j'ai investigué certaines des lacunes que j'avais identifiées. En effet, en 2005, pour limiter la complexité algorithmique du processus de synthèse, la dynamique des opérateurs matériels et des registres au sein du chemin de données était généralement considérée comme constante [Ell00]. Il résultait de ce choix une sous-optimalité des architectures matérielles produites. L'étude publiée dans [Ste00] estimait alors que 40% des bits impliqués dans les opérations arithmétiques étaient inutiles de même que 20% des bits mémorisés. J'ai donc décidé d'investiguer ce domaine pour proposer une approche méthodologique permettant de tirer avantage de ces spécificités algorithmiques lors de la conception d'architectures sous contraintes.

La nécessité d'optimiser les formats des données au sein des architectures numériques se justifie par le fait que le coût des ressources matérielles ainsi que leurs consommations d'énergie est fonction de la dynamique des informations manipulées comme cela est illustré dans le tableau 1.1 pour une technologie ASIC 65 nm. Afin d'améliorer l'adéquation entre l'algorithme modélisé et les architectures générées par les outils de synthèse HLS, j'ai proposé plusieurs évolutions du processus de synthèse afin d'estimer automatiquement la dynamique des données pour ensuite optimiser les architectures.

Ces travaux ont débuté par la proposition d'une approche d'estimation de la dynamique des données au sein de la description comportementale. Cette approche se basait sur la

Table 1.1 - Caractéristiques des ressources matérielles usuelles sur une technologie ASIC 65nm.

Paramètres	Ressources	Dynamique des données (bits)							
		4	8	12	16	20	24	28	32
Complexité (#NAND)	Additionneur	6	11	16	22	28	33	39	45
	Multiplieur	26	104	237	426	670	981	1338	1751
	Mux $2 \rightarrow 1$	3	5	8	10	13	15	18	20
	Registre	6	13	19	25	32	38	45	51
Energie (μW@40MHz)	Additionneur	12	23	35	46	58	70	81	93
	Multiplieur	15	42	84	139	209	298	421	558
	Mux $2 \rightarrow 1$	12	22	33	43	54	65	76	86
	Registre	3	7	10	13	17	19	23	26

propagation de l'espace de variations des opérations contenues dans la description algorithmique [ci:LeG06b, rn:LeG08]. Cette stratégie, proche des travaux [Ste00, Nay01, Tal03, Lee06], permet d'évaluer de manière sure la dynamique des données et des calculs à implanter. Les informations collectées lors de cette étape d'estimation servent à annoter le modèle de représentation interne employé lors du processus de synthèse tel que cela est schématisé dans la figure 1.2.

La première proposition que j'ai formulée pour améliorer la qualité des architectures matérielles produites se focalisait sur l'optimisation *a postériori* des architectures [ci:And06a, ci:And06b, ci:LeG06a]. Cette stratégie permettait de limiter l'augmentation de la complexité calculatoire des algorithmes mis en œuvre durant le processus de synthèse HLS. En effet, seule l'étape de génération de l'architecture matérielle nécessitait d'être modifiée pour produire des ressources matérielles en adéquation avec les opérations qui leur étaient assignées. Malgré la simplicité de

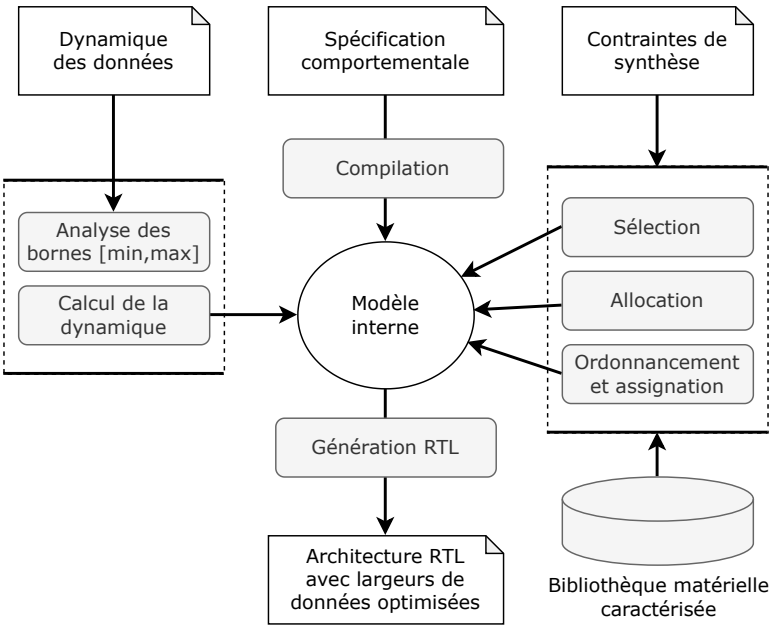
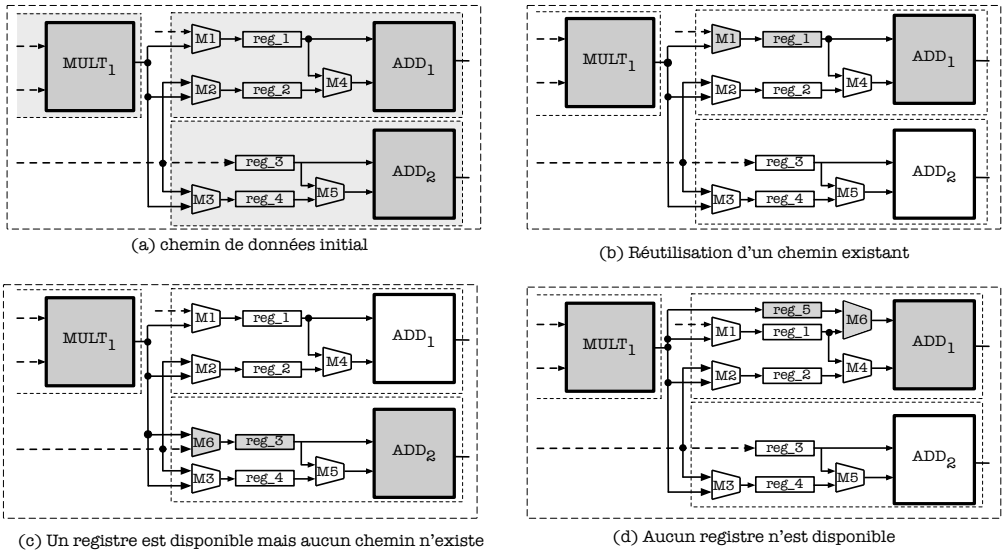


Figure 1.2 - Flot de synthèse d'architectures intégrant l'analyse et l'optimisation de la dynamique des données.



(a) chemin de données initial

(b) Réutilisation d'un chemin existant

(c) Un registre est disponible mais aucun chemin n'existe

(d) Aucun registre n'est disponible

Figure 1.3 – Différentes évolutions possibles d'un chemin de données permettant la réutilisation totale ou partielle des ressources déjà allouées pour réaliser une nouvelle opération d'addition.

l'approche proposée, des réductions de 10% à 40% de la complexité matérielle du chemin de données ont été mesurées [ci:And06a]. Toutefois, la variabilité des gains était fortement liée aux choix réalisés en amont durant les étapes d'ordonnancement et d'assignation.

Pour palier à cette forte variabilité, une seconde approche incluant la modification des étapes d'ordonnancement et d'assignation a été mise au point. Cette approche effectue dans un premier temps la sélection et l'allocation des ressources sans prendre en considération la dynamique des données. Les étapes d'ordonnancement et d'assignation sont réalisées conjointement. Cette étape conjointe exploite un modèle de coût basé sur l'étude des différents scénarios possibles (figure 1.3) pour estimer finement le coût d'assignation d'une opération à une ressource. La modélisation précise du chemin de données qui a été développée permet de guider de manière pertinente les choix réalisés lors de ces deux phases d'ordonnancement et d'assignation des opérations et des données sur les ressources matérielles. L'ensemble de ces travaux a été décrit et valorisé dans [ci:Cas09, rn:LeG08, ri:LeGIIb].

Les algorithmes ont été implantés dans l'outil de synthèse d'architectures *GraphLab*, outil de synthèse d'architectures que j'avais développé lors de mon passage à l'IRISA/INRIA en 2005[1]. Le tableau 1.2 synthétise quelques résultats expérimentaux obtenus post-synthèse logique pour des applications usuelles. La réduction de la complexité matérielle obtenue sur cible FPGA est en moyenne de 48% lorsque l'architecture de référence possède une dynamique constante [Cou08]. Un gain moyen de 10% est observé lorsque l'approche proposée est comparée aux approches antérieures [Con05, ci:LeG06a]. Les gains obtenus sur technologie ASIC 65 nm sont du même ordre de grandeur tel que cela est détaillé dans [ri:LeGIIb].

Ces travaux relatifs ont fait émerger d'autres leviers d'amélioration des caractéristiques des architectures générées. En effet, dans l'état de l'art [Kum98, Con00, Con01, Con05, Cou08], la durée d'exécution des opérations projetées sur les ressources matérielles était considérée comme

1. L'outil GAUT auquel j'avais contribué durant ma thèse n'était pas disponible à cette époque sous licence open-source.

Table 1.2 - Comparaison de la méthodologie proposée par rapport aux travaux antérieurs - Résultats obtenus post-synthèse logique [ri:LeGIIb] sur une cible FPGA de type Altera Cyclone III.

	Débit 10^6 exec/s	Dynamique d'entrée	Dynamique maximale	Complexité (LE)	Gains en complexité versus [Cou08]	[ci:LeG06a]	[Con05]
FIR 64	5.00	8 bits	20	3027	41.8%	1.7%	14.0%
	5.00	16 bits	36	4949	51.9%	1.3%	4.8%
DCT 8	13.33	8 bits	22	1475	44.0%	13.3%	35.1%
	13.33	16 bits	30	2650	47.1%	8.4%	25.6%
DCT 8x8	1.95	8 bits	35	36313	28.2%	2.4%	5.5%
	1.95	16 bits	43	50190	25.5%	2.8%	1.9%
FFT 64	0.83	8 bits	40	47952	24.2%	10.1%	10.7%
	5.33	16 bits	48	56399	20.9%	11.8%	8.3%
SAD 8x8	3.81	8 bits	14	216	64.6%	6.9%	15.6%
	3.81	16 bits	22	384	58.9%	6.1%	15.8%

uniforme en termes de nombre de cycles d'horloge et cela, peu importe les dynamiques. Cette hypothèse impliquait par exemple qu'une multiplication entre deux nombres codés sur 16 bits projetée sur un multiplieur matériel 64 bits nécessitait la même latence pour produire le résultat qu'une multiplication entre 2 nombres codés sur 64 bits. Physiquement, le temps de calcul varie en fonction de la durée de propagation du MSB. Dans la réalité, la structure matérielle des opérateurs arithmétiques usuels est telle que la durée de propagation dans les opérateurs arithmétiques est généralement proportionnelle à la dynamique des entrées comme cela est illustré dans la figure 1.4.

Ces observations, confirmées par des simulations post-synthèse logique, nous ont permis de valider par exemple que la durée nécessaire à l'établissement des sorties d'une multiplication 16×16 bits sur un multiplieur 32×32 bits est de 25% à 50% inférieure au temps de réalisation d'une multiplication 32×32 en fonction de la technologie d'intégration. En conséquence, nous avons fait évoluer les algorithmes d'ordonnancement et d'assignation précédemment mis au point afin de favoriser une gestion fine du chemin critique pour les opérations pouvant en bénéficier.

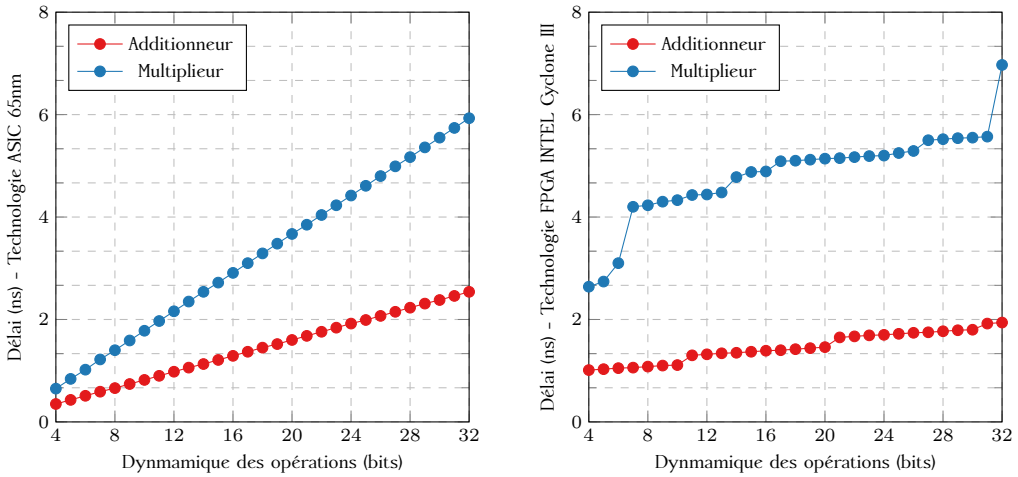


Figure 1.4 - Délai nécessaire à la production du MSB en fonction de la dynamique des données sur des opérateurs de taille 32×32 bits.

En effet, dans certains cas, comme pour l'opération de division, cette optimisation est impossible, car le temps de chemin critique est lié au calcul du LSB. Ces modifications algorithmiques se sont accompagnées d'un travail fastidieux de caractérisation des ressources matérielles sur différentes cibles technologiques. Cette tâche était nécessaire pour posséder des modèles mathématiques liant la dynamique des opérations à exécuter au temps de propagation maximum du résultat en sortie de l'opérateur.

L'impact de la modélisation précise du temps de propagation dans les ressources matérielles a été évalué [ri:LeGlla]. Des expérimentations réalisées avec un décodeur d'image JPEG, différentes contraintes de synthèse et différents formats de quantification des calculs sur cibles ASIC et sur circuits FPGA ont montré des gains pouvant atteindre 36% en termes de réduction de la latence lorsque l'architecture était contrainte matériellement. Par ailleurs, 25% de réduction de la complexité matérielle était obtenue lors de synthèses sous contrainte de latence.

1.3 Intégration des contraintes de sécurité dans la HLS

L'expertise scientifique et technique acquise dans les études précédentes couplée à une volonté partagée de collaborer avec Lilian BOSSUET, membre de l'équipe CSN de 2005 à 2010, et dont la thématique de recherche était liée aux problématiques de sécurité dans les systèmes reconfigurables, ont mis en évidence le manque de solutions de sécurisation de la propriété intellectuelle au sein des flots de synthèse HLS. En effet, ce thème était jusque-là peu traité dans le domaine de la conception automatisée d'architectures car la majorité des travaux de recherche à l'époque se concentrait sur les problématiques d'efficacité en termes de complexité matérielle, de latence et de débit [Ell00, Con05, Cou08, bluebook].

Le premier axe de recherche que nous avons identifié afin d'engager des actions conjointes était lié à la protection de la propriété intellectuelle. En effet, le commerce de circuits électroniques et d'IP matérielles était un marché en plein expansion. L'essor de ce marché était dû à l'adoption des méthodologies de conception basées sur la réutilisation (*design and reuse*) pour palier à l'augmentation importante de la complexité des systèmes intégrés. Les IP pouvaient être vendus à différents niveaux d'abstraction [ci:Cas04]. Toutefois, à tous les niveaux, les vendeurs d'IP doivent faire face au risque de copies illégales, de réutilisation ou de revente de copies frauduleuses. Un vendeur d'IP à plusieurs solutions pour prévenir, détecter et se protéger de la copie illégale de ses produits. La prévention s'appuie sur des moyens légaux (contrats, NDA, brevets) permettant des poursuites en cas d'attaques repérées et identifiées. La détection de copie s'appuie principalement sur des systèmes de marquage d'IP (watermaking ou fingerprinting) qui en cas de suspicion permet de prouver qu'une copie illégale a été effectuée. Différentes techniques de marquage applicables au niveau algorithmique [Ras99, Cha00] ou au niveau physique [Lac99, Jai03, Fan03, Sun06] avaient été proposées dans la littérature. Quelques travaux ciblaient [Tor00, Oli01, Kir98, Hon99] aussi le marquage automatisé d'architectures au niveau RTL en modifiant à moindre coût le contrôleur de l'architecture. Cependant les approches étaient peu flexibles et/ou limitées d'un point de vue applicatif. En conséquence, nous avons décidé de travailler à la définition d'une approche permettant d'introduire la marque sous une forme permettant de la retrouver facilement et de façon sure, tout en limitant le surcoût matériel.

Cette étude que nous avons financée sur fonds propres a été supportée en partie par les stages Simon LAMI (M1) et Jérémie VIGUIER (M2) respectivement en 2008 et 2009. Les travaux ont été

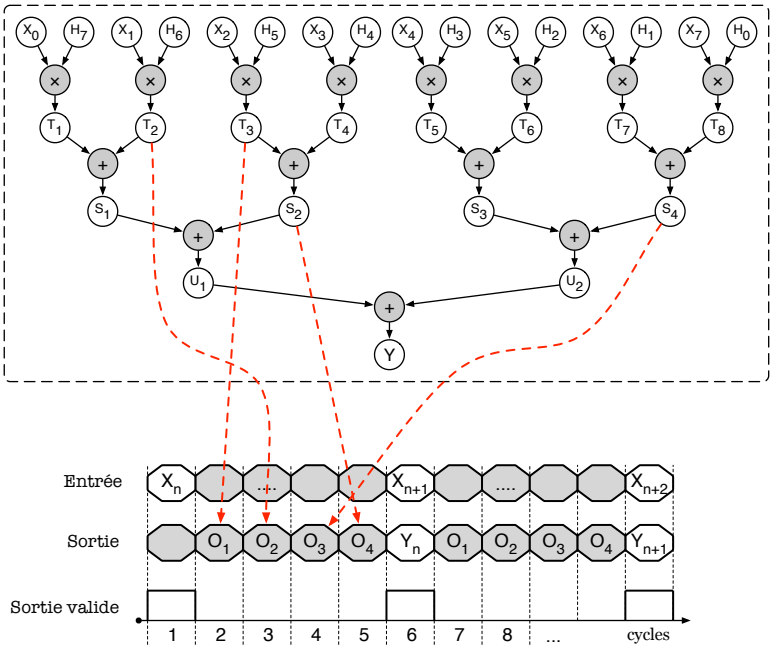


Figure 1.5 – Exemple didactique de marquage des sorties inutilisées du circuit à partir des résultats des calculs internes.

motivés par une observation du comportement des E/S des IP générées par les outils de synthèse d'architectures. Les ports de sortie des architectures ne sont utilisés qu'à la fin des traitements pour transmettre les résultats au système. Ainsi les architectures disposent de nombreux cycles d'horloge durant lequels les sorties ne sont pas employées dans le but de transmettre des informations utiles. Il est possible d'utiliser ces créneaux temporels pour faire apparaitre des valeurs numériques singulières. Ainsi le comportement normal du circuit n'est nullement altéré par la marque introduite. La figure 1.5 fournit comme exemple didactique un graphe modifié représentant un filtre FIR. Cet exemple permet de comprendre les liens possibles entre les valeurs intermédiaires disponibles dans le graphe et les valeurs qui peuvent être exposées sur les sorties du circuit durant les *slots* temporels inutilisés, ici $\{O_1, O_2, O_3, O_4\}$. C'est ce lien inattendu et singulier constitué par le routage des données internes vers les E/S qui constitue le marquage du circuit. La présence ou l'absence de marque peut être vérifiée facilement, même si l'IP est intégrée dans un système plus complexe grâce à une simple observation de ses E/S.

L'originalité de l'approche vient du fait que les marques transmises ne sont pas matériellement caractérisées, mais mathématiquement caractérisées. En effet, au lieu de faire apparaitre des données statiques en sortie du circuit, dont les valeurs, peuvent être *facilement* identifiées et modifiées, nous avons préféré lier les valeurs des marques aux données issues du traitement de l'algorithme. Ainsi, dans notre travail, la marque insérée dans le circuit est caractérisée par une relation mathématique issue de l'algorithme. Sa valeur évolue en fonction des entrées du circuit. À travers l'exemple présenté dans la figure 1.5, on comprend que le nombre de marques possibles croît avec la complexité de l'algorithme considéré. Plus il y a de calculs intermédiaires et de *slots* temporels disponibles, plus le nombre de marques possibles est conséquent. Des expériences ont été réalisées pour dénombrer le nombre de marques possibles sur des applications usuelles. Comme le prouve le tableau 1.3, le nombre de marques possibles croît exponentiellement avec la

Table 1.3 - Nombre de marques différentes pouvant être produites en fonction des contraintes imposées aux architectures lors du processus de synthèse HLS.

Application	Latence (cycles)	Nombre d'entrées	Nombre de sorties	Nombre de marques possibles
FIR 16	9	1	1	$\geq 2^{42}$
FIR 64	21	1	1	$\geq 2^{142}$
SSD 16	15	1	1	$\geq 2^{82}$
DCT 8	8	8	8	$\geq 2^{279}$
DCT 8×8	42	8	8	$\geq 2^{2195}$
SSD 8×8	71	4	1	$\geq 2^{634}$
FFT 512	282	50	50	$\geq 2^{17703}$

latence et le nombre de sorties disponibles dans l'architecture. Ce nombre important de marques possibles est un facteur important dans le domaine de la sécurité. En effet, il minimise les risques de collisions entre les marques.

Le marquage de l'application se traduit matériellement par la réutilisation ou la création de chemins de données supplémentaires entre les opérateurs matériels qui réalisent les calculs et les sorties du circuit. Cela implique l'insertion de multiplexeurs et la génération de nouveaux signaux de contrôle pour les piloter. Un exemple de circuit matériel généré et dans lequel une marque a été insérée est fourni en figure 1.6. Les éléments en pointillés sont ceux qui ont été ajoutés ou modifiés pour marquer le circuit.

Le flot traditionnel de synthèse d'architectures a été modifié [ri:LeGl2] pour permettre la génération et l'insertion de la marque dans l'architecture matérielle générée post-synthèse HLS. Deux étapes ont été ajoutées au flot traditionnel : la première étape analyse le modèle interne issu de la spécification algorithmique, afin de déterminer le nombre et la longueur des marques utilisables. La seconde étape réalise le marquage en ajoutant au circuit synthétisé les ressources matérielles nécessaires à l'implantation de la marque en adéquation avec les contraintes de l'utilisateur. Des expérimentations ont été menées sur circuit FPGA afin d'estimer l'impact de l'approche sur la complexité matérielle ainsi que sur la fréquence maximale de fonctionnement des architectures. Les résultats d'implantations matérielles sur un circuit FPGA de type Virtex-5

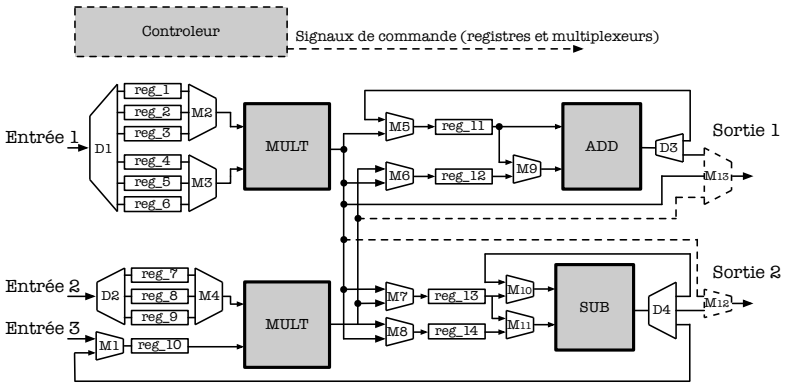


Figure 1.6 - Architecture matérielle générée dans laquelle une marque a été ajoutée. Les ressources matérielles modifiées ou insérées afin d'ajouter la marque sont représentées en pointillés.

montrent que, dans tous les cas, le surcoût en surface et la réduction de la fréquence maximale de fonctionnement de l'architecture liés au marquage sont négligeables (inférieur à 1%). Ceci est d'autant plus vrai que l'application est complexe.

Ces travaux de recherche qui ont donné lieu à la publication d'un article dans une revue internationale [ri:LeGl2], 2 publications dans des conférences internationales [ci:LeG07b, ci:LeGll] et une communication nationale [cn:LeG07a] se sont arrétés avec la mutation de Lilian BOSSUET en 2010 à Saint-Etienne.

1.4 Intégration de contraintes non fonctionnelles dans la HLS

Les travaux initiés dans le domaine de la protection intellectuelle ont mis en évidence le manque de considération de contraintes non fonctionnelles utiles et nécessaires à la protection, la vérification ou la mise au point d'architectures matérielles. Afin d'investiguer ce domaine de recherche, j'ai sollicité et obtenu un financement de thèse MNRT qui a permis le démarrage de la thèse d'Aurélien RIBON en 2009. Cette thèse a été co-encadrée avec Dominique DALLET. En parallèle de l'obtention de la bourse MNRT, un financement obtenu via un appel à projets BQR (Bonus Qualité Recherche) interne à l'université de Bordeaux a apporté les fonds nécessaires à l'achat de matériels (oscilloscope numérique et carte FPGA ML507).

Dans un premier temps, nous avons décidé de nous focaliser sur l'amélioration du processus de vérification et d'intégration des architectures matérielles générées. En effet, 70% du temps de développement d'un système est passé à déboguer les architectures développées et/ou leur intégration au sein du système [Bor07]. Afin d'aider les concepteurs, des approches méthodologiques avaient été développées au niveau RTL [Bha07]. Ces dernières permettent de réduire l'effort nécessaire à fournir lors de ces étapes rédhibitoires et pour remonter des symptômes détectés à la source de l'erreur.

L'approche nommée Assertion-Based Verification (ABV) [Fos03] appliquée au niveau RTL se base sur l'intégration d'assertions permettant de s'assurer que les hypothèses issues des spécifications ou des choix d'implantation réalisés en amont sont toujours vérifiées. Ces assertions décrites à l'aide d'instructions spécifiques au langage RTL employé ou à l'aide de langages dédiés tels que le langage PSL (*Property Specification Language*) permettent de formaliser des propriétés comportementales ou temporelles sans impacter les caractéristiques des circuits post-synthèse. Dans les deux cas, leur utilisation améliore fortement le pouvoir de détection au plus tôt des erreurs en s'appuyant sur la capacité des simulateurs et/ou des outils de preuves formelles à les détecter au plus près de la source.

L'usage d'assertions s'avère aussi bénéfique lors de la réutilisation d'IP en dehors de leur contexte original, car elles permettent de s'assurer de l'adéquation entre l'IP conçue et le nouveau système dans lequel elle doit s'intégrer d'un point de vue comportemental et temporel [Kun09]. L'insertion d'assertions peut être vue comme la déclaration de contrats formels entre les données d'entrée et de sortie de l'algorithme décrit, mais aussi comme une formalisation des relations (E/S) avec le reste du système. Elles permettent en effet de valider que les données reçues ou émises respectent les spécifications du système global. Cette approche de programmation est nommée *Design-By-Contract* (DbC) [Mey92] dans les travaux relatifs au développement logiciel où elle est couramment mise en application. Les outils de synthèse HLS [bluebook, Vitis-HLS, Catapult-HLS]

```
int linearize(int x) {

  // invariant (1)
  assert ( current_mode == LINEAR );

  // precondition
  assert ( X_MIN <= x && x <= X_MAX );

  int k = x * CONST_A + CONST_B;

  // internal assertion
  assert ( k > 0 );

  int y = k >> 1;

  // postcondition
  assert ( y > 0 );

  // invariant (2)
  assert ( current_mode == LINEAR );

  return y;
}
```

Figure 1.7 – Exemples d'assertions dans un code comportemental avant synthèse HLS.

suppriment automatiquement les assertions des descriptions comportementales. Nous avons orienté nos travaux dans ce domaine afin d'assurer la propagation des assertions du niveau comportemental dans lequel les notions de temps (cycle d'horloge) et d'espace (registre) n'existent pas au niveau RTL.

La figure 1.7 illustre les différents types d'assertions supportés au niveau comportemental. Comme cela a été évoqué précédemment, l'insertion de préconditions et de postconditions permet de vérifier les contraintes liées aux entrées reçues depuis le système dans lequel sera intégrée l'architecture générée ainsi que de vérifier les sorties. Des assertions peuvent aussi être utilisées pour vérifier le comportement de l'algorithme par exemple via l'analyse de données temporaires. Avec pour objectif de transformer ces assertions comportementales en moniteurs au niveau RTL, nous avons proposé une évolution du flot de synthèse HLS. En effet, l'utilisation d'assertions dans les codes sources comportementaux se traduit dans le modèle de représentation interne à l'outil par l'ajout d'opérations et de données temporaires comme cela est schématisé dans la

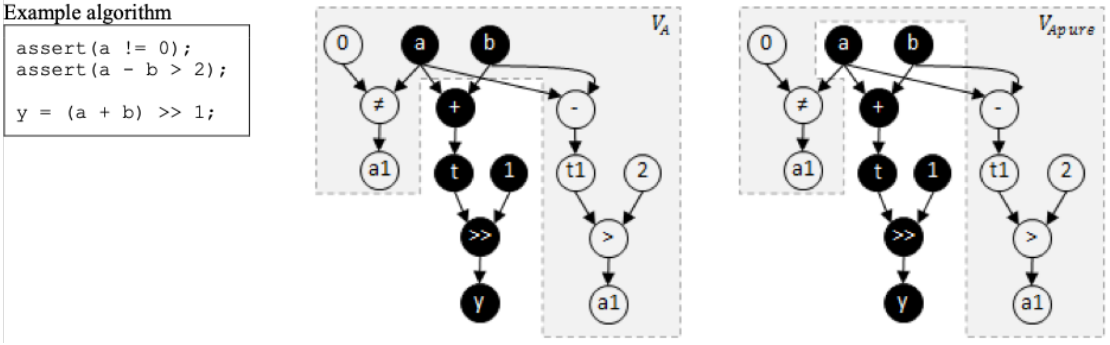


Figure 1.8 – Conversion des assertions comportementales sous forme de graphe de type DFG, et isolation de ces dernières afin de ne pas perturber le processus de synthèse HLS.

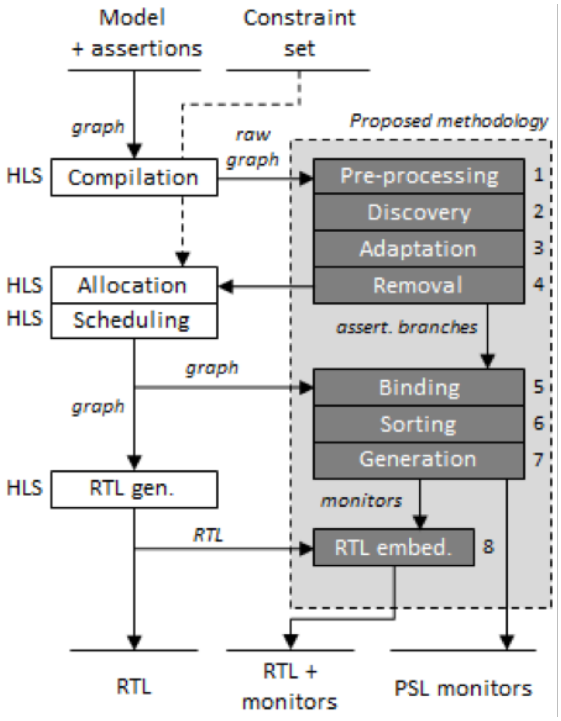


Figure 1.9 – Flot de synthèse d'architectures modifié afin de gérer les assertions décrites au niveau comportemental.

figure 1.8. Ces opérations et ces données temporaires, qui décrivent les conditions d'activation des moniteurs, ne doivent cependant pas impacter le déroulement du processus de synthèse HLS. Ainsi dans un premier temps, nous avons mis au point un algorithme permettant d'isoler les traitements et les données dédiés à la vérification des assertions de ceux nécessaires pour implanter le comportement de l'algorithme. Les branches liées aux assertions sont ensuite extraites du modèle lors des étapes de synthèse puis réintroduites ensuite comme cela est représenté dans la figure 1.9.

La réintroduction des assertions dans le modèle interne post-synthèse est une étape nécessaire liant les variables et les opérations décrites au niveau algorithmique à leurs projections au niveau matériel. En effet, au niveau architectural, les données sont conservées quelques cycles d'horloge dans des registres spécifiques impliquant, le cas échéant des traitements particuliers pour permettre le calcul d'assertions par la suite. L'architecture RTL est générée à partir du modèle post-synthèse incluant les informations architecturales et les assertions. La génération RTL se déroule en deux temps :

1. l'architecture RTL du circuit est générée de manière traditionnelle ;
2. en parallèle de la description RTL du circuit, au sein du même fichier, les moniteurs sont ajoutés afin d'implanter les opérations de vérification.

Ces moniteurs évoluent de manière synchronisée avec l'architecture (état du contrôleur, horloge). Pour s'assurer de la non-dégradation des performances post-synthèse logique, la description des moniteurs est délimitée par des directives *synthesis off/on* lorsqu'ils sont décrits en langage VHDL.

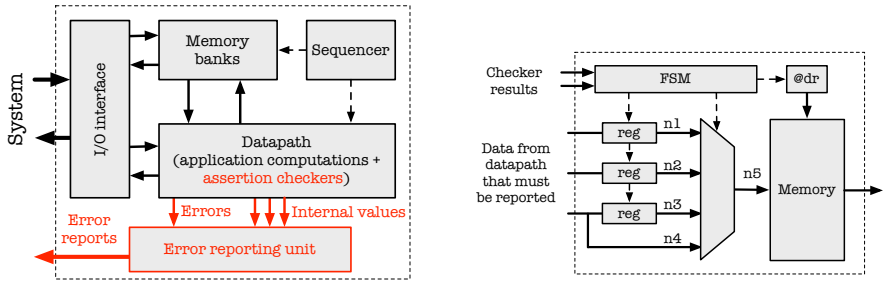


Figure 1.10 – Architectures matérielles (a) de l'architecture matérielle post-HLS inétgrant l'unité de gestion des assertions (b) de l'unité en charge de la mémorisation et de la restitution des déclenchements d'assertions et des données associées.

Différentes manières de spécifier les moniteurs au niveau RTL ont été évaluées afin de minimiser l'impact des moniteurs sur le temps de simulation. L'approche la plus intéressante que nous avons identifiée consiste à décrire l'ensemble des moniteurs dans un unique processus synchrone. Cette approche permet dans les exemples que nous avons évalués de limiter l'augmentation du temps de simulation à 1% du temps total. Ces travaux et les résultats obtenus ont été détaillés et valorisés dans [ci:Rib10].

Les travaux réalisés permettaient de faciliter la mise au point des systèmes lors des phases de simulation. En vue de permettre la détection d'erreurs de fonctionnement ou de tentatives d'attaques matérielles à l'exécution, nous avons dans un second temps débuté l'intégration de moniteurs matériels au sein des architectures RTL. Le verrou scientifique était lié à la minimisation des complexités matérielles et mémoire des moniteurs. Le choix a été fait de concevoir une unité spécialisée dans la gestion des moniteurs au sein de l'architecture comme cela est illustré dans la figure 1.10a. L'évaluation des conditions booléennes des assertions est confiée au chemin de données traditionnel pour mutualiser au mieux les ressources arithmétiques et logiques. À l'opposé, la mémorisation des données internes à l'architecture permettant de remonter aux causes du déclenchement des assertions est déportée dans l'unité de gestion des moniteurs. Cette dernière doit stocker temporairement dans ses registres internes les valeurs internes susceptibles d'être mémorisées si une assertion est déclenchée. En cas de déclenchement d'une assertion, l'ensemble des données à rapporter est stocké de manière structurée dans une mémoire RAM prévue à cet effet tel que cela est schématisé dans la figure 1.10b.

La conception de l'unité de gestion des moniteurs est contrainte par les choix réalisés lors de la synthèse du chemin de données : les cycles d'horloge ou les résultats des assertions sont connus, la durée de vie des données temporaires dans les registres, etc. La structuration des données dans les rapports au sein de la mémoire impacte fortement la capacité temps réel du système ainsi que son coût silicium. En effet, cette unité de mémorisation doit être capable de mémoriser en temps réel les données à rapporter au système tout en minimisant l'espace mémoire requis. Différentes stratégies de stockage de l'information ont été considérées comme illustré dans la figure 1.11. Les données peuvent être fractionnées ou concaténées en fonction de la largeur de la mémoire et des contraintes temporelles. Ces techniques permettent d'explorer l'espace des solutions afin d'identifier la largeur mémoire la plus faible qui permet de répondre aux contraintes temps réel. Ces approches permettent de minimiser l'occupation mémoire en contrepartie d'une augmentation de la complexité matérielle (logique de multiplexage et du

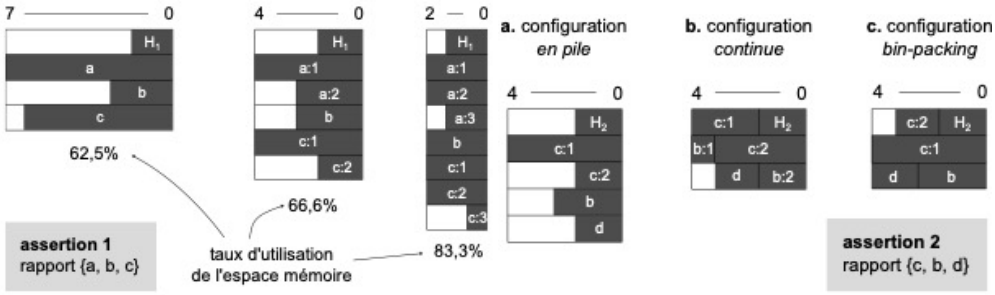


Figure 1.11 – Stratégies de mémorisation et de compression des données issues du déclenchement d'une assertion.

contrôleur) de l'unité représentée dans la figure 1.11.

L'ensemble des algorithmes développés et les résultats théoriques sont explicités dans le manuscrit de thèse d'Aurélien RIBON. Ces travaux ainsi que d'autres non détaillés ici ont donné lieu à 4 publications dans des conférences internationales [ci:Rib10, ci:LeG10b, ci:LeG10c, ci:Rib11], 2 publications en conférence nationale [cn:Rib10a, cn:Rib10b], 1 article dans un journal national [rn:LeG11] et une présentation invitée lors d'une journée scientifique organisée par le CNES [pn:LeG09b].

1.5 Conception de décodeurs ECC pour le standard 5G

Après avoir proposé pendant plus de 10 ans des solutions originales afin d'améliorer les méthodologies associées aux outils de synthèse HLS, j'ai souhaité en 2016 étudier l'efficacité et la pertinence de ces derniers dans le cadre de la conception de systèmes réels. En effet, les progrès réalisés par les dernières générations d'outils [Mart09, bluebook] tels que Vivado HLS [Vitis-HLS] de Xilinx ou Catapult-C [Catapult-HLS] de Mentor Graphics ont abouti à leur adoption par des acteurs industriels [Mig22, STM21]. Cela a motivé ma réflexion. J'ai décidé d'évaluer la pertinence et les performances atteignables lors de la conception de systèmes de communications numériques à des fins de prototypage et/ou d'implantation en temps réel. Ce thème de recherche couplant de la modélisation et des aspects méthodologiques appliqués aux systèmes de communications numériques issus du standard 5G 3GPP [5G] était en phase avec les travaux entrepris depuis 2011 avec Christophe JEGO comme cela est décrit dans les Chapitres 2 et 3. Ces travaux ont pu se développer grâce à une bourse de thèse MNRT dont a bénéficié Yann DELOMIER en octobre 2016.

La principale difficulté de ce travail de thèse visant à modéliser et à implanter des décodeurs matériels pour les codes LDPC et les codes polaires définis dans le standard 5G provenait de la pluralité des expertises à posséder d'afin d'espérer obtenir des architectures de décodage compétitives par rapport à celles développées et optimisées manuellement au niveau RTL. En effet, les codes correcteurs d'erreurs possèdent des propriétés intéressantes en termes de pouvoir de correction des erreurs. Ces propriétés sont obtenues aux dépens de structures algorithmiques de décodage complexes à implanter, aboutissant à des architectures matérielles très spécifiques. À l'inverse, les outils de synthèse d'architectures sont des outils (semi-)automatisés très dépendants des modèles algorithmiques décrits [bluebook, VivadoHLS]. La qualité des architectures générées

Table 1.4 – Synthèse des travaux relatifs à l'implantation de décodeurs LDPC matériels à partir de descriptions comportementales. Les descriptions sont issues d'implantations logicielles des décodeurs sur cibles CPU ou GPU.

Référence	Année	Outil HLS	Langage	Cible FPGA	Débit	Complexité matérielle
[Fal14]	2014	Altera OpenCL	OpenCL	Stratix-5	++++	++++
[Sch13a]	2016	Vivado HLS	SystemC	Spartan-6	+	++
[And14]	2014	MaxCompiler	Java	Stratix-5	++	++++
[And14]	2014	Altera OpenCL	OpenCL	Stratix-5	++	++++
[Roh16]	2016	SDSoC	C/C++	Artix-7	+	++

dépend fortement de la compréhension du concepteur du comportement intrinsèque de l'outil HLS employé (sémantique d'entrée, algorithmes de synthèse, technique de guidage de l'outil...). De plus, elles sont elles sont contraintes par les parallélismes de calculs et de tâches. Dans ce conteste, nos travaux se sont d'abord focalisés sur la description d'algorithmes de décodage pour les codes LDPC.

Des travaux antérieurs s'étaient déjà intéressés à l'usage d'outils de synthèse HLS pour implanter des décodeurs LDPC comme cela est indiqué dans le tableau 1.4. Les équipes de recherche s'étaient focalisées sur la possibilité de réutiliser des codes logiciels développés à l'origine pour implanter des décodeurs LDPC sur des cibles programmables de type CPU [Sch13a, Roh16] et GPU [And14, Fal14]. L'objectif visé était la réutilisation complète ou partielle des descriptions comportementales afin de faciliter le portage d'applications SDR déployées sur des plateformes associant un CPU avec un ou plusieurs GPUs vers des systèmes moins énergivores associant un CPU avec un FPGA.

Ces travaux ont démontré qu'il était possible d'obtenir des débits proches de ceux atteints par des architectures dédiées implémentées sur technologie ASIC [Mau16] et/ou circuit FPGA [Hai16] en offrant des temps de développement plus courts et une grande flexibilité. Toutefois, les complexités matérielles et les consommations énergétiques étaient bien plus importantes que celles des architectures dédiées développées manuellement au niveau RTL. Cet état de fait était prévisible vu qu'aucune transformation ou optimisation significative n'était appliqué aux modèles logiciels développés pour les cibles CPU/GPU [Sch13a, Roh16, And14, Fal14]. Or, les paradigmes d'implantation et d'optimisation sont orthogonaux :

— les compilateurs logiciels ciblant des architectures multi-cœurs SIMD ou many-cœurs tentent d'exploiter au mieux une architecture matérielle pour implanter le comportement décrit ;

— les outils de synthèse d'architectures proposent des architectures matérielles de niveau RTL implantant efficacement le comportement décrit.

Pour exploiter le parallélisme intrinsèque du processus de décodage des codes LDPC, les travaux de recherche ciblant des implantations sur GPU s'étaient focalisés sur des algorithmes de décodage simplifiés de type Min-Sum avec un ordonnancement des calculs par inondation [And16]. Cette formulation algorithmique a pour avantage de mettre en exergue un parallélisme de calculs élevé adapté aux matrices de cœurs FPU des architectures GPU. De plus la formulation algorithmique construite à partir d'un ensemble de nids de boucles imbriquées s'adapte bien

```
1
2  template<int Q> /* fonction de 1er niveau */
3  sc_int<Q> f_abs(const sc_int<Q> a) {
4      if( f_sign<Q>(a) == 1  ) return (sc_int<Q>)(-a);
5      else                     return (sc_int<Q>)( a);
6  }
7
8
9  template<int Q> /* fonction de 1er niveau */
10 sc_uint<1> f_sign(const sc_int<Q> llr){
11     const sc_uint<1> res = (sc_uint<1>)llr[Q-1];
12 }
13
14
15 template<int Q> /* fonction de 2nd niveau */
16 sc_int<Q> F(const sc_int<Q> a, const sc_int<Q> b) {
17     const sc_uint<1> sign1  = f_sign<Q>(a);
18     const sc_uint<1> sign2  = f_sign<Q>(b);
19     const sc_int<Q>  mag1   = f_abs <Q>(a);
20     const sc_int<Q>  mag2   = f_abs <Q>(b);
21     const sc_int<Q>  resMag = f_min <Q>(mag1, mag2);
22     const sc_int<Q>  minus  = -resMag;
23     return (sign1 != sign2) ? minus : resMag;
24 }
25
26
27 template<int P, int Q>  /* fonction de 3eme niveau */
28 sc_bigint<P * Q> VECT_F(const sc_bigint<P * Q> a, const sc_bigint<P * Q> b){
29     sc_bigint<P * Q> c;
30     for(int i = 0; i < P; i += 1)
31     {
32         const sc_int<Q> oa = (sc_int<Q>)a.range(Q * (i+1) - 1, $\mm{Q}$ * i);
33         const sc_int<Q> ob = (sc_int<Q>)b.range(Q * (i+1) - 1, $\mm{Q}$ * i);
34         c.range(Q * (i + 1) - 1, $\mm{Q}$ * i) = F<Q>(oa, ob);
35     }
36     return c;
37 }
```

Listing 1.1 - Exemple de spécifications de fonctions de calculs pour le décodage des ECC au niveau des couches scalaires et vectorielles.

aux capacités des langages CUDA ou OpenCL. Toutefois cette formulation de l'algorithme de décodage nécessite tout d'abord une bande passante mémoire importante mettant en œuvre des accès irréguliers aux données tandis qu'elle n'offre qu'une vitesse de convergence limitée vis-à-vis d'autres formulations plus complexes à mettre en œuvre en logiciel, mais d'usage dans les architectures RTL [Mar10, ri:LeG14b, Hai16, Mau16].

En rupture vis-à-vis de l'état de l'art, nous nous sommes intéressés à la formulation de l'algorithme de décodage Min-Sum basé sur un ordonnancement par couches horizontales. Nous possédions déjà (cf. Chapitre 2) des descriptions logicielles efficaces de cet algorithme pour des architectures multi-cœurs [ri:LeG14c, ri:LeG15c, ri:LeG16a]. Ces modèles logiciels étaient basés sur une représentation en virgule fixe de l'information équivalente à celle utilisée dans les décodeurs LDPC matériels. Pour pouvoir implanter ces modèles comportementaux développés à l'origine pour des architectures multi-cœurs SIMD exploitant les instructions intrinsèques définies par INTEL [SIMD], nous avons tout d'abord développé des couches d'abstraction *SIMD-like*. Elles permettaient de générer des ressources vectorielles afin de simplifier l'adaptation des décodeurs logiciels.

La couche d'abstraction SIMD pour les codes LDPC et les codes polaires est commune. Elle est décomposée en 3 sous-couches comme cela est illustré dans le Listing 1.1. La première couche décrit les spécifications bit accurate des opérations élémentaires telles que l'addition saturée, la valeur absolue (lignes 2-6) et l'extraction du signe (lignes 9-12). Chaque opération

peut être configurée finement en termes de dynamique des données (**Q**). La seconde couche décrit les fonctions scalaires nécessaires pour l'implantation des algorithmes de décodage des codes LDPC et des codes polaires. Ces fonctions (par exemple, la fonction **f**, lignes 15-24) qui exploitent la première couche sont aussi personnalisables en fonction de la largeur des données (**Q**). La troisième et dernière couche décrit enfin les fonctions SIMD utilisées dans les modèles comportementaux des décodeurs. La troisième couche est configurable via deux paramètres **P** et **Q**. Comme le montre le Listing 1 (lignes 27- 37), le paramètre **P** permet d'allouer et d'exécuter en parallèle **P** fonctions de type **f** qui traitent des valeurs codées sur **Q** bits.

Cette couche d'abstraction a facilité la réutilisation des modèles logiciels déjà en notre possession [ri:LeGl4c, ri:LeGl5c, ri:LeGl6a] sans éviter toutefois des modifications des modèles pour gérer les E/S. Les premières expérimentations que nous avons menées à l'aide de l'outil Xilinx Vivado HLS, publiées dans [ci:Dell8c], ont confirmé les résultats présents dans la littérature [Sch13a, Roh16, And14, Fal14]. En effet, même si des niveaux de performance supérieurs à ceux obtenus à partir de modèles pour GPU [And14, Fal14] ont été obtenus, ceux-ci ne supportent pas la comparaison avec des architectures matérielles décrites au niveau RTL [Hai16, Mau16]. Ce différentiel de performances s'expliquait par :

— le choix de la stratégie de parallélisation exploitée dans notre modèle comportemental [ri:LeGl4c, ri:LeGl5c, ri:LeGl6a] qui était basé sur le décodage de plusieurs trames en parallèle (parallélisation inter-trames). Cette stratégie simple à mettre en œuvre nécessite plus de ressources matérielles (LUT, blocs RAM). Elle est en phase avec les capacités des systèmes multi-cœurs possédant des giga-octets de mémoire RAM mais rédhibitoire sur sur des circuits FPGA ;

— l'utilisation d'une description en langage C/SystemC basée sur l'utilisation d'un seul processus pour décrire l'ensemble de l'algorithme de décodage. Cette description monotâche ne met pas en exergue la concurrence entre les kernels de calculs successifs. Cela empêche de fait son exploitation par l'outil de synthèse (Vivado HLS).

Ces observations nous ont amenés à faire évoluer le modèle comportemental. La seconde itération du modèle comportemental décrivant le décodage des codes LDPC a été conçue par mimétisme avec les architectures matérielles décrites au niveau RTL [Mar10, ri:LeGl4b, Hai16, Mau16] afin de :

1. mettre en exergue la concurrence entre les kernels de calculs et les accès aux mémoires. L'objectif consistait à aider l'outil à générer des architectures plus proches de celles développées pour implanter des décodeurs LDPC semi-parallèles. Le processus décrivant l'algorithme de décodage a été scindé en 5 tâches disjointes. Ces tâches décrites en langage SystemC via des sc_thread communiquaient au travers de FIFO (sc_fifos) tel que cela est schématisé dans la figure 1.12.
2. d'exploiter le parallélisme de calcul au sein du processus de décodage lorsqu'une trame ou un ensemble réduit de trames (≤ 3 trames) est décodé. Cette stratégie a pour vocation de permettre une exploitation efficace de l'architecture sans engendrer une explosion de la complexité mémoire contrairement au premier modèle développé (inter-trames).

Ce raffinement du modèle comportemental a permis de contraindre l'outil Vitis HLS à uniquement paralléliser et pipeliner les nids de boucle de manière efficace, aboutissant à une exploitation optimale de l'architecture générée. L'architecture ainsi obtenue est similaire à celle des architectures

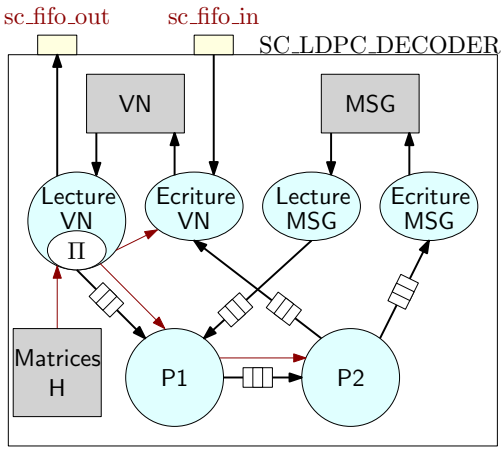


Figure 1.12 – Représentation structurelle gros grain de l'architecture matérielle issue du modèle de décodeur LDPC décrit en SystemC.

dédiées décrites au niveau RTL : une fois le processus de décodage débuté, le pipeline de traitement de l'architecture est exploité au maximum de ses capacités.

Les résultats d'implémentation post-placement et routage des architectures produites à partir de ce modèle par l'outil Vivado HLS [ri:Del20a] ont permis de démontrer la qualité du modèle développé ainsi que la maturité des méthodologies basées sur l'utilisation d'outils de synthèse HLS. Ces affirmations sont attestées par les résultats fournis dans le tableau 1.5 où Γ indique le débit en Mbps des décodeurs tandis que μs fournit les latences. Une analyse fine des performances obtenues en pratique par ces architectures a mis en évidence que les performances mesurées sont équivalentes à celles estimées de manière théorique pour des architectures de type semi-parallèles décrites manuellement au niveau RTL [ri:Del20a].

Le positionnement des travaux réalisés par rapport à la littérature nous est favorable. Les débits atteints par notre approche sont de 4× à 40× supérieurs à ceux des travaux antérieurs [Sch13a, Roh16, And14, Fal14]. De plus, la complexité matérielle des architectures générées est de 2× à 20× plus faible pour des niveaux de flexibilité équivalents. Vis-à-vis des architectures dédiées de décodeurs LDPC, les niveaux de performance sont similaires comme cela est détaillé

Table 1.5 – Performances des décodeurs LDPC-QC pour un parallélisme intra-trame à partir du second modèle comportemental sur un circuit FPGA Virtex-7 (10 itérations de décodage).

	Code LDPC			Fréquence cible = 300 MHz							
	N	**R**	**Z**	F_{max}	Δ	Γ	μs	LUT	FF	Slices	RAM
5G 3GPP	1024	3/4	32	230	3910	181	17	7719	14673	3273	24
	2048	3/4	64	208	3910	328	19	14402	28722	6637	43
	4096	3/4	128	195	3910	614	20	30292	56664	12547	82
	8192	3/4	256	129	3910	812	30	64535	112557	26655	159
	2376	8/9	88	219	2590	602	12	23063	39260	9444	57
	4048	1/2	88	223	6634	409	30	23611	39276	9389	72
	4752	8/9	176	183	2590	1010	14	48775	77639	18984	112
	8096	1/2	176	183	6634	672	36	48775	77639	18984	112

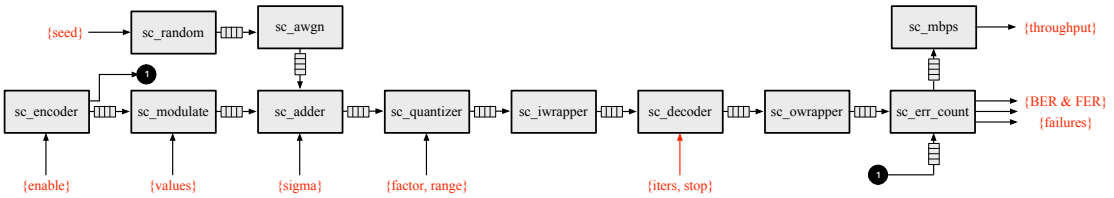


Figure 1.13 - Système d'évaluation décrit de manière comportementale et implanté sur circuit FPGA grâce à l'utilisation de la synthèse HLS.

dans [ri:Del20a]. Il est à noter que les architectures matérielles générées dans cette étude ont été validées par prototypage sur circuit FPGA. Le système schématisé dans la figure 1.13 permettant de caractériser les décodeurs LDPC (débits, pouvoir de correction d'erreurs, etc) a été modélisé et implémenté à l'aide de modèles comportementaux. Pour atteindre des débits de plusieurs Gbps ($\geq$ 100 Mbps), différents algorithmes et approches de parallélisation ont été mis en œuvre pour permettre par exemple la génération d'un bruit blanc gaussien de haute qualité comme cela est détaillé dans [ci:Del18b].

Confortés par les résultats obtenus pour le décodage des codes LDPC, nous avons ensuite étudié la seconde famille de codes correcteurs d'erreurs présente dans le standard 5G : les codes polaires. Les algorithmes de décodage de codes polaires proposés dans la littérature (ex : SC [Ari08, Ari09], SC-list [Tal15]) sont basés sur le parcours récursif d'un arbre binaire dans lequel le parallélisme de calcul varie dans l'intervalle $[log2(\mathbf{N}) \rightarrow 1]$.

La principale difficulté en termes de modélisation et d'implantation consistait à décrire efficacement les algorithmes de décodage pour permettre la génération d'architectures semi-parallèles configurables (dynamique des données, niveaux de parallélisme, etc.) sans exploiter la récursivité pour décrire le parcours séquentiel de l'arbre binaire. La récursivité était usuellement employée pour l'implantation logicielle de décodeurs (SC [ri:LeGl5a], SC-list [Leo19] et SC-SCAN [ri:LeGl9a]) de codes polaires sur circuit CPU. Cependant cette fonctionnalité n'était pas supportée par les outils de synthèse HLS [Vitis-HLS, Catapult-HLS, bluebook]. Nous avons donc transformé la formulation algorithmique à base de récursivité sous la forme d'une description comportementale basée sur l'utilisation d'un automate à pile. Cet automate composé de quatre états principaux (cf. figure 1.14a) et d'une pile permet d'implanter la descente du sommet de l'arbre binaire vers la première feuille de l'arbre puis le parcours progressif de toutes les feuilles de l'arbre. Lors du parcours de cet arbre binaire, à chaque changement de nœud, $\mathbf{N}/(\log_2(\mathbf{N}) - \mathbf{E})$ calculs sont réalisés pour l'étage $\mathbf{E}$ auquel est situé le nœud. La nature des $\mathbf{N}/(\log_2(\mathbf{N}) - \mathbf{E})$ calculs à réaliser ($\mathbf{f}$, $\mathbf{g}$ ou $\mathbf{h}$) dépend de l'orientation de l'arc comme cela est détaillé dans [Ari08, Ari09, ri:Del20b].

Pour atteindre les débits des architectures matérielles rapportées dans la littérature [LeRl2a, LeRl2b, Gia13, ci:LeGl6b], l'automate a été enrichi d'états complémentaires (figure 1.14b) permettant un élagage à la volée des branches de l'arbre [Gia13]. L'activation de l'élagage dans le modèle, et donc dans les architectures matérielles est paramétrable en amont de la synthèse HLS grâce à l'utilisation de directives traitées par le préprocesseur. Ce paramètrage du modèle permet d'adapter au mieux les architectures générées aux spécificités des décodeurs des codes polaires.

De multiples expérimentations ont permis de caractériser l'évolution de la complexité matérielle et du débit de décodage en fonction des paramètres algorithmiques supportés : la taille du code ($\mathbf{N}$), le rendement de codage ($\mathbf{R}$), les formats de quantification ($\mathbf{Q}$), le nombre d'unités de calcul

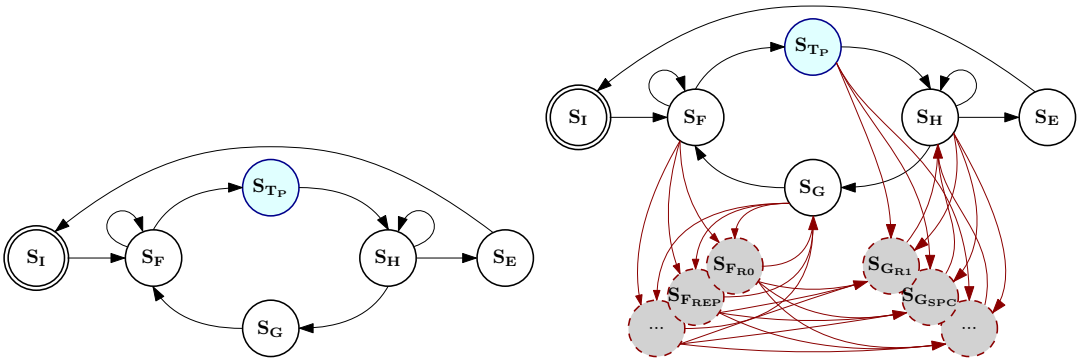


Figure 1.14 – Machines à états finis utilisées au niveau comportemental pour décrire l'algorithme de décodage SC (a) sans élagage et (b) avec élagage.

(**P**) et les optimisations d'élagage. Les résultats obtenus lors de ces expérimentations évoluaient de manière cohérente au regard des estimations théoriques. Le modèle comportemental de part sa flexibilité permet une vaste exploration de l'espace des solutions comme l'illustre le graphique fourni en figure 1.15. Ce dernier illustre une partie des compromis architecturaux possibles pour l'implantation d'un décodeur SC traitant des trames de taille $\mathbf{N} = 1024$ sur un circuit Xilinx Virtex-7 (xc7vx485tffg1761-2).

Les architectures obtenues grâce à l'approche proposée sont de $2\times$ à $4\times$ fois moins efficaces en termes de débit et/ou de complexité que celles décrites au niveau RTL [ci:Del19a, ri:Del20b]. Cet écart s'explique principalement par la manière dont l'outil Vivado HLS implante matériellement les machines d'états décrites à l'aide de la sémantique *switch-case*. Lors de l'implantation, des états de synchronisation et de détection d'erreurs sont insérés automatiquement lors des transitions. Ces états sont très pénalisants dans le cadre du décodage de codes polaires, car les quelques cycles d'horloge de pénalité qu'ils induisent ne sont pas négligeables vis-à-vis du temps passé dans les états de calcul.

Les niveaux de performance élevés lors de l'implantation de décodeurs de codes polaires basés sur l'algorithme de décodage SC nous ont incités à poursuivre les travaux pour supporter l'algorithme de décodage SCL [Tal15]. Cet algorithme améliore notablement les performances de décodage. Les décodeurs SCL possèdent une complexité calculatoire bien supérieure à celle des décodeurs SC en raison du traitement de $\mathbf{L}$ listes de $\mathbf{N}$ données en parallèle, mais aussi à cause de l'opération de tri & sélection ($2\mathbf{L} \rightarrow \mathbf{L}$) à réaliser lors du traitement de chacune des feuilles de l'arbre [Tal15, Sar16a]. Le premier verrou que nous avons identifié était lié à l'implantation efficace de l'opération de tri et de sélection ($2\mathbf{L} \rightarrow \mathbf{L}$) à partir d'une description comportementale. Une étude des différents algorithmes de tri efficaces dans le cadre du décodage SCL des codes polaires nous a encouragés à sélectionner un tri à bulles pour des raisons de flexibilité et de performance contrairement à ce que vous avions proposé pour des implantations de niveau RTL [ci:LeG20a]. À partir du modèle comportemental du décodeur SC et de la description comportementale des fonctions de tri, nous avons développé un nouveau modèle comportemental générique. Cette description comportementale supportant une configuration avant synthèse des paramètres $(\mathbf{N}, \mathbf{R}, \mathbf{L}, \mathbf{P})$ a été dans un second temps évaluée sur circuit FPGA. Les courbes présentées dans la figure 1.16, issues de [ri:Del20b], positionnent les performances d'implantation d'un décodeur SC lorsque $\mathbf{N} = 1024$ et $\mathbf{P} = 2$ par rapport à celles de décodeurs SCL équivalents avec $\mathbf{L} = \{2, 4, 8, 16, 32\}$. La complexité d'implantation croît linéairement avec le nombre de

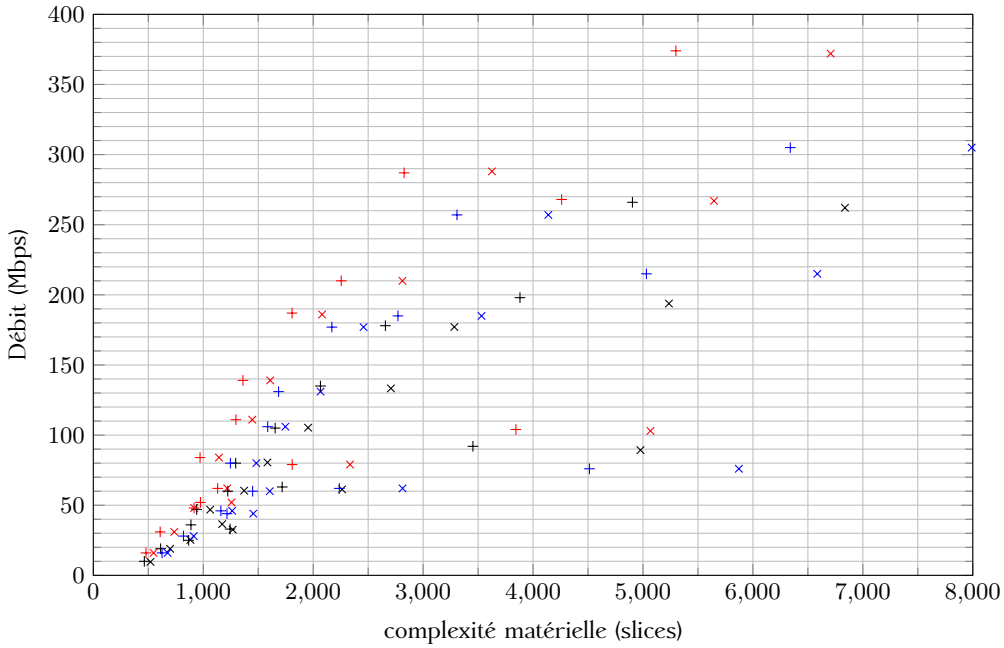


Figure 1.15 – Exemples de compromis architecturaux obtenus pour les architectures de décodeurs SC avec $\mathbf{N} = 2^{15}$. Les marques indiquent le format de quantification choisi ($\times = 8$ bits, $+ = 6$ bits). Les couleurs indiquent la fréquence de fonctionnement fixée avant HLS (noir = 100MHz, rouge = 200 MHz et bleu = 300 Mhz).

listes. Le débit décroit du même ordre de grandeur en raison du temps nécessaire pour tirer les listes et sélectionner les chemins les plus probables. Ces résultats préliminaires, n'incluant pas par exemple l'élagage dynamique des branches, démontrent la faisabilité de l'approche. Cependant afin de concevoir des architectures comparables à celles de l'état de l'art, des efforts d'optimisations supplémentaires sont nécessaires.

Les travaux de thèse réalisés par Yann DELOMIER concernant l'implantation de décodeurs ECC pour le standard 5G ont été soutenus en juin 2020. Ces travaux ont été valorisés par 2 articles dans des journaux internationaux [ri:Del20a, ri:Del20b] et 5 publications dans des conférences internationales [ci:Del18a, ci:Del18b, ci:Del18c, ci:Del19a, ci:LeG20a].

1.6 Diversification des domaines applicatifs

En 2014, j'ai initié une collaboration avec Yannick BORNAT, un collègue du groupe Bioélectronique du laboratoire IMS dont les thématiques de recherche nécessitent le prototypage et/ou la conception de systèmes numériques en temps réel traitant des signaux biologiques. Ces travaux avaient pour objectifs de simplifier la conception de systèmes numériques tout en apportant de la flexibilité et de la scalabilité grâce à l'élévation du niveau d'abstraction. Pour initier puis entretenir cette collaboration débutée sur fonds propres au travers du stage de master 2 de Rima TABESSI en 2014, nous avons sollicité et obtenu en 2017 le financement d'un stage de niveau master 2 (Safouane NOUBIR) sur un appel à projets interne au laboratoire (GPU SIS). Ces travaux ciblant la conception et le prototypage de systèmes d'analyse temps réel de l'activité de

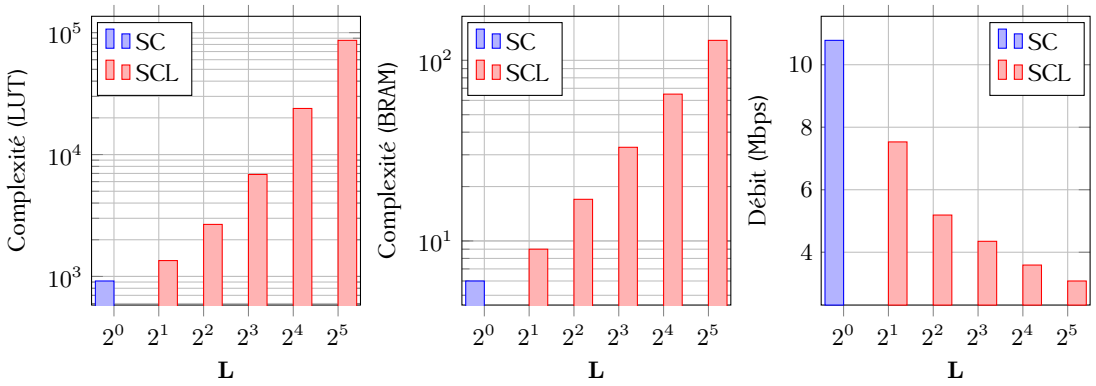


Figure 1.16 – Caractéristiques des décodeurs basés sur l'algorithme de décodage SCL générés pour $N = 1024$, $R = 50\%$, $\mathbf{P} = 2$ sans optimisation d'élagage.

cellules biologiques sur circuit FPGA ont mis en évidence l'intérêt de l'approche méthodologique basée sur des modèles comportementaux et la pertinence de la collaboration. Les résultats du stage ont été valorisés par 2 publications en conférences internationales [ci:Nou18a, ci:Nou18b]. L'absence de financement de thèse pour supporter ces travaux pluridisciplinaires n'a pas permis de continuer immédiatement cette collaboration. Ce lien entre les 2 groupes de recherche a été relancé grâce à un financement équivalent obtenu en 2020. Cela a permis de financer dans un premier temps le stage de Clémence GILLET que j'ai co-encadré avec Adrien VINCENT, stage qui s'est poursuivi en thèse via un financement MNRT. Les travaux ont débuté en 2020. Ils visent à la conception de systèmes numériques implantant des réseaux de neurones neuro-inspirés (Spiking Neural Networks ou SNN) [Bol9]. Dans un premier temps, afin de permettre l'étude et le déploiement de SNN dans des systèmes embarqués temps réel, nous avons œuvré à la mise au point d'une couche d'encodage générique et flexible pour adapter les signaux temporels issus de capteurs usuels (caméra, électrodes, etc.) sous forme d'évènements (*spike*) [ci:Gil22a]. Ces travaux constituent la brique de base nécessaire pour l'évaluation et la conception d'architectures numériques implantant des réseaux de type SNN.

En parallèle de cette étude, les niveaux de performance atteints dans le cadre de la thèse de Yann DELOMIER ont démontré que la conception de tout ou partie d'un système matériel sur FPGA à base de modèles comportementaux est possible. Ces résultats m'ont motivé à diversifier les cadres applicatifs au travers d'une collaboration industrielle avec la société ARELIS, société avec laquelle Jérémie CRENNE était en relation dans le cadre d'un projet FUI. Cette collaboration a débuté par l'encadrement de deux stages de niveau master 2, (Rémi FUMERON et Hugues ALMORIN) en 2019. L'objectif des stages respectifs était d'étudier l'intérêt de l'utilisation de la synthèse HLS et de la reconfiguration dynamique partielle pour la conception de systèmes de traitement radar. Les résultats encourageants que nous avons obtenus nous a décidé à poursuivre la collaboration industrielle ainsi initiée au travers d'une thèse CIFRE que je co-encadre avec Christophe JEGO et Jérémie CRENNE. Dans ce cadre collaboratif, Hugues ALMORIN travaille depuis 2020 sur le développement d'un environnement automatisé de modélisation, d'exploration et d'implantation de chaînes de traitement du signal pour des applications radar à base d'outils de synthèse HLS. Les premiers travaux se sont focalisés sur la modélisation, la parallélisation et l'implantation de l'opération de FFT sur circuit FPGA [ci:Alm22a]. Cette étape se justifie par la prévalence de sa complexité calculatoire dans les systèmes de traitement radar. La suite de l'étude vise à généraliser les méthodes développées pour la FFT à l'ensemble des traitements

Table 1.6 - Liste des encadrements scientifiques relatifs aux travaux relatifs relevant de la *modélisation et conception automatisée d'architectures*.

Nom	Statut	Période	Financement	Co-Encadrant(s)	% encadr.
Aurélien RIBON	PhD	2009-12	MNRT	D. DALLET	50%
Yann DELOMIER	PhD	2016-20	MNRT	C. JEGO & J. CRENNE	33%
Hugues ALMORIN	PhD	2020-...	CIFRE	C. JEGO & J. CRENNE	33%
Clémence GILLET	PhD	2020-...	MNRT	S. SAIGHI & A. VINCENT	33%
Jérémie VIGUIER	Master 2	2015	Fonds propres	–	100%
Rima TABESSI	Master 2	2014	Fonds propres	Y. BORNAT	50%
Vincent PIGNOLY	Master 2	2016	GdR ISIS	–	100%
Safouane NOUBIR	Master 2	2017	GPU SIS	Y. BORNAT	50%
Rémy FUMERON	Master 2	2019	ARELIS	–	100%
Clémence GILLET	Master 2	2020	GPU SIS	A. VINCENT	50%
Oussama AIT SIDI ALI	Master 2	2021	ANR QCSP	–	100%
Léa VOLPIN	Master 2	2022	ANR QCSP	–	100%

du signal nécessaires à la conception d'un système radar réel (entrelaceurs, filtres, etc).

1.7 Bilan des travaux

Dans cette partie est résumé l'ensemble des encadrements et des projets réalisés dans le cadre des activités de recherche en relation avec la *modélisation et conception automatisée d'architectures*. Il est à souligner que j'ai délibérément choisi de me focaliser sur un sous-ensemble des travaux menés. Ce dernier me semble avoir les contributions les plus notables.

Les encadrements de doctorants et d'étudiants de niveau Master 2 dont les travaux sont associés à cet axe de recherche sont récapitulés dans le tableau 1.6. Il est à noter que durant mes premières années de recherche, de nombreux étudiants de niveau Master 1 non mentionnés ici ont participé aux développements de l'outil *GraphLab*.

Une synthèse des valorisations scientifiques de cet axe est fourni ci-dessous :

— 9 articles dans des revues internationales à comité de lecture ;
— 2 articles dans des revues nationales à comité de lecture ;
— 32 articles dans des conférences & workshops internationaux avec actes ;
— 8 articles dans des conférences ou workshops nationaux avec actes ;
— 2 présentations lors de workshops nationaux ou journées du GdR.

L'ensemble des travaux présentés s'est déroulé dans le cadre d'activités partenariales ou de contrats de recherche publics. Les contrats relatifs à cet axe de recherche sont fournis ci-dessous :

— **ALIPTA** (projet RNTL) (2002-2005), Valorisation du rôle de la synthèse d'architectures dans les flots de développement systèmes, Budget UBS : 140 K€ ;
— **SCERNE** (PEA DGA) (2008-2009), Simulation de Chaînes d'Émission/Réception Nouvelle génération, Budget IMS : 108 K€ ;
— **ARDMAHN** (ANR ARPEGE) (2009-2012), Architecture Reconfigurable Dynamiquement et Méthodologie pour l'Auto-adaptation en Home Networking, Budget IMS : 50 K€ ;

— **ModCom** (GreenSoC) (2015-2018), Modélisation rapides et flexibles des communications dans les SoCs en SystemC (Accompagnement de la thèse de Guillaume DELBERGUE), Budget IMS : 45 K€ ;
— **SIMPADEC-NB** (GdR ISIS) (2016-2017), Simulation et IMplantation PArallèle de DECodeur Non Binaire, Budget IMS : 7 K€ ;
— **HLS for radar processing** (ARELIS) (2020-2023), Définition et mise en œuvre d'un flot de conception pour l'implantation de chaînes de traitement radar à partir de modèles comportementaux (Accompagnement de la thèse d'Hugues ALMORIN), Budget IMS : 30 K€.

2 Conception et utilisation d'architectures programmables

2.1 Introduction

Ce second chapitre contient une synthèse des projets que j'ai menés dans l'axe de recherche sur la conception et l'utilisation d'architectures programmables. Je décris brièvement le contexte de cet axe de recherche et introduis les travaux menés. Je donne également son organisation au cours de la dernière décennie. Afin d'illustrer l'activité de recherche, la suite de ce chapitre présente plusieurs études menées dans le domaine. La figure 2.1 synthétise les étudiants encadrés lors de ces travaux. La première section du chapitre décrit la conception d'un cœur de processeur adapté aux applications nécessitant des fonctionnalités d'analyse protocolaire. La conception d'une architecture matérielle programmable flexible pour le décodage des codes correcteurs d'erreurs LDPC est détaillée dans la deuxième section. La troisième section traite quant à elle du développement de techniques de parallélisation des algorithmes de décodage des ECC sur des cibles programmables de type multi-cœurs et many-cœurs dans un contexte radio-logicielle. La dernière section résume brièvement les actions de recherche en cours avant qu'une synthèse des activités soit réalisée.

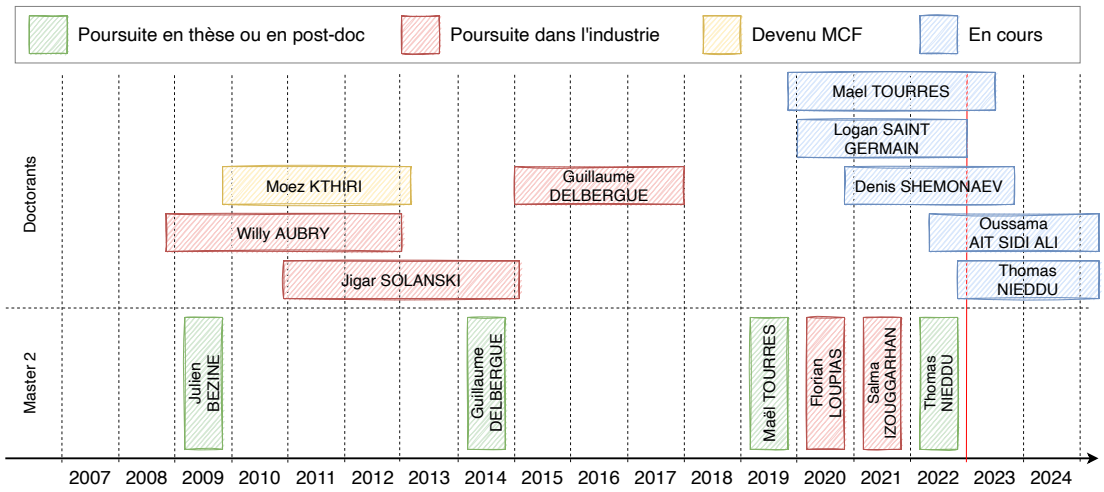


Figure 2.1 - Étudiants encadrés ayant participés aux études relatives à l'axe de recherche concernant la conception et l'utilisation d'architectures programmables.

2.2 Conception d'un système programmable pour l'analyse protocolaire

Deux ans après mon recrutement, en 2008, en parallèle des travaux visant à concevoir des architectures numériques sous contraintes, mes compétences en développement informatique couplé à mon appétence pour le domaine m'ont naturellement conduit à me rapprocher des chercheurs du laboratoire d'informatique (LaBRI) et plus particulièrement de Laurent REVEILLERE. Ce dernier concevait et déployait à partir de spécifications de haut niveau (langages DSL) des passerelles protocolaires sur des systèmes embarqués. Ces passerelles multi-utilisateurs et multi-protocoles avaient pour objectif d'analyser et de traduire les messages textuels échangés entre divers systèmes communicants dans un environnement ubiquitaire.

Le défi dans de tels systèmes consiste à traduire efficacement des messages textuels reçus dans un format $\mathbf{f}_1$ respectant le protocole $\mathbf{p}_1$ (ex : RTSP) sous la forme d'un message au format $\mathbf{f}_2$ respectant le protocole $\mathbf{p}_2$ (ex : SMTP). L'analyse et la décomposition des messages entrants ($\mathbf{f}_1$) consomment une part prépondérante du temps de traitement. En effet, cette étape d'analyse est implantée à l'aide d'automates qui se composent de plusieurs centaines d'états tel que cela est résumé dans le tableau 2.1 pour 4 protocoles usuels.

Dans ce contexte, j'ai proposé de concevoir un système dédié à la tâche d'analyse protocolaire, basé sur un cœur de processeur couplé à des accélérateurs matériels. Afin d'initier cette collaboration, j'ai candidaté et obtenu en 2009 un financement de projet BQR (Bonus Qualité Recherche, financement de l'ENSEIRB-MATMECA). Ce financement m'a permis d'amorcer les travaux en recrutant Julien BEZINE (Master 2) en 2010. Ce dernier a travaillé à la conception d'un prototype de système sur circuit FPGA couplant un processeur Xilinx Microblaze à un accélérateur décrit en VHDL. L'accélérateur était en charge de réaliser l'analyse protocolaire (implantation de l'automate) dans l'intention de décharger le processeur de cette tâche, mais aussi d'accélérer le temps de traitement des messages. Ce premier prototype déployé sur une carte Xilinx ML402 démontra l'intérêt de l'approche sur un exemple pédagogique, mais mit en exergue la nécessité de développer une méthodologie automatisée pour générer les accélérateurs matériels, les interconnecter automatiquement au cœur de processeur et générer le middleware simplifiant le déploiement d'applications réelles.

En parallèle de la demande de financement BQR, nous avions déposé un projet ANR (PRC) entre nos deux laboratoires dans le but de financer ces travaux de rechercher. Ce projet n'a cependant pas été retenu. Toutefois ce projet transdisciplinaire et trans-laboratoires a été soutenu par une

Table 2.1 - Complexité des automates (FSM) réalisant les opérations de vérification syntaxique et d'analyse sémantique des messages pour 4 protocoles réseau usuels.

Protocole	#états	Transitions non factorisées			Transitions factorisées		
		#trans	Moy. (trans/état)	Max. (trans/état)	#trans	Moy. (trans/état)	Max. (trans/état)
HTTP	66	3922	59.4	98	209	3.2	7
RTSP	56	2722	48.6	84	150	2.7	5
SIP	284	17451	61.4	119	1790	6.3	19
SMTP	333	6974	20.9	86	1083	3.3	7

```
Request = Request_Line
 (( general_header
  | request_header
  | entity_header ) CRLF)*
 CRLF
 message_body? {clen} ❸;
Request_Line   = Method SP Request_URI ^uri ❷ SP
 HTTP_Version CRLF;
Request_URI    = '*' | absoluteURI | abs_path | authority;
entity_header  = Allow
   | Content_Length
   | ...
Content_Length = 'Content-Length: ' digit+ ^clen as uint32 ❶;
```

Figure 2.2 - Exemple synthétique de spécification du protocole HTTP dans le langage Zebra.

bourse de thèse CNRS pour le recrutement de Jigar SOLANSKI en octobre 2010. J'ai co-encadré cette thèse avec Laurent REVEILLERE et David BROMBERG.

Les travaux ciblaient la mise au point d'une méthodologie de conception permettant de générer un système de type SoPC et son infrastructure logicielle à partir de descriptions protocolaires exprimées dans un DSL proche des notations BNF. Un exemple de spécification simplifiée du protocole HTTP est présenté dans la figure 2.2. Cette spécification est décrite dans le DSL nommé Zebra que nous avons développé pour l'étude. À partir d'une ou de plusieurs spécifications de protocoles applicatifs, la méthodologie devait permettre la génération du SoC et des codes logiciels associés. Comme cela est illustré dans la figure 2.3. La conception du système dans sa globalité impliquait la production de multiples éléments tant logiciels que matériels :

— Génération au niveau RTL des **P** accélérateurs matériels à intégrer dans le SoC pour accélérer l'analyse protocolaire ;
— Génération de l'architecture du SoC intégrant les **P** accélérateurs matériels ;
— Génération en langage C des drivers nécessaires à la configuration et à l'utilisation des **P** accélérateurs ;
— Génération en langage C du middleware fournissant les services de détection et d'utilisation

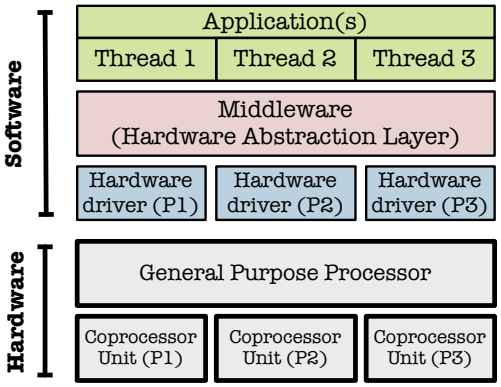


Figure 2.3 - Piles logicielles et matérielles mises en oeuve pour le déploiement d'applications d'analyse protocolaire au sein d'un SoC/SoPC.

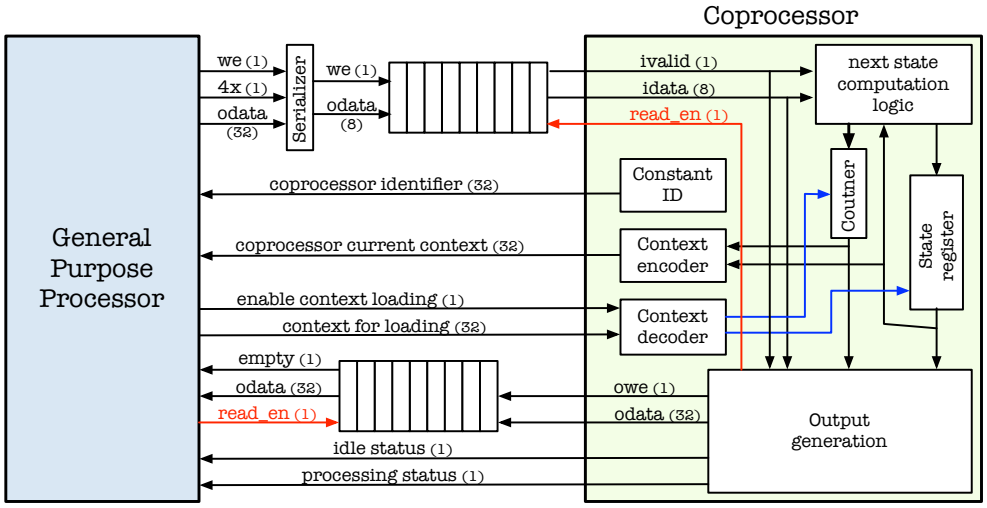


Figure 2.4 - Architecture interne d'un accélérateur matériel interfacé avec un processeur *soft-core*.

des accélérateurs matériels par les couches applicatives ;
— Génération des squelettes des couches applicatives en langage C. La complétion de ces dernières est à la charge de l'utilisateur.

Dans un premier temps, les travaux se sont sommes focalisés sur l'architecture interne des accélérateurs. Afin d'assurer les mêmes fonctionnalités que les automates implantés sous forme logicielle, il était nécessaire d'intégrer en plus des fonctions de traitement des données un support multi-utilisateurs et multi-protocoles. En conséquence, en supplément des E/S nécessaires à la consommation des messages et au transfert des résultats, un ensemble de signaux complémentaires a été ajouté comme illustré dans la figure 2.4. Par exemple des ports d'E/S sont dédiés au chargement et à la sauvegarde du contexte d'exécution de l'accélérateur. Cela permet l'entrelacement temporel de plusieurs tâches d'analyse en fonction de la disponibilité des données issues de l'interface réseau.

En interne, différentes architectures des automates ont été étudiées et deux approches ont été sélectionnées pour implanter les automates. La première approche se base sur une implantation traditionnelle d'une machine d'états de type Moore à base d'éléments combinatoires [Ped13]. Ce type d'architectures permet d'exploiter au mieux le parallélisme dans le calcul des conditions. En revanche, elle contraint l'architecture à ne supporter qu'un type d'analyse protocolaire (celui pour lequel l'accélérateur est conçu). La seconde se base sur une décomposition de la machine d'états sous forme de tables qui sont parcourues séquentiellement afin d'évaluer les transitions possibles et réaliser les changements d'états [Har81]. La seconde approche moins classique a pour principal avantage d'être reconfigurable par un simple changement du contenu des mémoires. Toutefois, le temps de calcul de l'état futur est plus lent que celui de l'approche combinatoire en raison d'un parcours dichotomique des données en mémoire. Pour pouvoir générer automatiquement ces différents automates à partir des spécifications exprimées dans notre DSL, nous avons modifié l'outil RAGEL [Thu06] développé à l'origine pour la génération d'automates en langage C. Nous avons ajouté des phases d'optimisation adaptées à la génération des automates matériels ainsi que deux *backends* pour assurer la génération des architectures au niveau RTL en langage VHDL.

Ce travail a été en partie supporté par Julien MERCADAL, d'un post-doctorant dont j'ai encadré le travail.

Pour concevoir un SoC intégrant ces accélérateurs, nous avons réalisé un état de l'art des processeurs *soft-cores* disponibles afin d'identifier le plus pertinant en termes de flexibilité, de périphériques disponibles et de performances (MIPS). Notre choix s'est porté sur le LEON-3 développé par Gaisler [Leon3, SPARCv8]. Ce choix a été motivé par :

1. ses performances élevées vis-à-vis des cœurs de type Microblaze ;
2. les périphériques inclus dans le SoPC (contrôleur mémoire, interface Ethernet), périphériques qui étaient déjà préconfigurés pour la carte de développement Xilinx ML605 ;
3. le fait que la description en VHDL était libre de droit nous autorisant ainsi à modifier l'architecture interne du processeur et du système.

L'architecture du LEON-3 a été amendée pour permettre un couplage fort entre les **P** accélérateurs matériels et le processeur tel que cela est schématisé dans la figure 2.4 en vue de maximiser les performances. Cette insertion des accélérateurs directement au sein du pipeline d'exécution du processeur permet de minimiser les pénalités temporelles liées à l'échange des données. Des FIFO de découplage ont été insérées sur les voies véhiculant des données autorisant ainsi une exécution localement synchrone et globalement asynchrone des éléments. Le pilotage des **P** accélérateurs par le processeur a nécessité une extension du jeu d'instructions du processeur LEON-3 qui est décrite dans [ri:LeGl6b]. Ces instructions spécifiques sont exploitées par les drivers logiciels exécutés par le cœur Léon. Ces derniers gèrent l'échange de données entre le CPU et les accélérateurs, mais aussi l'identification des protocoles supportés, l'analyse de l'état coprocesseur (ready/busy) et la sauvegarde/rechargement des contextes d'exécution. Au-dessus de cette couche logicielle permettant d'interagir directement avec les accélérateurs, un middleware fournissant les services nécessaires à la couche applicative a été mis au point. Cette couche a pour vocation d'abstraire la manière dont le service d'analyse protocolaire était exécuté, rendant transparent pour la couche applicative l'utilisation, le partage et la reconfiguration des accélérateurs matériels.

Dans le but d'évaluer les capacités de la plateforme dans de multiples scénarios, nous avons développé un flot complètement automatisé comme cela est décrit dans la figure 2.5. En entrée de la méthodologie, l'utilisateur fournit une description des protocoles à supporter exprimés dans le DSL Zebra ❷. À partir des descriptions des protocoles, les accélérateurs matériels sont générés ❹ dans leurs versions dédiées et configurables à l'aide de l'outil RAGEL amendé à cet effet. Les drivers logiciels des accélérateurs ainsi que le middleware ❺ sont aussi générés lors de cette étape. À partir d'un modèle générique et flexible de l'architecture SoPC ❶ et des propriétés des accélérateurs à intégrer, une architecture spécialisée du cœur de processeur LEON-3 et du système est produite ❸. L'architecture matérielle (❸ + ❹) est ensuite synthétisée à l'aide l'outil Xilinx Vivado et déployée sur carte Xilinx ML605 ❻. Les codes logiciels générés ❺, associés à l'application fournie par le concepteur, sont compilés à l'aide de GCC ❼.

Les performances des accélérateurs matériels ainsi que la pertinence de notre approche de conception ont été évaluées dans un premier temps. Nous avons comparé les niveaux de performance atteints à ceux d'architectures que nous aurions pu obtenir à l'aide d'outils de synthèse d'architectures en utilisant des descriptions C des analyseurs protocolaires [Thu06]. Différents outils de synthèse HLS ont été testés. La majorité d'entre eux (GAUT v2.4.3 [Cou08],

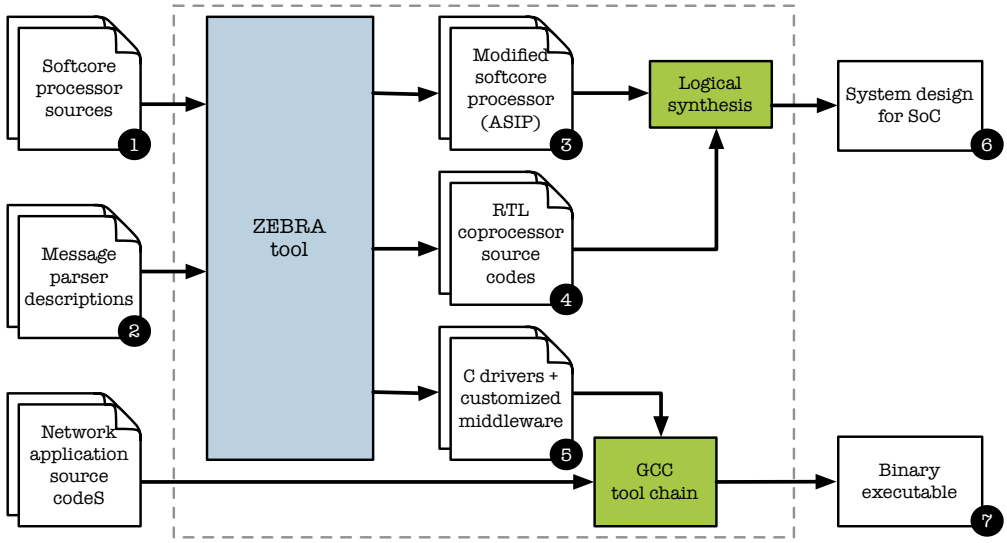


Figure 2.5 – Flot de conception global de la plateforme à l'aide de la méthodologie Zebra.

Vivado HLS 2014.2 [VivadoHLS], Excite 5.0c [eXC13] et C2Verilog 2013 [Ben10]) ne supportaient pas les sémantiques présentes dans la description des parsers logiciels (par exemple les GOTO) et/ou cessaient de fonctionner correctement durant l'étape de synthèse. Ainsi, nous n'avons pu comparer les performances de nos accélérateurs qu'à ceux produits par l'outil LegUp [Can12]. Les résultats expérimentaux ont montré que les outils de HLS, généralistes par nature, n'étaient pas efficaces pour la génération de tels accélérateurs. En effet, les accélérateurs générés par l'outil LegUP possédaient une complexité matérielle environ $10\times$ supérieure à celles obtenues par notre approche tandis que dans le même temps les performances temporelles étaient jusqu'à $40\times$ inférieures.

Dans un second temps, les gains ont été mesurés au niveau driver pour les 4 protocoles considérés. À ce niveau, l'utilisation des accélérateurs permet de réduire le temps d'exécution d'un facteur de $3,7\times$ à $11,2\times$ par rapport au traitement purement logiciel des données. Des mesures ont aussi été réalisées au niveau applicatif pour prendre en compte le temps d'exécution nécessaire pour remplir la structure logicielle avec les contenus identifiés lors du traitement d'analyse protocolaire. Le remplissage de la structure nécessite l'allocation dynamique de mémoire et la copie du contenu du flux analysé dans la structure logicielle. Dans ce contexte, les facteurs

Table 2.2 – Comparaison de la complexité matérielle des accélérateurs par rapport à celle du système complet sur ASIC et sur circuit FPGA.

	FPGA (@75MHz)			ASIC (@400MHz)
	LUT	REG	RAM18k	NAND gates
Accélérateur RTSP	812 (1%)	208 (< 1%)	0 (0%)	2661 (1.15%)
Accélérateur SMTP	2413 (1%)	217 (< 1%)	0 (0%)	5446 (3.04%)
Accélérateur SIP	4330 (3%)	213 (< 1%)	0 (0%)	8579 (4.80%)
Accélérateur HTTP	1178 (1%)	206 (< 1%)	0 (0%)	3995 (2.23%)
Système	34107 (23%)	15158 (5%)	42 (5%)	178781 (100%)

d'accélérations sont plus faibles. Ce résultat est dû au fait que les opérations d'allocation de mémoire (malloc) et de copie de mémoire (memcpy) consomment un temps non négligeable indépendamment de l'implémentation de l'analyseur protocolaire. Cependant, l'accélération reste significative : les temps d'exécution sont réduits de facteurs $3,1\times$ à $4,5\times$. La mise en perspective des gains obtenus vis-à-vis de l'augmentation de la complexité matérielle de la solution, données synthétisées dans le tableau 2.2, conforte la pertinence de la solution développée.

Les travaux scientifiques développés lors de cette étude ainsi que les résultats expérimentaux ont été valorisés par la publication de 2 articles dans des journaux internationaux (IEEE ESL [ri:Mer12] et ACM TRETS [ri:LeG16b]), d'un article dans une conférence internationale [ci:Sol13] et ont donné lieu à une présentation invitée lors d'une journée du GdR SoC-SiP [pn:LeG15].

2.3 Architectures programmables et flexibles pour le décodage des codes LDPC

Ce thème de recherche a débuté en 2011 suite au recrutement de Christophe JEGO. Mon objectif était d'appliquer mes activités de recherche antérieures au domaine des systèmes de communications numériques et plus particulièrement celui des ECC pour développer une synergie. Pour cela nous avons tiré parti de mes compétences liées aux méthodologies de conception couplé à l'utilisation d'architectures flexibles et programmables et de celles de Christophe JEGO concernant le domaine de la conception d'architectures matérielles de décodeurs ECC. Cette collaboration, financée sur fonds propres, s'est focalisée dans un premier temps sur l'implantation de décodeurs pour les codes LDPC.

Lors du démarrage de cette action, les travaux connexes concernant la conception de décodeurs LDPC matériels se répartissaient principalement selon deux catégories :

1. La première traitait de la conception d'architectures matérielles de décodeurs LDPC sur des circuits ASIC et/ou FPGA. L'objectif de ces travaux était principalement la minimisation de la complexité matérielle et/ou la maximisation du débit des architectures [Rov07, Gui07a, Mar11a, Hai16, Mau16]. Les hauts niveaux d'efficacité obtenus pour les codes LDPC Quasi-Cyclic (QC-LDPC) étaient cependant caractérisés par de longs cycles de conception et de faibles niveaux de flexibilité.
2. La seconde, afin de faciliter la réutilisation des architectures matérielles, se focalisait sur la conception de décodeurs plus flexibles. Pour atteindre ce but, il est possible de concevoir des cœurs de processeurs spécialisés de type ASIP programmables en assembleur [Nae08, Zha09, Mur11]. Toutefois la conception d'ASIP et de leurs environnements de développement est une tâche complexe et chronophage.

À partir des compétences développées dans le cadre de la conception d'architectures de cœurs de processeurs enrichis couplées avec celles issues des travaux sur les méthodologies de conception automatisées (Chapitre 1), nous avons œuvré au développement d'une approche alternative. L'objectif était de concevoir une architecture générique et flexible avec une méthodologie de conception automatisée. Cette méthodologie devait permettre la configuration et la (re)programmation automatique de l'architecture à partir des matrices $\mathbf{H}$ spécifiant les codes LDPC à décoder. Contrairement aux travaux antérieurs [All08, Nae08, Gen10], j'ai décidé de baser

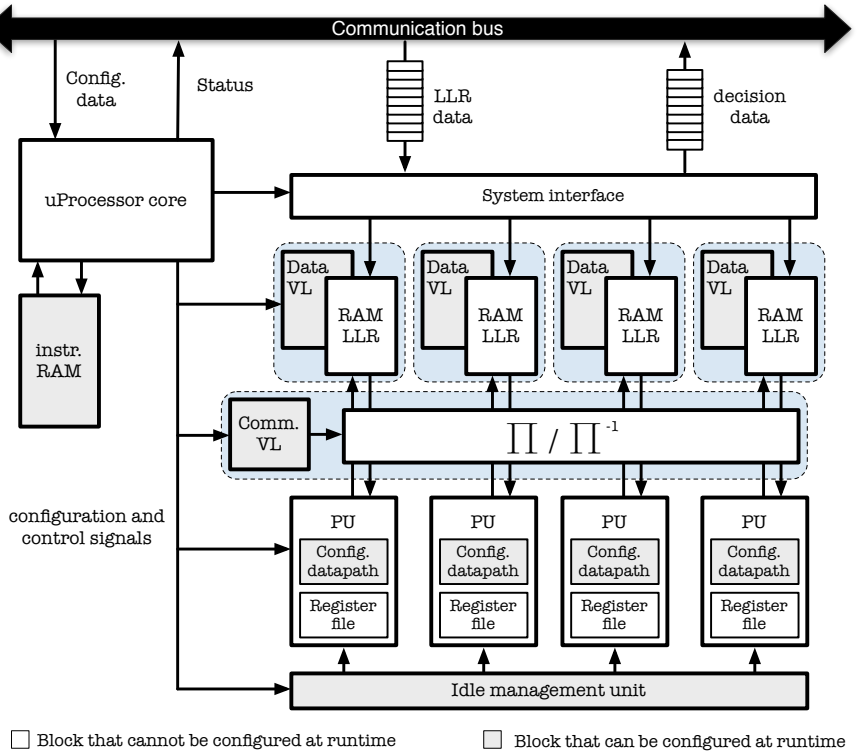


Figure 2.6 - Architecture matérielle générique proposée pour le décodage des codes LDPC.

ces travaux sur la réutilisation d'un cœur de processeur libre de droit et de son écosystème puis de l'enrichir à l'aide d'une matrice d'éléments de calculs adaptés au décodage des codes LDPC. Ce type d'architecture était inspiré par les architectures des processeurs many-cœurs de type GPU. L'architecture générique développée dans le cadre de ces travaux est présentée dans la figure 2.6. Cette architecture de type semi-parallèle qui permet le traitement des équations de parité par couches horizontales [Hoc04] se compose principalement :

— d'un cœur de processeur programmable de type MIPS nommé Plasma [Plasma],
— d'une matrice composée de $\mathbf{P}$ éléments de calcul,
— de $\mathbf{M}$ éléments de mémorisation avec ici $\mathbf{M} = \mathbf{P}$.

Deux réseaux de Benes $\mathbf{P} \times \mathbf{M}$ [Ben65] situés entre les $\mathbf{P}$ éléments de calcul aux $\mathbf{M}$ bancs mémoire assurent le routage correct des données à l'exécution. Le contrôle de ces structures d'entrelacement est généré à partir des matrices $\mathbf{H}$ à supporter ainsi que de l'ordonnancement des calculs. Pour assurer l'adaptabilité de l'architecture à tous types de codes LDPC, des couches de virtualisation des accès mémoires (VL) sont présentes dans l'architecture. Ces couches sont en charge d'alimenter à chaque cycle d'horloge les $\mathbf{P}$ éléments de calcul (PU) à partir des données contenues dans les $\mathbf{M}$ bancs mémoire et de mémoriser les résultats des calculs. Leurs contenus sont générés *off-line* lors de l'ordonnancement des calculs.

L'architecture interne des PU, détaillée dans [ri:LeGl4a], est similaire à celles proposées dans la littérature pour les architectures de décodeurs dédiées [Mar11a, Mar15]. Ces PU fonctionnent

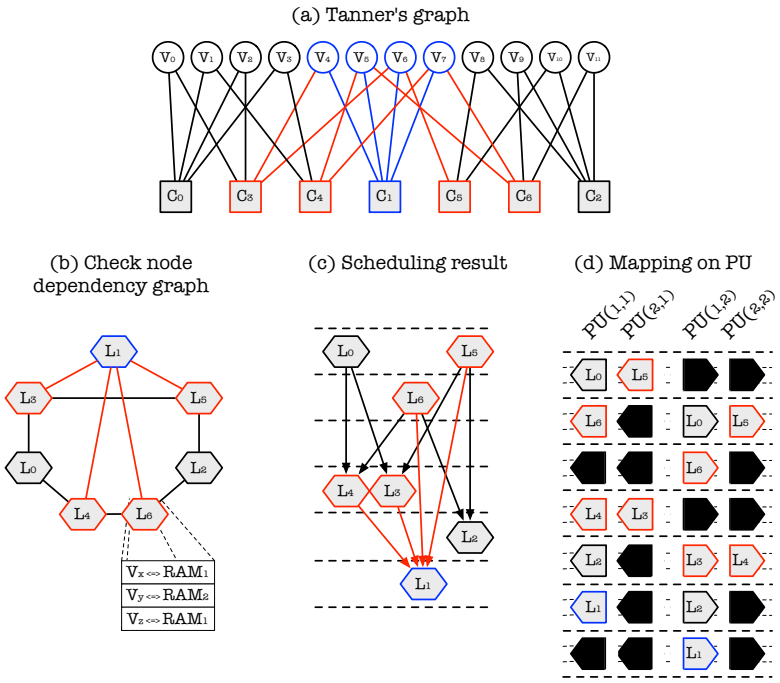


Figure 2.7 – Transformation de la matrice de parité sous forme de graphe de compatibilité afin d'ordonnancer et d'assigner le calcul des noeuds de parité.

de manière pipeline (2 tranches) afin de maximiser le débit de l'architecture. Toutefois, afin d'augmenter la flexibilité de la solution architecturale et pour répondre au mieux aux besoins applicatifs, nous avons choisi de (1) rendre le format de quantification interne configurable et (2) permettre la sélection de l'approximation Min-Sum [Che02b] sélectionnable à l'exécution.

La présence d'un cœur de processeur, programmable en langage C, au sein de l'architecture, rend la (re)configuration de l'architecture à la volée possible (couches d'abstraction, séquencement des calculs). L'architecture matricielle ($\mathbf{P}$, $\mathbf{M}$) couplée aux réseaux de Benes permet l'utilisation de la solution proposée pour les codes LDPC Quasi-Cyclic [Myu05], mais aussi pour des codes LDPC non structurés [Tan81]. Toutefois, même si cette architecture permet théoriquement d'obtenir un débit évoluant linéairement en fonction de $\mathbf{P}$, il est très difficile en pratique d'atteindre de tels niveaux en raison des dépendances de données et des conflits d'accès aux mémoires [Ama09, Mar09]. En effet, il est nécessaire d'accéder à $\mathbf{P}$ données en lecture et en écriture dans les $\mathbf{M}$ bancs mémoire chaque cycle d'horloge. Ce problème prépondérant pour les codes LDPC non structurés est aussi présent avec les codes QC-LDPC lorsque la valeur de $\mathbf{P}$ est différente de celle de $\mathbf{Z}$ [Marlla]. Pour résoudre ce problème d'optimisation, nous avons mis en œuvre une approche permettant d'ordonnancer automatiquement les calculs (CN) et d'assigner ces calculs sur les PU en prenant en compte les contraintes architecturales ($\mathbf{P}$, $\mathbf{M}$), l'assignation des données dans les bancs mémoire et les dépendances de donnée entre les calculs.

La matrice $\mathbf{H}$ décrivant les calculs de parité à réaliser (figure 2.7a) ainsi que les dépendances de données entre les calculs sont modélisées à l'aide d'un graphe de compatibilité (figure 2.7b). Ce modèle représente les calculs de CN qui ne sont pas interdépendants au niveau macroscopique : chaque nœud du modèle représente les d_c calculs et les d_c accès mémoires nécessaires à sa

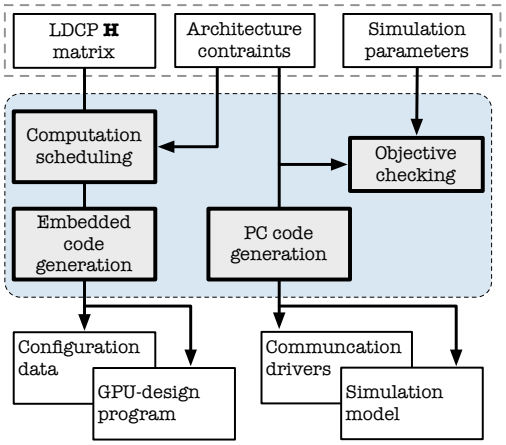


Figure 2.8 – Méthodologie de conception proposée pour configurer l'architecture matérielle et générer le programme logiciel assurant le pilotage à partir de la matrice spécifiant le code LDPC à supporter et des contraintes de l'utilisateur.

complétion. Les étapes d'ordonnancement et d'assignation exploitent ce modèle afin d'identifier les groupes de nœuds compatibles par macro-unité de temps (figure 2.7c). L'identification et la gestion des conflits d'accès aux bancs mémoire sont réalisées à ce moment-là. Si les conflits d'accès peuvent être résolus par une permutation temporelle de l'accès aux données alors l'opération est ordonnée. Sinon elle est reportée au macro-cycle suivant avec une insertion le cas échant de *slots* temporels supplémentaires (figure 2.7d). Les algorithmes utilisés ici sont dérivés de ceux utilisés dans les outils de synthèse HLS (Chapitre 1).

En vue de réduire le temps de conception des architectures de décodeur et/ou simplifier la réutilisation des architectures matérielles déjà déployées, nous avons inclus cette méthodologie dans un flot de conception plus global. Le flot de conception proposé est résumé dans la figure 2.8. L'ensemble de ces phases de génération et/ou de configuration se basent à la fois sur la ou les matrices $\mathbf{H}$ que doit supporter le décodeur matériel, mais aussi sur les contraintes de performances imposées par l'utilisateur ($\mathbf{M}$ et $\mathbf{P}$). À partir de ces informations, le flot ordonnance et assigne les calculs. Puis, à partir des résultats obtenus génère le code logiciel C pour le cœur MIPS. Le code logiciel permet de contrôler les éléments de calcul lors du processus de décodage, mais aussi de charger les mémoires RAM qui :

— virtualisent l'accès à l'information durant le décodage ;
— contrôlent les réseaux de Benes via des mots de commandes précalculés.

L'architecture de décodeur développée et la méthodologie associée ont été testées et validées par prototypage sur le circuit FPGA (Xilinx ML605). Des simulations conjointes avec un ordinateur hôte en charge de la génération des trames et de la simulation d'un canal AWGN ont été réalisées grâce à l'utilisation d'un lien PCIe. Les multiples courbes de BER/FER tracées ont permis de valider l'architecture et la méthodologie. La solution déployée sur un circuit FPGA de complexité modeste offre un degré de flexibilité équivalent aux décodeurs LDPC implantés sur des cibles GPU [Wanlla, Wanllc, Fallla, Falllb] grâce au flot de (re)programmation. Toutefois notre solution se démarque par des gains de l'ordre $\approx 3\times$ et $\approx 100\times$ en termes de débit et de consommation énergétique face aux implantations sur GPU tandis que le prix du circuit FPGA était $\approx 5\times$ moins

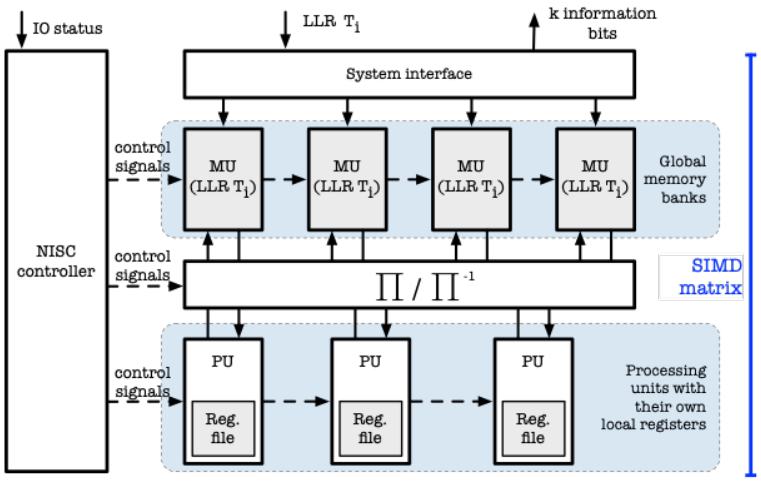


Figure 2.9 - Architecture matérielle du décodeur matériel de type NISC avec une asymétrie entre le nombre d'éléments de calcul et d'unités de mémorisation.

onéreux. Ces travaux de recherche et les résultats associés ont donné lieu à deux publications dans des conférences internationales [ci:LeGl2c, ci:LeGl2c] et à un article dans la revue ACM TRETS [ri:LeGl4a].

2.4 Extension de l'approche aux architectures dédiées

L'architecture développée avait pour vocation première d'être modulaire et (re)configurable. Par rapport aux architectures spécialisées pour le décodage des codes QC-LDPC, la solution proposée s'avère comme cela était prévisible plus complexe matériellement lorsque ($\mathbf{P} = \mathbf{Z}$) en raison des ressources assurant la flexibilité de l'architecture. Cependant l'approche proposée permet à l'instar des architectures dédiées de concevoir des architectures lorsque $\mathbf{P}$ souhaité est différent de la valeur de $\mathbf{Z}$ afin de répondre plus efficacement aux besoins applicatifs. En effet, de nombreux systèmes requièrent des architectures matérielles de faibles complexités ciblant des débits modérés pour lesquels la valeur optimale du paramètre $\mathbf{P}$ est inférieure à la valeur de $\mathbf{Z}$. Afin d'améliorer la pertinence des architectures générées dans ce contexte, et améliorer la compétitivité de notre approche face aux architectures dédiées, nous avons fait évoluer notre méthodologie de conception. La suppression de la capacité de reprogrammation post-synthèse de l'architecture a engendré une réduction du coût silicium des architectures générées :

— Dans un premier temps, afin de minimiser la complexité matérielle du contrôleur de l'architecture, le cœur de processeur MIPS a été transformé en un contrôleur de type NISC [Res05]. Ce type de contrôleur toujours programmable délivre les signaux de contrôle stockés dans une mémoire. La présence de motifs récurrents dans les signaux de contrôle a permis la mise en œuvre d'un mécanisme de compression permettant de minimiser la complexité mémoire. Le remplacement du cœur MIPS par un contrôleur de type NISC a aussi permis d'augmenter la fréquence de fonctionnement de l'architecture. En effet, le chemin critique était précédemment localisé dans le décodeur d'instruction du cœur MIPS.
— Dans un second temps, afin d'atteindre les compromis architecturaux les plus pertinents

pour un contexte applicatif donné, nous avons transformé la matrice de PU / bancs mémoire de manière à pouvoir la rendre asymétrique. Le nombre de bancs mémoire **M** peut ainsi être supérieur au nombre d'unités de calcul **P** comme cela est schématisé dans la figure 2.9. Cette asymétrie permet de résoudre simplement des problèmes de contention et/ou conflits d'accès à la mémoire lors de la phase de compilation de la matrice **H**.

— Dans un troisième temps, les éléments de mémorisation de type RAM en charge de l'abstraction des accès aux données permettant la reconfiguration à la volée du décodeur ont été transformés en mémoires de type ROM. Cette transformation de la nature des mémoires fournit des gains substantiels en termes de coût silicium sur les technologies ASIC.

Enfin la méthodologie a été enrichie de fonctionnalités permettant d'explorer automatiquement l'espace des solutions architecturales. Pour ce faire, à partir des contraintes utilisateur (**H**, débit ou complexité), le flot peut générer en quelques minutes un ensemble d'architectures pour différentes valeurs de **P** et **M** en utilisant différentes heuristiques d'ordonnancement de manière à identifier la ou les meilleures solutions architecturales selon le contexte applicatif. L'architecture matérielle flexible ainsi obtenue couplée à l'évolution de la méthodologie a permis de rivaliser favorablement avec les travaux issus de l'état de l'art comme cela est détaillé dans l'article publié dans IEEE TSP [ri:LeGl4b].

L'ensemble de ces travaux prospectifs réalisés entre 2010 et 2014 m'ont permis de développer une expertise dans le domaine de la conception de décodeurs matériels pour les codes LDPC. Les différents prototypes matériels déployés et validés sur des cartes de développement FPGA ont démontré une compétence et un savoir-faire qui ont permis par la suite l'amorçage de collaborations industrielles avec Thales Aliena Space et Airbus S&D comme cela est détaillé dans le troisième chapitre de ce manuscrit.

2.5 Implémentation de décodeurs ECC sur architectures HPC

A partir des années 2010, la mise sur le marché d'architectures many-cœurs et l'augmentation des performances des architectures multi-cœurs ont levé une partie des verrous limitant l'usage de ces solutions dans le développement logiciel de systèmes de communications numériques en temps réel dans un contexte radio-logicielle (SDR). De nombreux chercheurs se sont intéressés à l'implantation des fonctions nécessaires à ces systèmes de communication sur des cibles multi-cœurs et many-cœurs. Un nombre important de travaux se sont naturellement portés sur le décodage de codes LDPC qui était un des éléments limitant le débit. De plus, le décodage de codes LDPC est, *a priori*, facilement parallélisable. Ces décodeurs peuvent donc bénéficier des capacités SIMD, MIMD ou SIMT des architectures multi-core [Fal08] et many-core [Fall1b, Fall1a].

Lors de l'étude précédente, des modèles logiciels de décodeurs LDPC avaient été développés dans le but de minimiser le temps de prototypage et valider leur capacité de correction d'erreurs. Ces modèles servaient initialement à discriminer rapidement les choix de conception et à faciliter le débogage des architectures au niveau RTL. Pour accélérer l'estimation des performances des décodeurs, j'ai employé des stratégies de parallélisation usuelles dans le domaine de la programmation de systèmes HPC. Les niveaux de performance atteints par les modèles logiciels sur cible multi-cœurs se sont révélés plus efficaces que ceux rapportés dans la littérature pour des architectures many-cœurs [Fall1b, Fall1a]. Les niveaux de performance élevés en termes de débit que j'obtenais sur un ordinateur portable associés à mon appétence pour le domaine de la

parallélisation et l'optimisation sur les cibles de type HPC m'ont motivé pour initier des travaux de recherche dans ce domaine. Mon objectif initial était d'évaluer la pertinence des d'architectures multi-cœurs pour des systèmes SDR. Ces travaux débutés en 2013 et financés dans un premier temps sur fonds propres et mon implication personnelle ont permis de démontrer qu'il est possible d'implanter la fonction de décodage de la couche basse (couche physique du modèle OSI) des systèmes SDR ou Cloud-RAN (C-RAN) [Wul5, Siml6] sous contrainte temps réels.

Les architectures many-cœurs employées dans la littérature pour le décodage LDPC offraient des capacités de parallélisation très élevées (plusieurs centaines ou milliers de cœurs CUDA). Les algorithmes de décodage utilisés pour le décodage des codes LDPC étant composés de nids de boucles appliquant un traitement parallèle sur un grand nombre de données. C'est pourquoi, nos travaux se sont naturellement focalisés sur l'utilisation de cibles GPU. Les travaux rapportés à l'époque [Chal0a, Wanlla, Fallla, Grol2a, Kanl2a] se basaient sur le postulat erroné que l'ordonnancement par inondation offrant un parallélisme de calculs massif aboutirait à des implantations avec les meilleurs débits. Cependant, plusieurs verrous architecturaux limitaient les performances des solutions développées :

— la nécessité de réaliser des accès irréguliers aux données ;
— l'utilisation de données flottantes augmentant les besoins en bande passante mémoire ;
— la faible vitesse de convergence de l'algorithme de décodage.

Forts de l'expérience acquise dans la conception de décodeurs LDPC matériels, afin de lever ces verrous, nous avons fait le choix de nous focaliser sur la formulation de l'algorithme de décodage employant un ordonnancement par couches horizontales [Hoc04]. Cette formulation de l'algorithme de décodage accélérait environ 2× la vitesse de convergence du décodeur tout en réduisant les besoins en bande passante mémoire. Toutefois, cette formulation engendre aussi l'apparition de dépendances de données dans les cœurs de boucle réduisant drastiquement les possibilités de parallélisation comme cela est détaillé dans [ri:LeGl4c, ri:LeGl6a].

Pour permettre une exploitation optimale de l'architecture du GPU en rendant régulier les accès à la mémoire lors du processus de décodage et ainsi utiliser l'ensemble des cœurs CUDA disponibles, la stratégie de parallélisation que nous avons mise en œuvre visait à décoder $\mathbf{F}$ trames en parallèle avec $\mathbf{F}$ le nombre de cœurs CUDA disponibles. Cette stratégie de parallélisation consistant à décoder de multiples trames en parallèle permet d'obtenir une régularité au niveau des calculs et des accès mémoire sur l'ensemble de l'architecture. En effet, tous les cœurs réalisent les mêmes calculs sur des données indépendantes. Toutefois, l'utilisation de cette stratégie de parallélisation nécessite un entrelacement des données des différentes trames en amont et en aval de l'étape de décodage pour permettre des accès réguliers à la mémoire. L'implantation de ces deux transformations dont le principe est schématisé dans la figure 2.10 correspond à une transposition matricielle dont le temps d'exécution s'avère marginal vis-à-vis du temps de décodage de l'information.

Pour maximiser le débit, nous avons associé à cette parallélisation massive, l'utilisation des capacités SIMD des cœurs CUDA. L'utilisation des fonctionnalités SIMD permet le traitement de 4 données par cycle d'horloge par cœur CUDA lorsque les données manipulées sont exprimées sur 8 bits ($4 \times 8\text{b} = 32\text{b}$). La maitrise des formats de quantification en virgule fixe issus du développement de décodeurs matériels nous a permis de réaliser une implantation efficace de l'algorithme et donc de confier à chaque cœur CUDA le traitement en parallèle de 4 trames

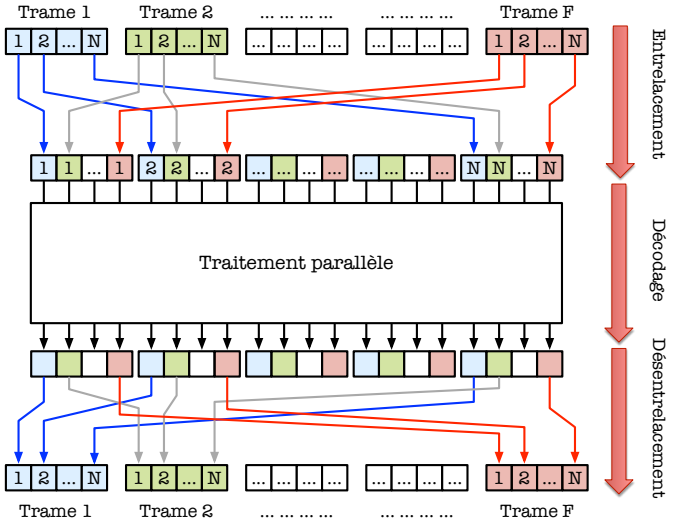


Figure 2.10 - Stratégie d'entrelacement et de desentrelacement des données provenant de différentes trames afin de permettre un accès régulier aux données lors du processus de décodage.

LDPC. Dans cette configuration, le GPU décode au total $\mathbf{P} = 4 \times \mathbf{F}$ trames par exécution.

Des expérimentations ont été menées sur un GPU NVIDIA GTX660 de milieu de gamme intégrant 960 cœurs CUDA. Les résultats obtenus, publiés dans un article IEEE ESL [ri:LeGl4c], ont démontré la pertinence de l'approche. Les débits de décodage obtenus sont d'un facteur 2× à 18× supérieurs à ceux de l'état de l'art alors que ces derniers manipulent des GPUs haut de gamme [Fall1a, Wan11c, Fall2a, Wan13] jusqu'à 15× plus onéreux avec jusqu'à 3× plus d'unités de calcul.

Ces travaux sur les architectures de type GPU ont mis en exergue que :

— la parallélisation inter-trames massive induisait des latences supérieures à une seconde. Cette latence très élevée limitait donc l'utilisation de ces décodeurs à la simulation de systèmes de communication.
— les architectures GPU qui offrent d'énormes capacités calculatoires sont massivement inefficaces en termes de bits/cœur/hertz et d'énergie/bit décodé surtout si l'on prend en compte les temps de transfert entre le GPU et le CPU hôte.

A la suite de ce travail, nous avons choisi de recentrer ces travaux de recherche autour des architectures multi-cœurs de type INTEL ou ARM. Ces dernières offrent des capacités de parallélisation moins élevées, mais possèdent à la fois une plus grande flexibilité en termes de stratégie de parallélisation (SIMD, MIMD, SPMD) et une plus grande efficacité pour le traitement des données entières. Ces cibles technologiques ont une puissance calculatoire importante si l'on arrive à exploiter conjointement leurs capacités superscalaire, SIMD et SPMD. Toutefois pour bénéficier de ces caractéristiques, il est indispensable que les descriptions logicielles soient méticuleusement optimisées afin d'exploiter au mieux les capacités des architectures matérielles. Dans le but de maximiser les performances en termes de débit et minimiser la latence de décodage nous avons fait le choix de conserver la formulation algorithmique du

Table 2.3 - Performances théoriques de l'implantation logicielle des boucles exécutant le calcul des nœuds de parité (CN) en fonction du degré des nœuds.

Degré du nœud	Nombre d'instructions	Temps d'exécution	ILP moyen
6	169 instr.	47 cycles	≈ 3.6
8	218 instr.	58 cycles	≈ 3.8
10	269 instr.	71 cycles	≈ 3.8
12	320 instr.	84 cycles	≈ 3.8
20	548 instr.	139 cycles	≈ 3.9
32	858 instr.	217 cycles	≈ 4.0

processus de décodage par couches horizontales [Hoc04]. Toutefois, la parallélisation offerte par les architectures multi-cœurs étant moins importante que pour les GPU, et les jeux d'instructions SIMD spécifiques, un travail conséquent de réécriture et d'optimisation des codes logiciels a été nécessaire.

Dans un premier temps, les capacités SIMD (SSE4 et AVX2) ont été mises à profit en vue de paralléliser le décodage de $\mathbf{F} = \{16, 32\}$ trames en parallèle par cœur. Cette stratégie de parallélisation permet de maximiser l'utilisation des unités SIMD. En effet, les données des différentes trames sont indépendantes, permettant ainsi d'obtenir un taux d'utilisation de 100% sur l'ensemble du processus de décodage. L'utilisation de transformations mathématiques a permis dans un second temps d'améliorer l'adéquation entre l'algorithme et les capacités architecturales en exploitant au mieux les instructions SIMD disponibles. Enfin, afin d'augmenter l'ILP de l'implantation logicielle, nous avons tiré parti d'approches génératives et de l'utilisation de *templates* pour supprimer des structures de contrôle et les structures itératives du code exécuté. L'ensemble de ces travaux d'optimisation a permis d'obtenir une implantation compacte des cœurs de boucle. Ces dernière nécessitent peu de cycles d'horloge comme cela est mis en évidence dans le tableau 2.3. Le nombre de cycles d'horloge correspond au temps nécessaire pour exécuter $\mathbf{F}$ nœuds CN. Il est à noter que l'ILP moyen de ces cœurs de boucle, sans prise en compte des temps d'accès à la mémoire, avoisine 4 mettant en évidence l'exploitation efficace de l'architecture superscalaire.

L'ensemble de ces travaux d'optimisation a permis d'obtenir des niveaux de performance plus élevés que ceux rapportés dans la littérature avec des architectures de type GPU à l'aide d'un simple ordinateur portable (INTEL Haswell Core-i7 4960HQ) comme cela est détaillé dans le tableau 2.4. Les débits obtenus sur une architecture multi-cœurs comprenant 4 cœurs physiques ($\mathbf{P} = 4 \times \mathbf{F}$) délivre des débits supérieurs à l'ensemble des travaux de la littérature sur cible GPU avec des facteurs d'accélération importants par comparaison aux travaux de l'état de l'art ($2\times$ à $6\times$). Il est à noter que la solution proposée est même $1.4\times$ fois plus efficace que celle mettant en œuvre quatre GPU NIVIDIA GTX TITAN. En dehors de l'augmentation notable du débit, cette solution multi-cœurs réduit aussi drastiquement la latence de décodage ($\leq 100us$) ainsi que la consommation d'énergie.

Ces travaux traitants d'implantation des codes LDPC sur des architectures multi-cœurs ont été valorisés au travers de 2 publications dans des revues internationales. La première dans IEEE TPDS [ri:LeGl6a] se focalise sur les architectures multi-cœurs INTEL tandis que la seconde explore les capacités des processeurs ARM [ri:LeGl5c].

Table 2.4 - Comparaison des débits atteints par les décodeurs LDPC proposés sur cible multi-cœurs (x86) avec ceux des travaux de la littérature déployés sur des architectures many-cœurs.

Code	Travaux de la littérature			Performances des décodeurs logiciels				
	Référence	Plateforme	Débit (Mbps)	1 core SSE (Mbps)	4 cores SSE (Mbps)	1 core AVX (Mbps)	4 cores AVX (Mbps)	Gain
(4000, 2000)	[Fal09a]	Tesla C1060	0.05	76	281	123	427	×8540
(4000, 2000)	[Cha10a]	Tesla C1060	2.34	354	1343	513	1828	×781
(64800, 32400)	[Fal11a]	Fermi C2050	76	91	299	120	347	×4.6
(4896, 2448)	[Fal11b]	8800 GTX	8	78	286	122	418	×54
(2304, 1152)	[Jil11a]	GTX 285	0.85	38	130	63	217	×255
(2304, 1152)	[Wan11c]	GTX 470	52	347	1288	508	2005	×38
(2304, 1152)	[Wan11a]	GTX 470	49	347	1288	508	2005	×41
(8000, 4000)	[Fal12a]	ATI HD5870	209	353	1315	468	1710	×8.2
(2048, 384)	[Kan12a]	GTX 480	25	184	717	226	887	×36
(16200, 7560)	[Gro12a]	GTX-570	186	121	402	172	496	×2.6
(2304, 1152)	[Wan13]	GTX TITAN	316	347	1288	508	2005	×6.3
(2304, 1152)	[Wan13]	4× GTX TITAN	1250	347	1288	508	2005	×1.6
(2304, 1152)	[Lil3a]	GTX 580	710	347	1288	508	2005	×2.8
(2304, 1152)	[Lil3a]	Fermi C2050	618	347	1288	508	2005	×3.2
(4000, 2000)	[Lin14]	GTX 660 Ti	107	76	281	123	427	×3.9
(64800, 32400)	[ri:LeGl4c]	GTX 660	664	227	864	311	931	×1.4

Les codes logiciels développés dans le cadre de ces études ont été rendus publics sous licence open-source afin d'attester l'authenticité des résultats que nous exposions. En effet, ces derniers étaient à l'opposé des résultats rapportés dans l'état de l'art. Les publications et la mise à disposition des codes logiciels ont engendré des retombées inattendues. Tout d'abord nous avons été sollicités par l'entreprise Prisma Telecom, rachetée depuis par KeySight afin d'obtenir une licence d'exploitation des codes logiciels issus de l'étude. L'objectif était de les intégrer dans des stations de base pour le standard 5G. En parallèle, ces codes logiciels ont été employés dans le développement de prototypes de stations de base pour des communications satellitaires par l'entreprise Aerospace Corp [Gra19]. De plus, nous avons été contactés par INTEL qui, intéressé par nos travaux, a réalisé une donation en 2019 d'un serveur de calcul ($\approx$ 40 K€) incluant deux Xeon Gold 6148 (2×20 cœurs physiques, 256 Go de RAM, AVX-512) sans contrepartie afin d'aider à la poursuite de cet axe de recherche.

La contractualisation avec Prisma Telecom associée avec la standardisation de la 5G nous ont amenés à poursuivre ces travaux. En effet, même si les débits obtenus étaient supérieurs aux besoins applicatifs, la stratégie de parallélisation inter-trames impactait la latence qui s'avère être un élément clef du standard 5G.

Nous avons ainsi mis au point une stratégie de parallélisation se focalisant sur le parallélisme de calcul situé dans l'algorithme de décodage (intra-trame). Cette approche mime le comportement des architectures matérielles de décodeurs LDPC [Mar11a] malgré le fait que pour les architectures multi-cœurs $\mathbf{F} \leq \mathbf{Z}$. L'étude s'est focalisée sur les codes QC-LDPC utilisés dans les standards de communication actuels (Wifi, WiMax, DBV-S2x, 5G). Pour ces codes LDPC, nous avons proposé une approche permettant de réaliser des accès irréguliers à la mémoire à moindre coût, et ce indépendamment du jeu d'instructions SIMD (SSE4, AVX2, AVX512, NEON). Toutefois, les accès à la mémoire sont moins efficaces que ceux réalisés avec l'approche inter-trames et le parallélisme de calcul varie en fonction des codes LDPC traités. En effet, ce dernier évolue en fonction de $\mathbf{Z}$ et de la configuration SIMD sélectionnée ($\mathbf{F}$). Ces inconvénients sont toutefois en partie compensés

Table 2.5 - Débits et latences des décodeurs LDPC logiciels en fonction de la stratégie de parallélisation sur un processeur faible consommation (INTEL Core-i7 5650U) pour 10 itérations de décodage.

Code		Inter-frame		Intra-frame	
N	**R**	Débit	Latence	Débit	Latence
576	1/2	262 Mbps	70 μs	102 Mbps	5.6 μs
2304	1/2	245 Mbps	300 μs	202 Mbps	8.6 μs
672	1/2	247 Mbps	155 μs	153 Mbps	7.8 μs
672	5/8	221 Mbps	173 μs	167 Mbps	7.2 μs
672	3/4	230 Mbps	166 μs	154 Mbps	7.7 μs
672	13/16	238 Mbps	161 μs	155 Mbps	7.7 μs

par la sobriété mémoire de cette stratégie de parallélisation. Une seule et unique trame est manipulée lors du décodage, réduisant *de facto* l'empreinte mémoire d'un facteur 16, 32 ou 64 et donc le nombre de défauts de cache de niveaux L1 et L2. Les résultats présentés dans le tableau 2.5, issus d'une publication à la conférence SIPS 2017 [ci:LeGl7a], illustrent les niveaux de performance atteints. On observe que pour le code LDPC court ($\mathbf{N} = 576$), la nouvelle stratégie de parallélisation intra-trame offre des débits $2\times$ inférieurs aux décodeurs inter-trames, mais elle permet une réduction de la latence de facteurs $12\times$ à $20\times$ lorsque $\mathbf{P} = 1$ et $\mathbf{F} = 32$. De plus, pour les codes LDPC avec $\mathbf{N} < 576$, la réduction du débit varie de $10\% \approx 35\%$ tandis que la latence est réduite jusqu'à $34\times$.

Cette étude s'est poursuivie afin d'améliorer la flexibilité de l'approche proposée dans [ci:LeGl7a] qui exploitait la génération de codes logiciels couplée à une utilisation intensive de templates. Ces travaux se sont déroulés dans le cadre de collaborations industrielles ciblant les standards CCSDS et DVB-S2 [DVB-S2] avec les entreprises SAFRAN, KRATOS et VASTECH Technology. L'étude a permis de lever les limitations de travaux antérieurs et d'aboutir à un modèle de décodeur flexible, performant et scalable comme cela est mis en évidence dans la figure 2.11. Les faibles latences de décodage ($\leq 100\mu s$) des décodeurs intra-trame font de cette solution, la seule compatible avec les contraintes du standard 5G.

En plus de ces études ciblant les architectures multi-cœurs développées par INTEL/ARM, nous avons récemment mené deux études pour l'entreprise Kalray (2021 & 2022) afin d'adapter les décodeurs logiciels développés à l'origine pour des cibles INTEL/ARM à leur architecture multi-cœurs MPPA. Ces travaux en collaboration étroite avec Kalray ont été en partie réalisé dans le cadre du stage de master 2 de Salma IZOUGGARHAN.

2.6 Extension des travaux aux autres familles d'ECC

Les résultats obtenus pour le décodage des codes LDPC nous a permis de prendre conscience de la pertinence de notre approche basée sur l'utilisation d'architectures multi-cœurs pour l'implantation d'algorithmes de décodage d'ECC dans des contextes SDR. Le recrutement en 2011 dans l'équipe de Camille LEROUX dont l'activité de recherche portait sur les codes polaires [Ari09] nous a permis d'initier une étude similaire autour de cette famille de codes correcteurs d'erreurs nouvellement proposées par [Ari08]. Les algorithmes de décodage des codes polaires possèdent des caractéristiques éloignées de ceux utilisés pour les codes LDPC : (a) les accès à la mémoire

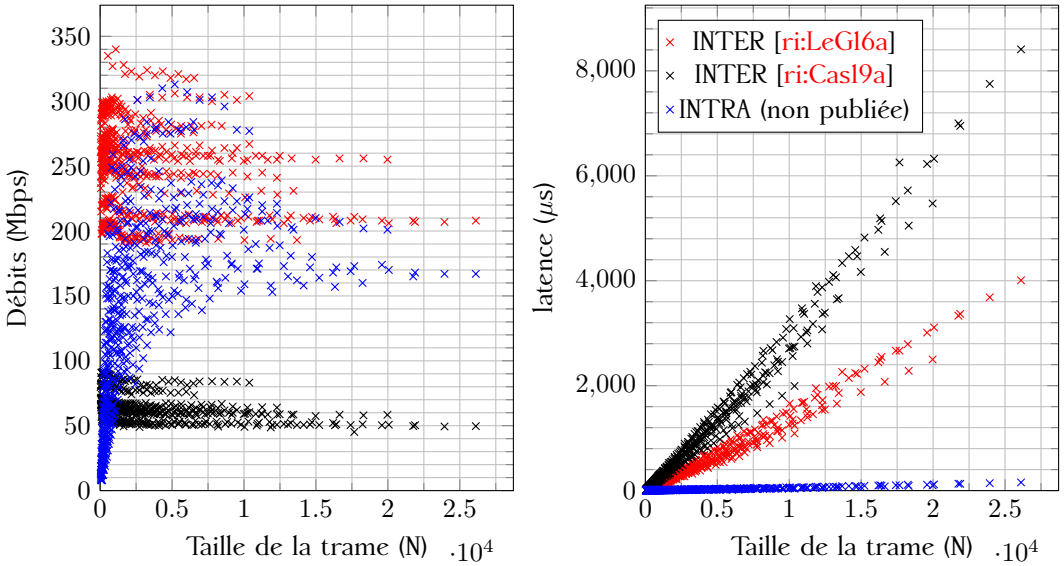


Figure 2.11 – Comparaison des débits et des latences des décodeurs LDPC logiciels décodant les codes LDPC du standard 5G en fonction de la stratégie de parallélisation employée - Cible INTEL Core-i7 i7-5650U - 1 cœur physique activé.

sont réguliers et (b) le parallélisme de calcul varie durant le décodage. L'algorithme de décodage SC [Alal1] peut se résumer au parcours d'un arbre binaire comme cela est présenté dans la figure 2.12. Le processus de décodage débute par l'exploration de la branche de gauche jusqu'à atteindre la feuille en bas à gauche puis il parcourt toutes les branches jusqu'à atteindre la feuille en bas à droite. Chaque nœud du graphe implique $\mathbf{N/L+1}$ calculs avec $\mathbf{L}$ le niveau du nœud dans l'arbre. Ainsi dans le nœud supérieur, $\mathbf{N}/2$ calculs identiques sur des données indépendantes sont à réaliser tandis que dans les feuilles un unique calcul est nécessaire.

De manière analogue aux décodages de codes LDPC, les traitements arithmétiques à réaliser sont de faible complexité et manipulent des données codées sur 8 bits. Contrairement aux travaux antérieurs [Gial4a], nous avons décidé d'appliquer une parallélisation inter-trames à l'algorithme

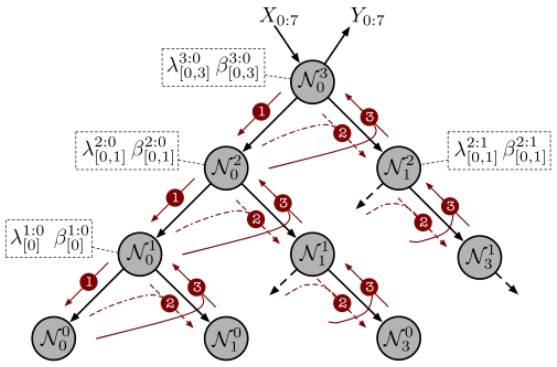


Figure 2.12 – Représentation de l'algorithme de décodage SC sous la forme du parcours récursif d'un arbre binaire.

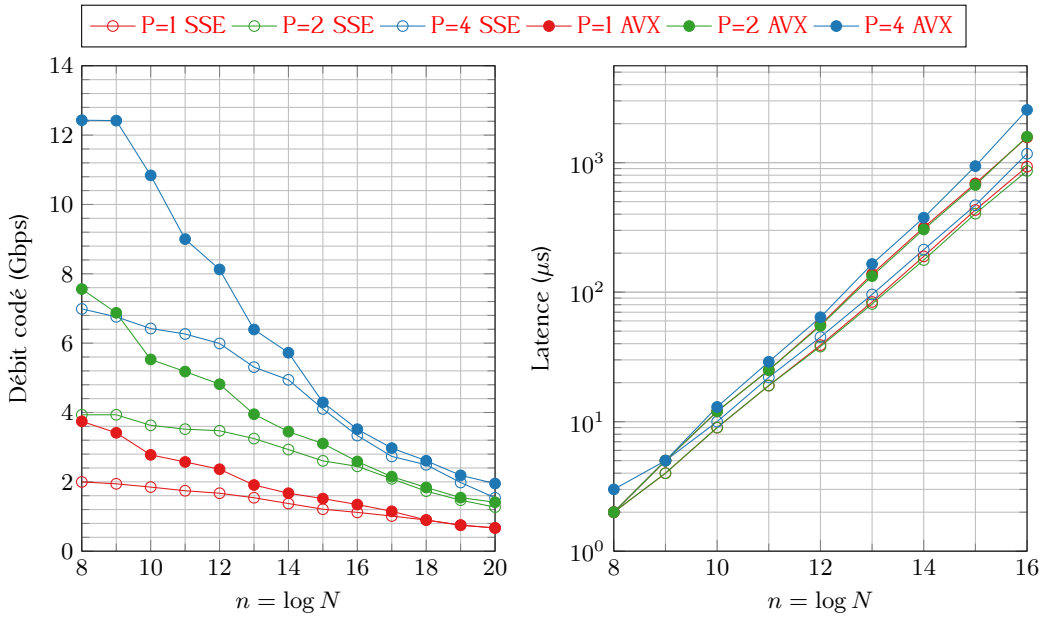


Figure 2.13 - Débits et latences des décodeurs SC proposés en fonction de **N** - Cible multi-cœurs x86 (INTEL Haswell Core-i7 4960HQ) - 4 cœurs physiques activés.

de décodage SC. L'objectif était de gagner en régularité et ainsi maximiser le débit du décodeur. Associés avec une utilisation adéquate des instructions SIMD et une compression mémoire des données facilitée par la stratégie de parallélisation, nous avons obtenu des débits 5× supérieurs à ceux rapportés dans [Gia14a]. La figure 2.13 présente l'évolution du débit et de la latence de décodage en fonction de **P**, **F** et de la taille des trames. Des débits atteignant jusqu'à 10 Gbps ont été obtenus sur un ordinateur portable équipé d'un processeur INTEL Core-i7 4960HQ lorsque les 4 cœurs physiques étaient exploités. Ces travaux ciblant les architectures INTEL x86 ont été présentés dans une publication dans la revue IEEE TSP [ri:LeGl5a]. Ceux dédiés aux cœurs à faible consommation (ARM) ont été publiés dans une conférence internationale [ci:LeGl4].

Ces travaux de recherche m'ont permis d'initier une collaboration avec l'INRIA à Bordeaux et plus précisément avec Denis BARTHOU. L'obtention d'un financement du Labex CPU nous a permis de recruter Adrien CASSAGNE sur un support d'ingénieur de recherche pendant un an. Durant cette période, nous nous sommes focalisés sur la parallélisation intra-trame du processus de décodage SC des codes polaires avec pour objectif de réduire la latence de décodage et améliorer l'efficacité énergétique des implantations logicielles. Les résultats obtenus grâce entre autre à une approche basée sur la métaprogrammation et génération de codes dédiés ont été valorisés dans deux conférences internationales [ci:Casl5] et [ci:Casl6a]. La pertinence des travaux et les résultats obtenus ont motivé dans un second temps, des travaux sur l'algorithme de décodage SC-SCAN gérant des sorties souples en sortie du processus de décodage. Les résultats de ces travaux évaluant à la fois les stratégies de parallélisation inter-trames et intra-trame ont été publiés dans une revue internationale [ri:LeGl9a].

Les travaux relatifs au décodage logiciel haute performance des codes LDPC ont abouti en 2015 à notre participation au projet ANR NAND. Ce projet ANR incluait les laboratoires ETIS, LabSTICC et l'IMT Atlantique, Turbo Concept, Thales et STmicroelectronics. Notre contribution dans le

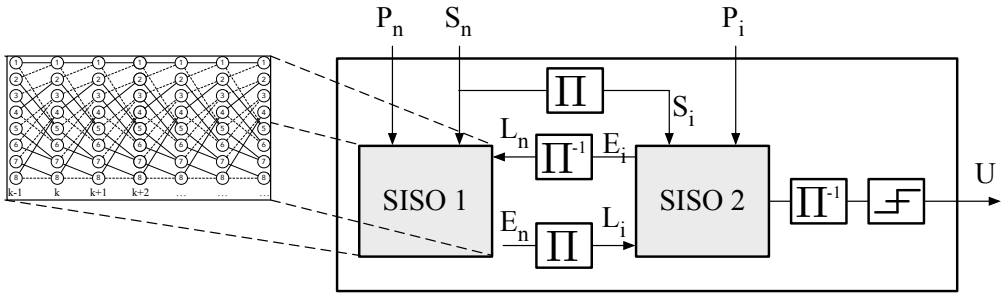


Figure 2.14 – Représentation du principe de fonctionnement d'un turbo décodeur.

cadre de ce projet collaboratif était de développer un environnement de simulation rapide et/ou d'exécution en temps réel de décodeurs de codes LDPC et de turbocodes. Ce projet qui a donné naissance à AFF3CT [ri:Casl9a] nous a permis d'investiguer les algorithmes de décodage des turbocodes. Ces derniers diffèrent fortement dans leurs formulations de ceux que nous avions précédemment implantés. Le décodage des turbocodes [Brel5] est un processus itératif qui implique durant chaque itération, le parcours de deux treillis (SISO), tous deux séparés par une phase d'entrelacement des données tel que schématisé dans la figure 2.14.

À l'époque, l'état de l'art traitant principalement de l'implantation des algorithmes de décodage des turbocodes sur cible GPU [Xial3,liul3,lil4], nous avons pris le parti de réaliser son implantation sur cible multi-cœurs. Pour faciliter l'exploitation du parallélisme de calcul lors du parcours du treillis et les accès mémoire lors de l'exécution de l'entrelacement des données, nous avons commencé par appliquer une parallélisation inter-trames. Ainsi la parallélisation des calculs lors des parcours *backward* et *forward* du treillis bénéficiait de l'intégralité des capacités SIMD de la plateforme multi-cœurs (SSE4 ou AVX2). L'exploitation efficace du jeu d'instructions SIMD disponible pour le traitement de données entières $\mathbf{Q} = 8$ bits et $\mathbf{Q} = 16$ bits a permis d'atteindre pour les codes du standard LTE des débits de 138 Mbps sur un ordinateur portable (INTEL Core i7-4960HQ). Par ailleurs, des débits jusqu'à 716 Mbps ont été obtenus sur un serveur de calcul (2× INTEL E5-2680v3). Ces résultats supérieurs à ceux rapportés antérieurement pour des cibles CPU et GPU ont été publiés dans la conférence internationale ISTC en 2016 [ci:Casl6a].

La solution de parallélisation imaginée et déployée permettait d'atteindre des débits supérieurs aux besoins applicatifs, mais impliquait : (1) une latence pouvant dépasser 3 ms lorsque $\mathbf{K} = 6144$, et (2) un manque de scalabilité de l'approche lorsque plusieurs cœurs physiques étaient employés en raison de la forte l'empreinte mémoire des décodeurs. Ces limitations étaient liées à la parallélisation inter-trames qui maximise le débit en décodant $\mathbf{F} = \{16, 32, 64\}$ trames en parallèle par cœur physique au détriment de la bande passante mémoire et des défauts de cache. Dans l'intention de lever les problèmes de contention mémoire, à partir d'une analyse des travaux réalisés dans le domaine de la conception matérielle de turbo décodeurs [Mul06, Brel5], nous avons identifié une approche [Mul06] permettant de ne traiter qu'une seule trame tout en conservant la régularité du décodage inter-trames. Cette stratégie consiste à décomposer le treillis en $\mathbf{F}$ sous sections indépendantes. Ces sous-sections sont synchronisées à la fin de chaque itération. L'utilisation de cette technique de sous fenêtrage a permis de réduire l'empreinte mémoire des décodeurs de facteurs $\{16, 32, 64\}$ de turbo-décodeurs dédiés au standard 4G LTE. La réduction des défauts de cache L1 et L2 ainsi obtenue nous a permis : (1) d'augmenter le débit des décodeurs d'un facteur 2 par rapport à la solution inter-trames et (2) de réduire d'un

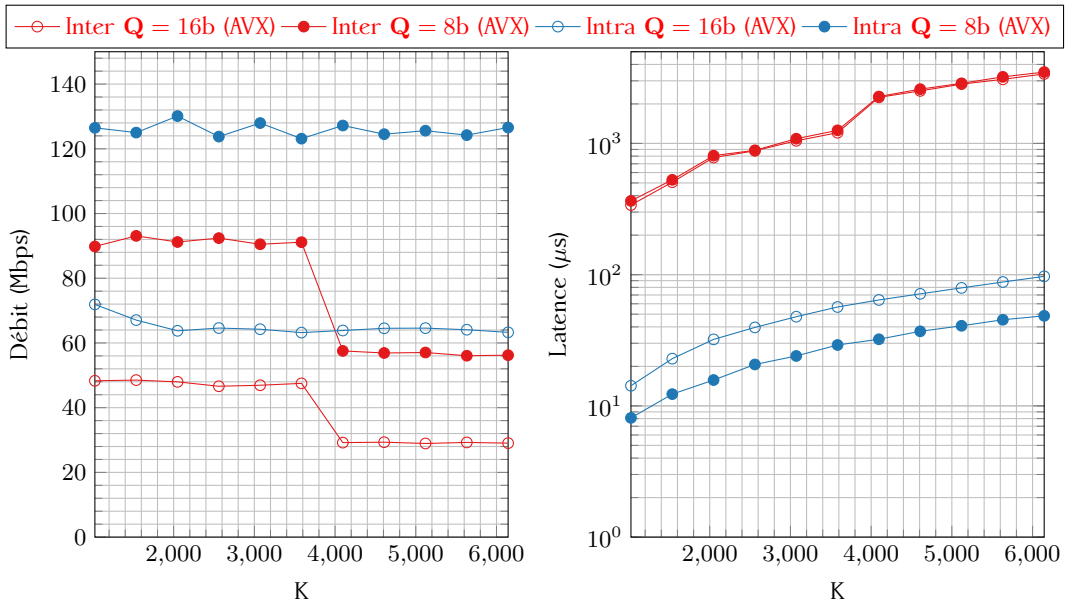


Figure 2.15 - Débits et latences des turbo décodeurs utilisant une parallélisation inter-trames et intra-trame lorsque 6 itérations de décodage sont exécutées sur une architecture INTEL Haswell Core-i7 7700T.

facteur 40 de la latence de traitement. Les courbes présentées dans la figure 2.15 mettent en évidence les gains obtenus pour différentes tailles de trame. Il est à noter que les débits mesurés lorsque plusieurs cœurs sont activés augmentent linéairement avec le nombre de cœurs. Les résultats de cette étude appliquée aux turbocodes du standard 4G LTE ainsi que la stratégie de parallélisation et les différentes optimisations employées ont été détaillés dans une publication en revue internationale [ri:LeG19c]. De plus, les résultats de cette étude ont donné lieu à une contractualisation avec l'entreprise Prisma Telecom qui a intégré ces travaux dans ses produits.

Des échanges dans le cadre du projet ANR NAND m'ont permis d'initier une collaboration avec Emmanuel BOUTILLON en 2016. Ce dernier travaillait activement sur l'implantation d'architectures matérielles pour le décodage de codes LDPC non binaires (LDPC-NB). Cette famille de codes correcteurs d'erreurs avait été intégrée dans le standard CCSDS [Orange] en 2015 pour fiabiliser les communications entre la Terre et l'espace. Ainsi, nous avons répondu à l'appel à projets exploratoires du GdR ISIS (SYMPA-DEC). Ce financement a permis de financer le stage de master 2 de Vincent PIGNOLY en 2017 ainsi que la mobilité à Bordeaux d'Assan HARB qui était doctorant au laboratoire LabSTICC à l'UBS. Le stage portait sur la conception d'un modèle comportemental pour la génération de décodeurs LDPC non binaires matériels. Cette collaboration nous a permis de monter en compétence sur les algorithmes de décodage des codes LDPC-NB. Dans un second temps nous avons tenté de les implanter efficacement sur cible multi-cœurs. Deux algorithmes de décodage ont ainsi été investigués : l'algorithme de décodage FFT-SPA et son approximation Min-Sum. Seule la stratégie de parallélisation intra-trame a été considérée, car le parallélisme de calcul était plus saillant et afin de minimiser la latence des décodeurs. Le travail d'optimisation algorithmique et de parallélisation s'est révélé pertinant, nous permettant d'atteindre des débits de 1.5× à 40× supérieurs vis-à-vis d'implantations sur des cibles GPU haut de gamme (NVIDIA GTX TITAN X). L'ensemble de ces travaux a été détaillé dans une publication dans la revue internationale [ri:LeG19b]. Ces décodeurs LDPC-NB logiciels

sont actuellement utilisés par le doctorant Camille MONIERE dans le cadre du projet ANR QCSP comme cela est indiqué dans le Chapitre 3.

2.7 Extension du jeu d'instructions des architectures programmables pour les ECC

Les travaux décrits précédemment ont démontré la pertinance et la viabilité des architectures programmables dans le contexte de systèmes de communication de type SDR ou Cloud-RAN où la flexibilité des infrastructures est prépondérante. Cependant les performances de ces implantations logicielles sont contraintes par le jeu d'instructions et par l'architecture interne des circuits multi-cœurs et many-cores. En effet, les jeux d'instructions de ces architectures programmables sont peu ou pas en adéquation avec les besoins intrinsèques des algorithmes de décodage ECC. Des opérations simples d'un point de vue algorithmique et/ou matériel requièrent un enchainement plus ou moins important d'instructions SIMD. Cela explique pourquoi malgré des fréquences de fonctionnement de 10× à 40× plus élevées, les cibles multi-cœurs rivalisent difficilement avec des architectures dédiées sur circuit FPGA [ci:Pig20b, ci:LeG22a]. Nous avons décidé d'orienter les travaux sur la conception d'extensions aux jeux d'instructions usuels afin d'améliorer l'adéquation entre les architectures programmables et les algorithmes de décodage des ECC. Pour atteindre cet objectif j'ai initié une collaboration scientifique avec le laboratoire LAB-STICC et plus particulièrement avec Cyrille CHAVET et Phillipe COUSSY, membres de l'équipe ARCANE. Nous avons obtenu en 2019 dans le cadre d'appels à projets propres à nos établissements respectifs, deux 1/2 financements thèse qui nous ont permis de financer la thèse de Mael TOURES.

Ces travaux, encore en cours, sont basés sur nos études précédentes autour de l'implantation de décodeurs ECC efficaces sur des cibles multi-cœurs INTEL et ARM. L'étude a pour objectif la définition d'extensions pertinentes pour le décodage des codes LDPC, des codes polaires, des turbocodes et des codes LDPC-NB. Afin d'amorcer ce travail, nous avons considéré que les nouvelles instructions ne peuvent posséder que deux registres pour les données sources. De plus, elles ne peuvent produire qu'une seule donnée. À partir d'un travail conséquent d'analyse et réécriture des décodeurs en langage C, différents groupes d'instructions ont été identifiés pour les différentes familles d'ECC. Le tableau 2.6 récapitule une liste non exhaustive des instructions pertinentes pour l'implantation des algorihtmes de décodage des codes LDPC et des codes polaires. À des fins de prototypage, nous avons intégré ces instructions à plusieurs cœurs de processeur RISC-V. Les résultats préliminaires démontrent des gains substantiels même lorsqu'une architecture RISC-V 32b qui limite le parallélisme à 4×8 bits est employée. Ce travail a été valorisé, à ce jour, par 2 communications en conférence [ci:Tou21a, ci:Tou22a].

L'extension des travaux à des architectures RISC-V 64b plus complexes (ex. intégrant par exemple la capacité à supporter des instructions à 3 opérandes) est en cours. Cette étude permettra dans un second temps une transposition des gains envisageables pour des architectures multi-cœurs INTEL et ARM.

Table 2.6 - Liste des instructions permettant d'améliorer l'adéquation entre les algorithmes de décodage de codes ECC et les architectures programmables.

	SISD (int8_t)	SIMD (int8x4_t)	Mnemonic	Implementation
LDPC	•	•	ld.subsat(rD, rSl, rS2)	Res = rSl - rS2 Res = (Res >127)? 127 : (Res <-127)? -127 : Res rD = Res
	•	•	ld.addsat(rD, rSl, rS2)	Res = rSl + rS2 Res = (Res >127)? 127 : (Res <-127)? -127 : Res rD = Res
	•	•	ld.abs(rD, rSl)	rD = (rSl >= 0)? rSl : - rSl
	•	•	ld.max (rD, rSl, rS2)	rD = (rSl >rS2)? rSl : rS2
	•	•	ld.min (rD, rSl, rS2)	rD = (rSl >rS2)? rS2 : rSl
		•	ld.sign (rD, rSl, rS2)	rD = (rSl <0)? 1 : 0
	•	•	ld.maskEval (rD, rSl, rS2)	rD = (rSl != rS2)? 0 : 0xFF
	•	•	ld.Rsign (rD, rSl, rS2)	rD = rSl xor (rS2 >= 0)
	•	•	ld.newMess (rD, rSl, rS2)	rD = (rSl == 1)? rS2 : - rS2
Polar	•	•	pl.f(rD, rSl, rS2)	min1 = abs(rSl) min2 = abs(rS2) min1 = min(min1 , min2) sign = (rSl <0) xor (rS2 <0) rD = (sign == 0)? min1 : -min1
	•	•	pl.r(rD, rSl ,rS2)	rD = (rS2)? 0 : (rSl <0)? 1 : 0
	•	•	pl.addsat(rD, rSl, rS2)	Res = rSl + rS2 Res = (Res >127)? 127 : (Res <-127)? -127 : Res rD = Res
	•	•	pl.subsat(rD, rSl, rS2)	same as ld.subsat
		•	pl.maskEval(rD, rSl, rS2)	rD = (rSl ==1)? 0xFF :0
		•	pl.decode (rD, rSl, rS2)	rD = (rS2 ==0)? rSl :0

2.8 Bilan des travaux

L'ensemble des encadrements menés dans le cadre des activités en relation avec la conception et utilisation d'architectures programmables est récapitulé dans cette partie. Il est à noter que dans la synthèse, l'ensemble des encadrements, des projets de recherche étatiques ou industriels n'ont pas été décrits. Les encadrements de doctorants et d'étudiants de niveau master 2 dont les travaux sont associés à cet axe de recherche sont donnés dans le tableau 2.7.

Une synthèse des valorisations scientifiques associées aux travaux realtifs à cet axe de recherche est fournie ci-dessous :

— 16 articles dans des revues internationales à comité de lecture ;
— 1 article dans des revue nationale à comité de lecture ;
— 22 articles dans des conférences & workshops internationaux avec actes ;
— 7 articles dans des conférences ou workshops nationaux avec actes ;
— 4 présentations lors de workshops nationaux ou journées du GdR.

Les travaux présentés se sont déroulés dans le cadre d'activités partenariales ou de contrats de

Table 2.7 - Liste des encadrements scientifiques relevant de travaux relatifs à la *conception et utilisation d'architectures programmables*.

Nom	Statut	Période	Financement	Co-Encadrant(s)	% encadr.
Julien MERCADAL	Post-doc	2011 – 12	LABRI	L. REVEILLERE	33%
Willy AUBRY	PhD	2008 – 12	ANR ARDMAN	D. NEGRU & al.	40%
Moez KTHIRI	PhD	2009 – 12	Fonds propres	H. LEVY & P. KADIONIK	40%
Jigar SOLANKI	PhD	2010 – 14	CNRS	L. REVEILLERE & D. BROMBERG	33%
Guillaume DELBERGUE	PhD	2015 – 17	CIFRE	C. JEGO	50%
Mael TOURRES	PhD	2019 – ...	Bdx-INP/UBS	P. COUSSY & C. CHAVET	25%
Logan Saint GERMAIN	PhD	2019 – ...	CIFRE	C. JEGO & al.	33%
Denis SHEMONAEV	PhD	2020 – ...	CIFRE	C. JEGO & J. CRENNE	33%
Oussama AIT SIDI ALI	PhD	2022 – ...	CIFRE	R. TAJAN	50%
Thomas NIEDDU	PhD	2022 – ...	ANR SEC-V	S. PILLEMENT	50%
Julien BEZINE	Master 2	2009	Fonds propres	–	100%
Maël TOURRES	Master 2	2019	Fonds propres	J. CRENNE	50%
Florian LOUPIAS	Master 2	2019	ANR QCSP	–	100%
Salma IZOUGGARHAN	Master 2	2021	KALRAY	–	100%
Thomas NIEDDU	Master 2	2022 – ...	ANR SEC-V	–	100%

recherche publics. La liste des contrats relatifs à cet axe de recherche est fournie ci-dessous :

— **Cantest-X** (DGA) (2002-2005), Développement d'un système d'étude et de caractérisation automatique des performances des Convertisseurs Analogiques Numériques, Budget IMS : 50 K€ ;
— **Cantest-XI** (DGA) (2010-2012), Intégration de nouvelles approches de caractérisation & de fonctionnalités d'analyse en temps réel dans l'outil Cantest-X, Budget IMS : 20 K€ ;
— **CHAMELEON** (ENSEIRB-MATMECA) (2010-2012), Passerelle reconfigurable multi-protocoles et multi-radios, Budget IMS : 20 K€ ;
— **FEC-CPU** (LabEx CPU de Bordeaux) (2015-2016), Accélération des simulateurs des systèmes de communications numériques, Budget IMS : 50 K€ ;
— **NAND** (ANR) (2016-2019), Noise Against Noise Decoder - Amélioration des performances des codes correcteurs d'erreurs par l'introduction de bruit de calcul, Budget IMS : 124 K€ ;
— **PRISMA** (Prisma technology) (2017), Développement de décodeurs LDPC logiciels efficaces pour le standard 5G, Budget IMS : 20 K€ ;
— **Keysight** (Keysight technology) (2019), Développement de décodeurs LDPC et turbocodes à faible latence efficaces pour les standards 4G LTE et 5G 3GPP, Budget IMS : 40 K€ ;
— **Neural networks for μC** (CATIE) (2019-2022), Définition d'une méthodologie d'optimisation et de déploiement de réseaux de neurones sur cibles microcontrôleurs (Accompagnement de la thèse de Logan SAINT GERMAIN), Budget IMS : 45 K€ ;
— **Multi-view object tracking** (EMG2) (2020-2023), Étude et implantation d'un système de tracking d'objets dans un environnement multi-cameras (Accompagnement de la thèse de Denis SHEMONAEV), Budget IMS : 60 K€ ;
— **KRATOS** (KRATOS) (2021), Software LDPC decoder for spatial communications (DVB-S2, CCSDS), Budget IMS : 40 K€ ;
— **SAFRAN** (SAFRAN) (2021) Software LDPC decoder for spatial communications (DVB-S2, CCSDS), Budget IMS : 30 K€ ;
— **Kalray (évaluation)** (Kalray) (2021), Évaluation des caractéristiques de la plateforme MMPA pour le décodage temps réel des codes LDPC du standard 5G 3GPP, Budget IMS : 5 K€ ;

— **Vastech** (Vastech Technology) (2022), Software LDPC decoder for spatial communications (DVB-S2, CCSDS), Budget IMS : 40 K€ ;
— **Kalray (parallélisation)** (Kalray) (2022), Proposition de stratégies de parallélisation efficaces dédiées à la plateforme MMPA en vue du décodage temps réel des codes LDPC du standard 5G 3GPP, Budget IMS : 20 K€ ;
— **SECV** (ANR) (2022-2025), Increasing security in open-hardware approaches, Budget IMS : 156 K€ ;

3 Algorithmes de décodage et prototypage matériel d'ECC

3.1 Introduction

Ce troisième chapitre dresse un bilan des projets que j'ai menés dans l'axe de recherche sur la proposition et/ou la modification d'algorithmes de décodage pour les codes correcteurs d'erreurs en vue de leurs implantations logicielles et matérielles. Dans un premier temps, le contexte de cet axe de recherche est décrit. Afin d'illustrer l'activité de recherche dans le domaine, la suite d'une sélection de trois études est plus spécifiquement détaillée. Une première section présente les travaux menés afin d'améliorer les performances de décodage des turbocodes. Puis, la conception d'architectures très haut débit pour le décodage des codes LDPC dans le contexte de communications optiques spatiales est abordée dans la deuxième section. Enfin, les travaux traitant de l'étude, de l'optimisation et du prototypage d'algorithmes innovants pour le décodage des codes LDPC sont détaillés dans une dernière section. Les thèses en cours dans le cadre de ce thème de recherche sont évoquées à la fin de cette section avant qu'une synthèse des activités menées durant la période 2006-2022 soit fournie. L'ensemble des étudiants dont le travail s'est inscrit dans cet axe de recherche est récapitulé dans la figure 3.1.

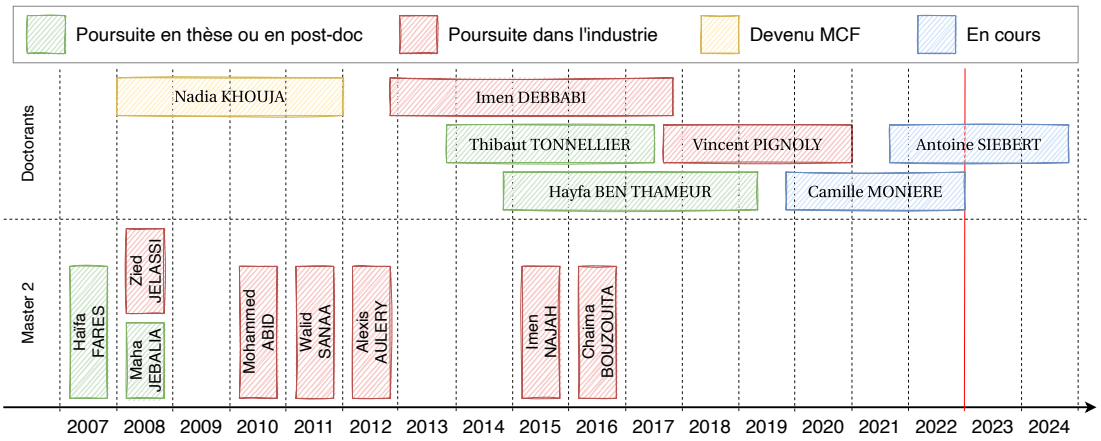


Figure 3.1 – Étudiants encadrés ayant participés aux études relatives à l'axe de recherche sur les algorithmes de décodage et le prototypage matériel de décodeurs ECC décrit dans le Chapitre 3.

3.2 Amélioration des performances de décodage des turbo décodeurs

Dans les thèmes précédents, j'ai présenté mes travaux de recherche en phase avec la conception d'architectures dédiées et/ou d'architectures flexibles. Ces travaux se focalisaient principalement sur la conception de circuits ou de systèmes implantant une fonctionnalité prédéfinie. Cependant, afin d'améliorer les performances applicatives, il est nécessaire de concevoir conjointement les algorithmes et les architectures matérielles. Pour atteindre cet objectif d'efficacité, il m'a été nécessaire de monter en compétences dans le domaine des systèmes de communications numériques afin de les associer aux compétences développées dans mes autres axes de recherche. Ces travaux ont débuté après la montée en compétence de l'équipe autour des architectures de décodeurs LDPC et la réalisation de démonstrateurs fonctionnels sur circuits FPGA (Chapitre **2**, section **3**). Ces travaux ont intéressé Benjamin GADAT récemment recruté à Thales Aliena Space. La volonté de collaborer a bénéficié du soutien de la direction de recherche de Bordeaux INP qui a accepté de financer la moitié du coût d'une thèse, l'autre moitié étant à la charge de Thales Aliena Space. Suite à de nombreuses discussions autour des problématiques de leurs systèmes de communication, nous avons décidé avec Camille LEROUX et Christophe JEGO d'œuvrer à l'amélioration des performances de décodage des turbocodes, travaux réalisés dans le cadre de la thèse de Thibaud TONNELLIER, doctorant que j'ai co-encadré à partir de novembre 2013. Dans la suite de cette section, je détaillerai les travaux de recherche réalisés pour répondre aux problèmes de performance des turbo décodeurs dans le cadre des standards LTE et CCSDS.

Les turbocodes [Ber93] sont une famille de codes correcteurs d'erreurs utilisés dans de nombreux standards de communications numériques actuels tels que les standards 4G LTE et DVB-RCS. Ils ils offrent de bonnes performances de décodage et les complexités calculatoires associées sont modérées. Cependant, même si les turbocodes disposent d'un excellent pouvoir de correction à bas SNR, ils souffrent malheureusement de la présence de planchers d'erreurs comme illustré dans la figure 3.2 où l'on observe un changement de la pente de la courbe lorsque le SNR augmente.

La présence d'erreurs résiduelles post-décodage provoquant ces planchers d'erreurs peut contraindre l'utilisation de turbocodes dans des contextes nécessitant de très faibles taux d'erreurs binaires. La présence de ces erreurs résiduelles est en grande partie due aux entrelaceurs. C'est la raison pour laquelle différentes optimisations favorisant la construction de turbocodes atténuant ce plancher d'erreurs [Garz16, Garz18] ont été proposées. D'autres solutions basées sur une concaténation série du turbocode avec un autre code correcteur d'erreurs [And96, Nar97, Ros11, Ton11, Boh15] ont aussi été proposées. Cependant, l'ensemble de ces approches n'est exploitable que lors de la définition d'un nouveau standard. Cela ne peut pas être appliqué aux turbocodes déjà standardisés tel que les standards 4G LTE [4G-LTE] ou DVB-RCS [DVB-RCS]. En parallèle de ces travaux, des stratégies alternatives exploitant la présence d'un code détecteur d'erreurs (CRC) dans les standards de communication ont été proposées. Une série de travaux s'est focalisée sur le décodage en parallèle de différentes hypothèses de décodage [Sad97, Nar98, Akm14]. Ces différentes hypothèses mémorisées et traitées sous forme de listes sont confrontées les unes aux autres et discriminées par un code CRC. Ces sélections peuvent intervenir pendant ou après le turbo décodage [Xu12, Fos98, Wei13]. L'amélioration des performances de décodage est obtenue au détriment d'une explosion de la complexité calculatoire et mémoire du récepteur réduisant l'intérêt de l'approche aux seuls décodages de trames de petites tailles. Dans le but

de limiter l'augmentation de la complexité du processus de décodage, des travaux ont proposé de procéder à un post-traitement des trames non décodées [Oul06, Oul07, Pfl13]. Si, à la fin du processus itératif, le décodeur n'a pas convergé vers le bon mot de code, les valeurs absolues des informations *a postériori* sont triées par ordre croissant. La valeur la moins fiable est alors forcée à une valeur supérieure à celle de la distance minimale du code. Un nouveau turbo décodage est appliqué et si le mot ne vérifie toujours pas le code détecteur d'erreurs, alors, la seconde position la moins fiable est considérée. Cette approche évite une augmentation de la complexité mémoire. Cependant, elle induit une augmentation de la complexité calculatoire et de la latence de décodage. Elle est quadratique avec le nombre de possibilités étudiées rendant la mise en application de cette proposition algorithmique inintéressante. L'activité menée au cours de ce projet de recherche a consisté à développer de nouvelles stratégies de décodage des turbocodes pour améliorer les performances de décodage de turbocodes déjà standardisés tout en maitrisant le surcoût de l'implémentation matérielle associée. De plus afin d'assurer une rétrocompatibilité de l'approche avec les turbocodes déjà standardisés, nous nous sommes interdit de modifier la chaine de communications numériques du côté de l'émetteur.

Afin d'identifier les raisons de l'apparition des planchers d'erreurs lors du processus de décodage, l'étude a débuté par une analyse de l'évolution des valeurs internes au cours du décodage. Cette action a été motivée par des travaux antérieurs [Rei01, Sha99] traitant de l'implantation de critères d'arrêt. Ces derniers ont mis en exergue que l'observation des oscillations des données internes lors des itérations de décodage permet de prédire si une trame ne pourra pas être décodée. En effet, comme cela a été démontré dans [Rei01], l'évolution des valeurs associées aux informations *a postériori* ($\mathbb{N} \rightarrow \mathbb{N}$) au cours du processus itératif :

1. convergent toutes rapidement vers une solution stable lorsque la trame est correctement décodée (figure 3.3b) ;
2. oscillent pour une partie d'entre-elles lorsque le décodage de la trame échoue (figure 3.3a).

L'analyse des oscillations [cn:Tonl5a, cn:Tonl5b] montre que, pour les trames corrigées (figure

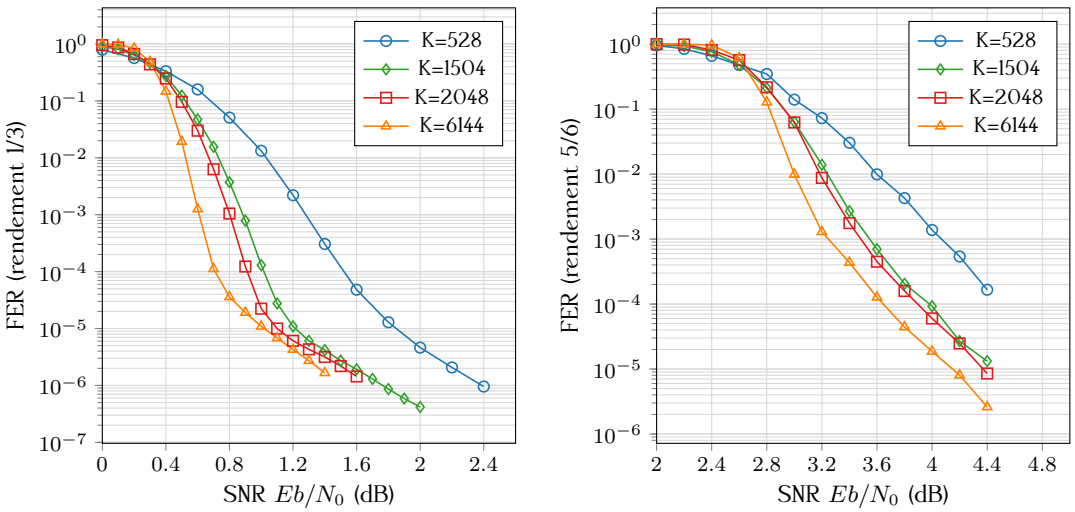
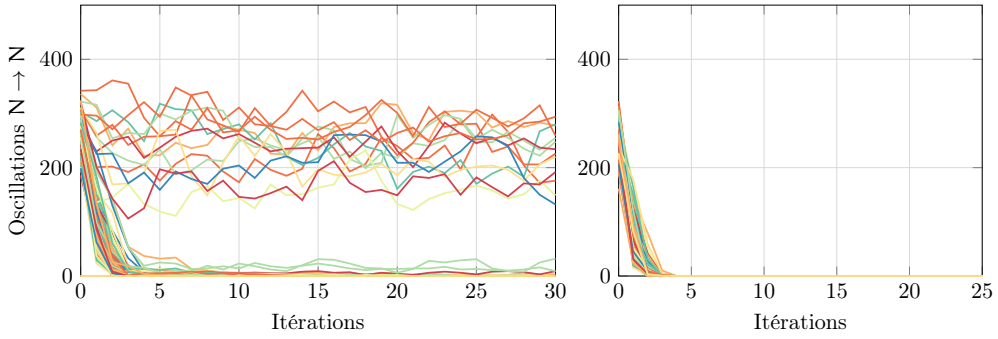


Figure 3.2 – Taux d'erreurs trame pour le standard LTE pour deux rendements et différentes tailles de trame. Décodage par l'algorithme EML-MAP effectuant 8 itérations.



(a) : Trames erronées (b) : Trames corrigées

Figure 3.3 - Oscillations au cours du processus itératif dans le cadre du standard LTE (K=1024, R=1/3) pour un taux d'erreur trame de 10^{-5}. Représentation des données issues de : (a) 100 trames erronées et (b) 100 trames corrigées.

3.3b), la convergence est rapide, moins de 5 itérations sont nécessaires contrairement aux trames erronées (figure 3.3a) qui oscillent plus ou moins fortement (environ dix fois plus).

Cette analyse a aussi démontré que 20% des trames ont plus de 20 bits erronés. Elles correspondent à celles présentant le plus grand nombre d'oscillations. De fait, la majorité des trames erronées possèdent un faible nombre de bits d'information erronés. Afin de réduire le nombre d'oscillations et aider le processus de décodage à converger en corrigeant les quelques bits erronés restants, nous avons transposé une approche proposée pour les codes LDPC en 2008 [Sav08], nommée *self-corrected*, aux turbocodes. Le principe de l'algorithme *self-corrected* est de détecter les messages non fiables échangés lors du décodage itératif et de les *effacer*. Dans le processus de décodage, l'annulation de la contribution peut être effectuée lorsqu'une oscillation $N \rightarrow N$ ou $I \rightarrow I$ est détectée. L'extension du principe de cette approche est donc comme suit : si l'information extrinsèque calculée par un décodeur SISO pour un bit de la trame change de signe d'une itération à la suivante, alors, celle-ci est annulée avant qu'elle ne soit transmise à l'autre décodeur SISO. Cet algorithme de décodage est détaillé dans [cn:Tonl5a, cn:Tonl5b]. Une évaluation exhaustive de l'approche proposée a été réalisée sur les turbocodes binaires des standards LTE et CCSDS. Si l'on applique 32 itérations de décodage, une amélioration des performances de décodage est observée comme cela est illustré dans la figure 3.4. Toutefois ces gains substantiels sont localisés dans la zone de convergence et non au niveau du plancher d'erreurs dans lequel aucun gain n'a été observé.

Ces résultats associés à l'absence de gain pour les turbocodes doubles binaires ont mis un coup d'arrêt à ces travaux. Cependant, les observations réalisées concernant le faible nombre d'erreurs résiduelles dans les trames non décodées ont motivé le développement d'une nouvelle approche. Lors de la première phase de l'étude, nous avons remarqué que le nombre moyen de bits erronés par trame erronée est généralement inférieur à 10 pour les turbocodes des standards 4G LTE et CCSDS lorsque l'on se situe dans la zone du plancher d'erreurs. La figure 3.4 illustre cette observation pour un turbocode du standard CCSDS. Une analyse équivalente a été menée pour les turbocodes non binaires (DVB-RCS) et les constatations sont similaires.

Les travaux présentés dans [Tak01] corroborent notre observation et lui fournissent un cadre théorique. Fort de cette observation, nous avons décidé d'étudier les approches permettant d'identifier les bits erronés une fois le processus de décodage terminé en prévision de les

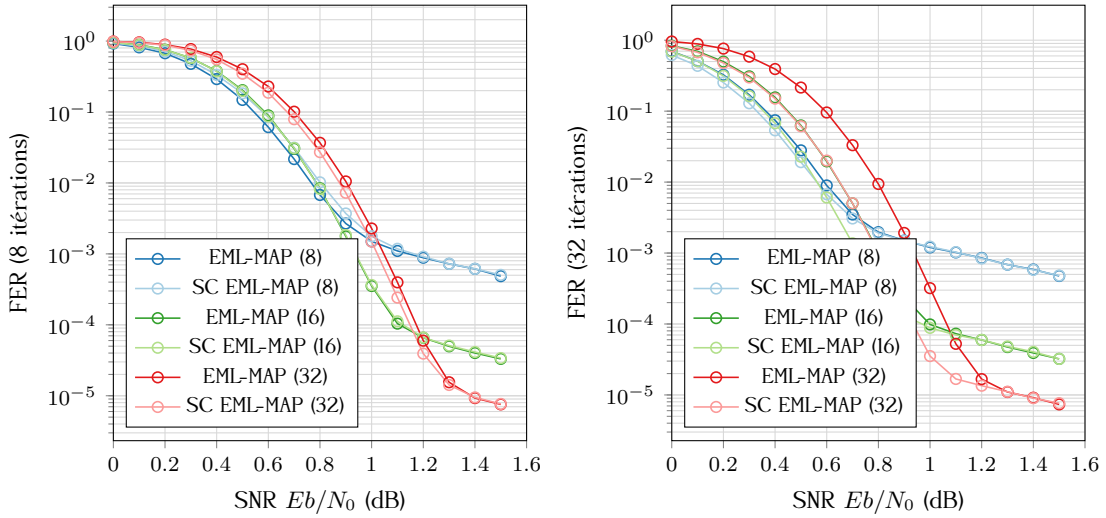


Figure 3.4 - Comparaison des performances de turbo-décodage entre l'algorithme EML-MAP et l'algorithme SC EML-MAP pour le turbocode $\mathbf{K} = 1504$, $\mathbf{R} = 1/3$ du standard LTE.

corriger *a postériori*. Cette méthode de correction *a postériori* exploite le fait que les turbocodes standardisés sont concaténés avec un code détecteur d'erreurs (CRC) permettant de s'assurer de la validité du résultat. Un algorithme dédié au post-traitement des bits décodés nommé *Flip and Check* (FNC) a été proposé. Cet algorithme de type brute force inverse séquentiellement la valeur des bits dans le but de générer un nouveau mot de code avant de valider ce dernier à l'aide du CRC.

La complexité calculatoire de cet algorithme évolue en $\mathbf{K}^{\mathbf{L}}$ avec $\mathbf{K}$ le nombre de bits composant

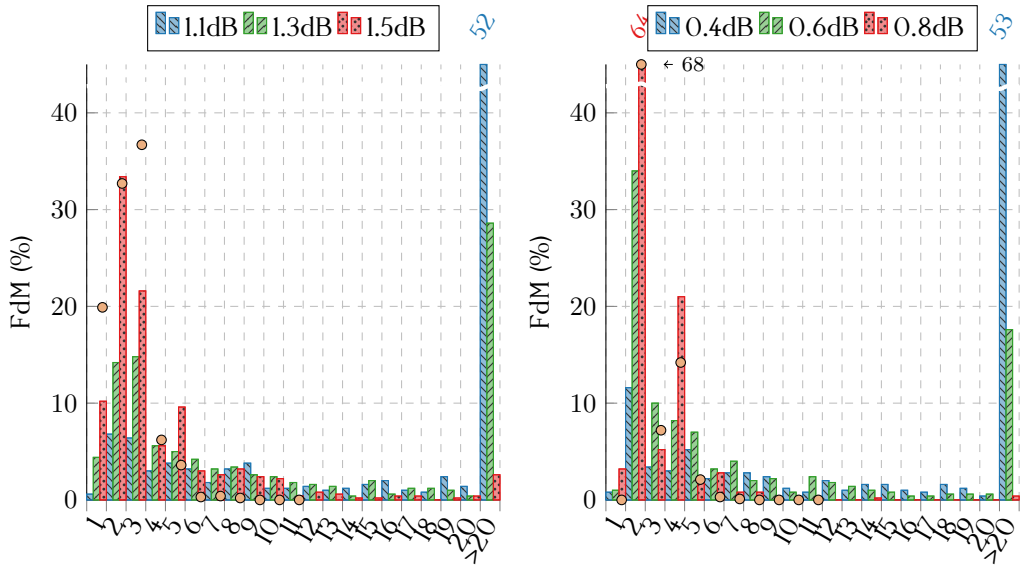


Figure 3.5 - Distribution du nombre d'erreurs binaires pour différentes valeurs de SNR pour les turbocodes $\mathbf{K} = 1024$ et $\mathbf{K} = 6144$ du standard LTE - Décodage EML-MAP itérant 8 fois.

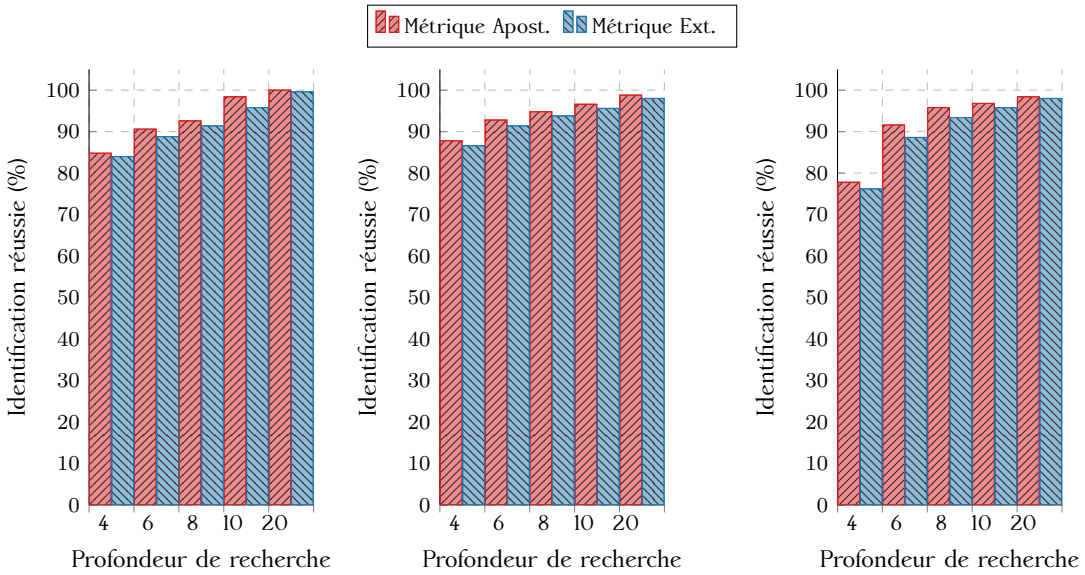


Figure 3.6 - Pourcentage d'identification réussie des erreurs résiduelles pour différents turbocodes du standard LTE $\mathbf{K} = 528$, $\mathbf{K} = 2048$ et $\mathbf{K} = 6144$ avec $\mathbf{R} = 1/3$. Décodage EML-MAP itérant 8 fois.

la trame et $\mathbf{L}$ le nombre de bits à permuter. Cette complexité calculatoire est rédhibitoire pour toute implantation logicielle ou matérielle lorsque $\mathbf{L}$ est grand. Afin de réduire la complexité calculatoire de l'approche, il est nécessaire de cibler au mieux les bits à inverser. Pour ce faire, nous avons étudié différentes métriques basées sur des données internes au turbo décodeur.

La figure 3.6 présente les statistiques d'identification réussies des deux métriques les plus pertinentes. Dans les trois cas considérés, nous remarquons qu'il est possible d'identifier précisément les bits erronés lors du processus itératif de décodage. En effet, au moins 90% des trames erronées ont l'ensemble de leurs erreurs identifiées par la métrique 1 pour $\mathbf{L} = 6$ alors que si $\mathbf{L} = 10$ nous approchons 98%. Ainsi pour corriger 90% ou 98% des erreurs dans le plancher d'erreurs, une complexité de 2^q calcul de CRC avec $q \in [6, 10]$ est suffisante limitant de fait les problèmes d'intégration. Des gains équivalents ont été observés pour le standard CCSDS. En ce qui concerne les tubocodes non binaires du standard DVB-RCS, une adaptation de la métrique d'identification a été nécessaire. En effet, le processus de décodage travaille en interne sur des symboles. Ces travaux, qui ont donné lieu à trois communications [ri:Tonl6, ci:Tonl6a, ci:Tonl6b], sont mis en valeur par les courbes issues de simulation de Monte-Carlo fournies dans la figure 3.7.

Un raffinement du processus de turbo décodage exploitant les formats de quantification en virgule fixe usuels [Brel5] a été réalisé afin de permettre le dimensionnement et la conception d'un prototype matériel de l'unité de post-traitement. Une étude des architectures de turbo décodeurs a été réalisée afin d'identifier les contraintes temporelles liées au post-traitement. Il s'est avéré que les architectures de turbo décodage nécessitent généralement $2\mathbf{K}$ cycles par itération de décodage et produisent les informations *a postériori* (et donc les métriques d'identification) durant $\mathbf{K}$ cycles à raison d'une donnée par cycle.

En conséquence le processus de correction des erreurs résiduelles est contraint à une latence de

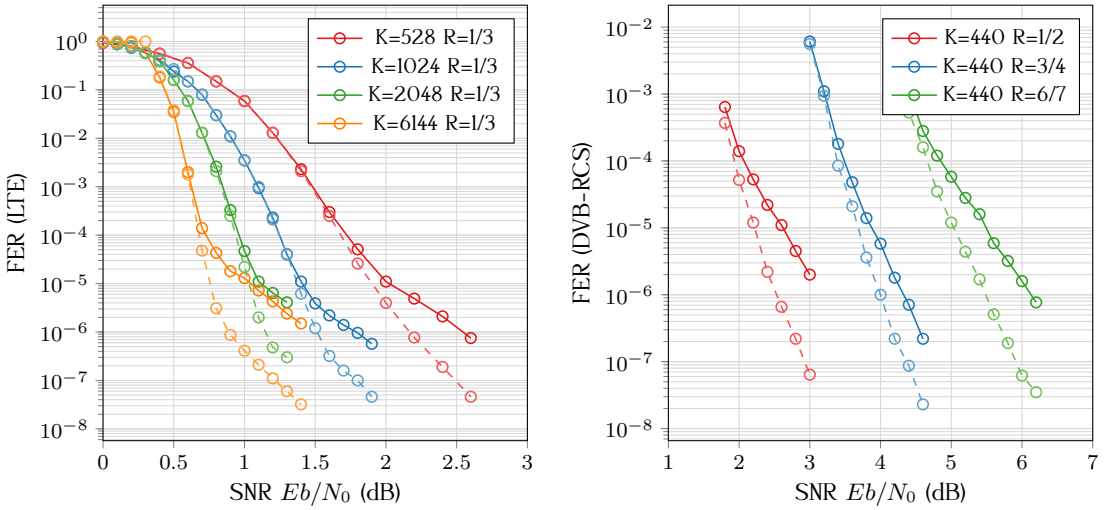


Figure 3.7 - Comparaison de performances de décodages entre EML-MAP (courbes pleines) et FNC (courbes en pointillés)- Décodeurs itérant jusqu'à 8 fois.

K cycles pour assurer un fonctionnement pipeline en parallèle avec celui du turbo-décodeur.

L'architecture que nous avons spécifiée puis prototypée est présentée dans la figure 3.8. Le principe de l'algorithme FNC est de générer 2^{q-1} mots candidats à partir du mot décidé à l'itération courante par le turbo décodeur. Cette génération n'est possible qu'après avoir effectué l'identification des q positions les moins fiables dans le mot décidé d. C'est la raison pour laquelle ce dernier doit être mémorisé. En parallèle de la réception des décisions d_i du turbo décodeur,

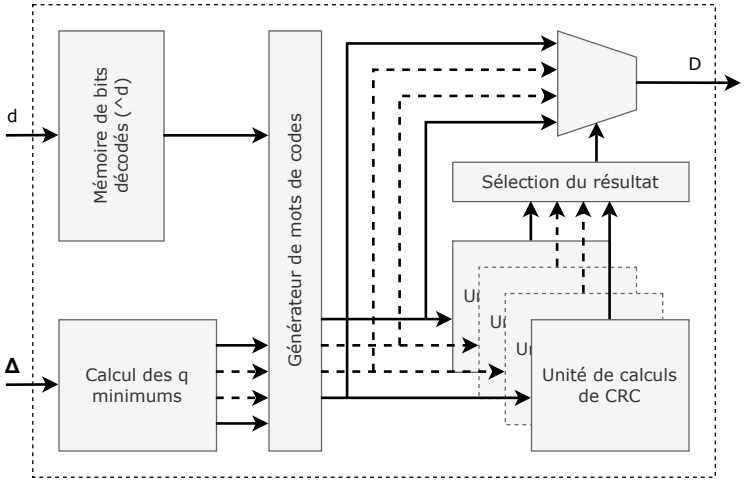


Figure 3.8 - Architecture matérielle générique implantant le traitement en temps réel de l'algorithme FNC en réalisant 2^q calculs de CRC en parallèle.

Table 3.1 - Résultats post placement-routage pour l'architecture associée à l'algorithme FNC en fonction de q - $\mathbf{K} = 6144$ - Circuit FPGA Xilinx Virtex-6 (xc6vlx75t-1).

q	LUT	FF	Slices	BRAM	fréquence
4	533	539	610	1	293 MHz
5	849	940	1038	1	230 MHz
6	1438	1725	1844	1	216 MHz
7	2302	3278	3127	1	207 MHz
8	4283	6267	5731	1	172 MHz
9	7970	12528	10850	1	132 MHz
10	15193	24833	23720	1	103 MHz

les métriques Δ_i reçues sont triées au fil de l'eau, pour ne conserver que les q positions des données les moins faibles. Une fois les positions des q éléments les moins fiables déterminées, le générateur de mots de code produit séquentiellement (bit à bit) les 2^{q-1} mots candidats en parallèle, en combinant le bit décidé et les positions des bits les moins fiables. L'ensemble de ces 2^{q-1} mots candidats est transmis à l'unité de vérification des CRC. Cette unité est composée de 2^q unités de calcul de CRC fonctionnant en parallèle. L'unité de contrôle est composée d'une machine à 3 états permettant de séquencer les traitements et le cas échéant de régénérer le mot de code valide. Cette architecture matérielle massivement parallèle de par sa conception respecte la contrainte temporelle de $\mathbf{K}$ cycles imposée pour l'évaluation des 2^q mots de code.

Le tableau 3.1 présente les résultats issus des prototypes implantés et validés sur circuit FPGA (Xilinx Virtex-6 xc6vlx75t-1) en fonction du paramètre q. La complexité matérielle de l'architecture croit exponentiellement avec la valeur du paramètre q. La fréquence de fonctionnement maximale décroit lorsque q augmente en raison de l'allongement du chemin critique présent dans le module de tri. À titre de comparaison, la configuration $q = 6$, qui permet de corriger de 90% des trames erronées, possède une complexité matérielle équivalente (en nombre de LUT) au turbo décodeur disponible sous forme d'un IP Core dans l'outil Xilinx Vivado.

L'ensemble des travaux réalisés durant la thèse de Thibaud TONNELLIER en collaboration avec Thales Alenia Space a donné lieu à 2 publications en revue [ri:Ton16, ri:Cas19a], 3 publications dans des conférences internationales avec comité de lecture [ci:Cas16b, ci:Ton16a, ci:Ton16b] et 2 publications en conférences nationales [cn:Ton15a, cn:Ton15b]. De plus, les résultats obtenus lors de ce travail ont motivé le partenaire industriel de l'étude à intégrer l'approche développée dans ses produits.

3.3 Étude et implantation de décodeurs LDPC matériels à très haut débit

Les résultats de l'étude partenariale avec Benjamin GADAT pour *Thales Aliena Space*, associés à notre capacité à concevoir des architectures matérielles efficaces pour le décodage des codes LDPC nous ont motivés à poursuivre des actions conjointes. Toutefois, cette collaboration avec Benjamin GADAT s'est poursuivie avec *Airbus Defense and Space* dans le domaine de communications optiques satellitaires. Nous avons proposé de travailler sur la définition de schémas de codage efficaces en termes de pouvoir de correction pour les liens montants et descendants ainsi que sur le développement des architectures matérielles de décodeur. La

principale difficulté de cette étude provenait des débits ciblés (10 Gbps). Cet objectif de débit devait être atteint par le recepteur la station de base au sol (lien descendant) mais aussi pour le le recepteur dans le satellite (lien montant). Dans ce dernier cas, associés à cette contrainte de débit, des contraintes de matérielles et énergétique fortes sont aussi présentes. En effet, dans le cadre du lien montant, le décodeur LDPC se situe au sein du satellite, limitant : (1) le choix de circuit FPGA et (2) la consommation d'énergie pour respecter les contraintes de dissipation thermique. Ce projet de recherche financé à moitié par Airbus et l'autre moitié sur fonds propres a permis le financement de la thèse de Vincent PIGNOLY qui a contribué à 2 projets R&T CNES (R&T FEC descendant et R&T FEC montant). J'ai co-encadré cette thèse avec Christophe JEGO et j'ai contribué aux deux R&T CNES.

L'applicabilité des systèmes de communications optiques satellitaires est en cours d'étude en vue de leur déploiement à moyen terme. Ces liens optiques doivent à terme remplacer les traditionnels liens RF pour permettre des débits de plusieurs centaines de Gbps entre la terre et les satellites. En plus de débits élevés, les liens optiques sont plus directifs que les liens RF réduisant de fait les perturbations et les interférences tout en limitant les risques d'interception. Cependant, lors de sa traversée de l'atmosphère l'onde optique est affectée par les turbulences atmosphériques qui peuvent engendrer des pertes importantes d'informations (phénomènes d'effacement). Dans l'intention de rendre cette technologie robuste et viable, il est nécessaire de repenser les formes d'onde et donc les chaînes d'émission et de réception. Durant les projets R&T CNES nous avons participé au dimensionnement des différentes parties des systèmes d'émission et de réception. Nos contributions étaient relatives à la spécification des entrelaceurs (taille, structure) et à la définition des contraintes liées aux codes correcteurs d'erreurs (ex. taille des trames). Ces travaux ne sont pas développés dans ce document.

La première étape des travaux de thèse de Vincent PIGNOLY a été d'évaluer la capacité des codes LDPC standardisés, tels ceux spécifiés dans le standard DVB-S2 [DVB-S2], à atteindre les niveaux de performance souhaités en termes de taux d'erreurs (BER, FER), de débit et de complexité. En effet, pour atteindre un débit de 10 Gbps sous contrainte de complexité matérielle, il est indispensable que le processus de décodage converge rapidement pour minimiser le nombre d'itérations de décodage. Les données issues de la littérature couplées à nos propres simulations ont montré que l'utilisation des codes LDPC issus du standard DVB-S2, malgré leur forte résilience à des quantifications faibles [Mar10], n'était pas opportune :

— les codes LDPC du standard DVB-S2 compatibles avec nos besoins présentent des planchers d'erreurs incompatibles avec les contraintes système sans l'utilisation d'un décodeur BCH supplémentaire ;
— pour converger, les codes LDPC du standard DVB-S2 nécessitent un nombre élevé d'itérations (> 10) afin de corriger efficacement les trames. Ce point limite les performances des décodeurs en termes de débit [DVB-S2, Mar15].

En collaboration avec Airbus Defence and Space, nous avons décidé de concevoir des codes LDPC dédiés aux besoins applicatifs. La famille des codes LDPC Quasi-Cyclique a été sélectionnée dans l'intention de simplifier dans un second temps la conception des décodeurs matériels. À partir des résultats de l'étude R&T CNES étudiant la forme d'onde complète et pour faciliter la montée en débit des décodeurs tout en limitant la présence de planchers d'erreurs, la taille des trames a été fixée à $\mathbf{K} = 16384$. Le facteur d'expansion qui règle le parallélisme de calcul de l'architecture matérielle a, quant à lui, été fixé à $\mathbf{Z} = 256$ pour éviter l'apparition de cycles courts

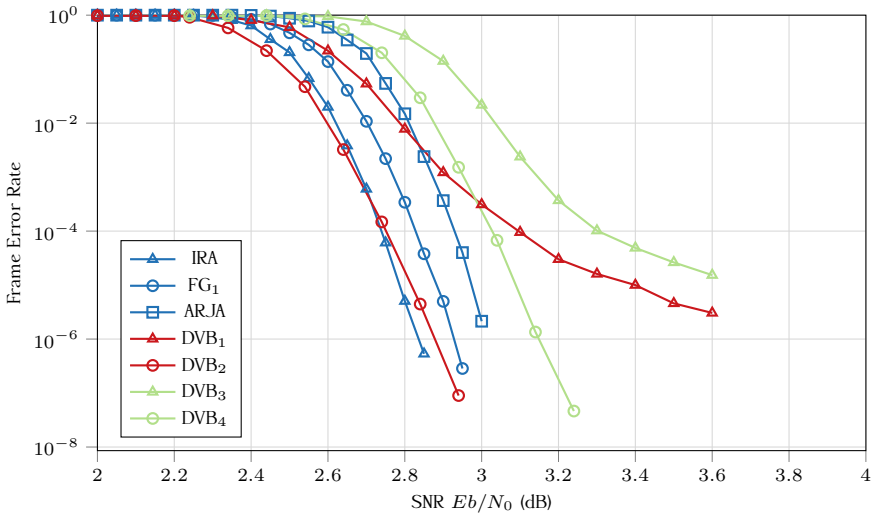


Figure 3.9 - Performances des codes LDPC considérés sur canal AWGN - Algorithme de décodage NMS par couches horizontales exécutant 10 itérations.

dans les codes. Trois méthodes de construction de codes LDPC (IRA [Jin00], ARJA [Div05,Div06] et FG [Kou01]) ont été employées. De nombreux codes LDPC ont été définis, évalués et comparés à l'aide de simulations de Monté-Carlo. Les algorithmes de décodage Offset Min-Sum (OMS) et Normalized Min-Sum (NMS) [Mar10] exploitant un ordonnancement horizontal par couche [Hoc04] ont été étudiés, car ils sont usuellement mis en œuvre dans les architectures matérielles. Les performances de deux codes LDPC sélectionnés et leurs positionnements par rapport aux codes du standard DVB-S2 sont fournis dans la figure 3.9.

Les codes LDPC (DVB$_1$ et DVB$_3$) issus du standard DVB-S2 [DVB-S2] présentent des planchers d'erreurs trop élevés par rapport au cadre de l'étude lorsqu'ils ne sont pas concaténés en série avec le code BCH. L'utilisation du code BCH défini dans le standard améliore drastiquement leurs performances comme le prouvent les courbes DVB$_2$ et DVB$_4$. Le code IRA que nous avons développé dépasse les performances du code DVB$_2$ (DVB$_2$ + BCH) à partir d'un SNR de 2.7 dB et cela malgré un rendement plus élevé. Le code FG$_1$ se positionne lui aussi favorablement à moins de 0.1 dB du code IRA. Ayant pour objectif de minimiser la complexité matérielle du décodeur, nous avons fait le choix d'opter pour les codes LDPC IRA & FG$_1$. En effet, leurs niveaux de performance sont équivalents à ceux des codes du standard DVB-S2 et surtout il ne nécessitent pas l'ajout dans le récepteur d'un décodeur BCH devant atteindre 10 Gbps.

Ces résultats théoriques nous ont amenés dans un second temps à utiliser des informations quantifiées en entrée du décodeur et dans le processus de décodage afin de mesurer plus précisément les performances de décodage. Les résultats de simulation fournis dans la figure 3.10 démontrent que les codes LDPC conçus présentent une forte résilience à la quantification de l'information. Les formats de quantification $Q_{8,6}$ et $Q_{7,5}$ garantissent des performances de décodage proches du modèle flottant pour le code LDPC FG$_1$. De plus ces formats de quantification ne font pas apparaitre de plancher d'erreurs avant BER $< 5.10^{-10}$. Le code LDPC de type IRA atteint des niveaux de performance équivalents, mais seul le format de quantification Q_{86} permet de respecter les contraintes liées à l'apparition du plancher d'erreurs.

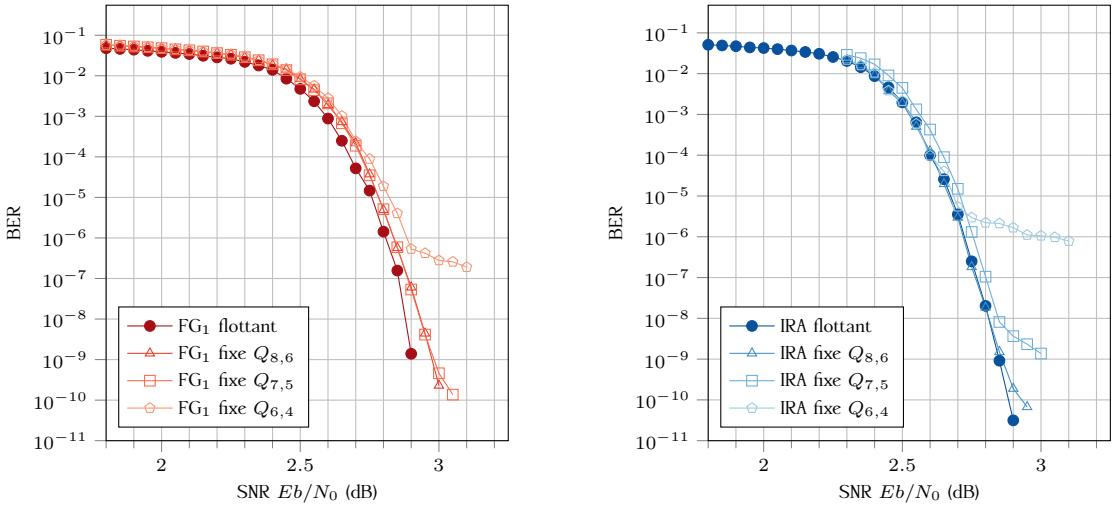


Figure 3.10 – Comparaison des performances de décodage des codes LDPC FG$_1$ et IRA en fonction du format de quantification employé sur un canal AWGN – Algorithme de décodage OMS avec un ordonnancement par couches horizontales.

Dans l'intention de développer un décodeur matériel atteignant un débit de 10 Gbps sur circuit FPGA, nous sommes repartis de l'architecture de décodeur que nous avions développé précédemment [ri:LeGl4a, ri:LeGl4b]. Toutefois, cette dernière avait été conçue initialement pour supporter tous types de codes LDPC. Elle offrait donc des niveaux de flexibilité et généricité inutiles et coûteux dans le cadre de ces travaux. En effet, les codes développés sont des LDPC-QC permettant par construction de simplifier l'accès aux données et l'exploitation du parallélisme de calcul. L'architecture initiale [ri:LeGl4b] a donc été spécialisée afin de réduire autant que possible la complexité matérielle et aussi maximiser la fréquence maximale de fonctionnement. Les principales transformations appliquées à l'architecture de décodage sont listées ci-dessous :

— Le premier entrelaceur de type Benes employé dans [ri:LeGl4b] a été remplacé par un *Barrel Shifter*. Ce dernier est généré automatiquement en fonction des besoins réels du code LDPC implanté. Ainsi les combinaisons inutiles sont automatiquement supprimées afin de minimiser la complexité matérielle post-implantation.
— Le second entrelaceur liant les **Z** éléments de calcul aux **Z** bancs mémoires a été supprimé à l'aide de l'optimisation décrite dans [Marl1a].
— Des tranches de pipeline ont été ajoutées au niveau des éléments de calcul et des entrelaceurs dans l'intention de supprimer des chemins critiques.

Le flot de conception automatisé [ri:LeGl4b], amendé en conséquence, nous a permis de prototyper plusieurs décodeurs LDPC sur circuit FPGA afin d'estimer précisément les performances post-placement routage, mais aussi valider leur comportement grâce à la chaine de simulation en temps réel développée par Yann DELOMIER [ci:Del18b, ri:Del20a]. Le tableau 3.2 présente les performances des décodeurs matériels retenus avec notre partenaire industriel. Les gains obtenus vis-à-vis de l'architecture initiale sont mis en évidence dans le tableau 3.3 qui rapporte les performances du code LDPC FG$_1$. L'architecture finale nécessite 6× moins de mémoire et 2.5× moins de LUT que la version originale de l'architecture [ri:LeGl4b]. De plus la fréquence de fonctionnement a été augmentée d'un facteur 1.8× permettant l'obtention d'un débit de 11,7

Code	Format de quantif. ($Q_{v,m}$)	Complexité (1× décodeur) LUT (%FPGA)	FF (%FPGA)	RAM18K (%FPGA)	Performance débit (Gbps)	latence (μs)	Débit max. sur xczu9eg-2 (Gbps)	#dec
IRA	$(8,6)$	49121 (17.9%)	26903 (4.9%)	50 (5.5%)	1.64	10.96	6.6	4×
FG	$(8,6)$	49121 (17.9%)	26903 (4.9%)	50 (5.5%)	2.13	8.44	8.5	4×
FG	$(7,5)$	38913 (14.2%)	23483 (4.3%)	43 (4.7%)	2.13	8.44	10.6	5×

Table 3.2 – Performances des décodeurs matériels pour les deux codes LDPC retenus sur cible Zynq Ultrascale + avec $F = 250$ MHz.

Gbps lorsque 5 décodeurs sont instanciés au sein du circuit Zynq Ultrascale+ ciblé. L'ensemble de ces travaux a été valorisé par une publication à la conférence ISTC [ci:Pig18a].

Les travaux menés ciblaient la conception d'une station au sol pour la communication par lien optique. Nous avons donc comparé les propriétés de la solution développée sur circuit FPGA à une solution de décodage purement logicielle sur cible multi-cœurs qui aurait pu être déployée [ri:LeGl16a, ci:LeGl17a]. Les résultats comparatifs en termes de débits, de latences et d'énergie consommée par bit décodé sont présentés dans le tableau 3.4. Les résultats ont été obtenus sur un CPU multi-cœurs INTEL Xeon Gold 6148 et un circuit FPGA XILINX Ultrascale+ (xczu9eg-3ffvb1156e). Le prix de ces composants était à l'époque respectivement d'environ 7500$ et 5200$. Les débits atteints par ces deux types de solution sont équivalents. La solution à base d'architecture matérielle dédiée sur circuit FPGA se démarque par une latence de traitement 8× à 20× inférieure et une consommation d'énergie de 16× à 55× inférieure. Cette étude comparative a fait l'objet d'une publication dans une conférence internationale [ci:Pig20b].

La seconde partie de l'étude menée en concertation avec Airbus Defence and Space et le CNES adressait la fiabilisation du lien montant vers le satellite. Ce lien dont le débit codé devait aussi atteindre 10 Gbps tout en respectant des contraintes plus sévères en termes de complexité matérielle et de dissipation thermique. Le circuit FPGA sélectionné était le circuit Kintex Ultrascale KU060 (space grade xqrku060), mais seulement 30% de la matrice est allouée à la fonction de décodage. De plus, la fréquence d'horloge maximum est bridée à 250 MHz pour éviter des problèmes de dissipation thermique. Contrairement aux systèmes de communications numériques

Cible	Complexité (1× décodeur) LUT (%FPGA)	FF (%FPGA)	RAM18K (%FPGA)	Performances débit (Gbps)	latence (μs)	Débit max. sur freq. (MHz)	xczu9eg-2 (Gbps)	#dec
NISC [ri:LeGl14b]	96078 (35.0%)	28949 (5.3%)	267 (29.3) %	1.34	13.46	158	2.67	2
Publication [ci:Pig18a]	49121 (17.9 %)	26903 (4.9%)	50 (5.5) %	2.13	8.44	252	8.5	4
Architecture finale	40416 (14.7%)	30385 (5.5%)	50 (5.5) %	2.34	7.71	274	11.7	5

Table 3.3 – Évolution des performances d'implantation des décodeurs matériels sur un circuit Zynq Ultrascale+ avec le code FG_1 en $Q_{8,6}$ pour 10 itérations de décodage.

Cible	Code LDPC	Nombre de cœurs	Fréquence (MHz)	Débit (Gbps)	Latence (μs)	Puissance (Watts)	Énergie (nJ/bit)
FPGA	(16384, 4096)	5	274	11.65	7.7	13.80	1.19
	(64800, 29160)	3	270	4.08	47.7	12.92	3.20
	(22528, 6144)	2	273	8.04	6.1	6.62	0.83
multi-cœurs	(16384, 4096)	40	2194	11.25	60	343	30.49
	(64800, 29160)	40	2194	7.60	345	370	48.69
	(22528, 6144)	40	2194	7.55	122	345	45.70

Table 3.4 – Comparaison des implantations de décodeurs LDPC de type OMS sur un circuit programmable (INTEL Xeon Gold) et un circuit reconfigurable (FPGA XILINX Ultrascale+).

RF usuels, l'étage amont ne peut fournir qu'une information dure (0/1) au décodeur. L'ensemble de ces contraintes interdisaient la réutilisation de la solution développée pour la station de base au sol.

Afin d'atteindre les objectifs en termes de débit et de complexité matérielle, l'utilisation d'une architecture semi-parallèle a été privilégiée. Cependant, les algorithmes proposés dans la littérature pour le décodage efficace d'informations dures [Gal62, Wad10, Ras14, Una17] ne répondaient pas aux contraintes de performances (ex. pas de plancher d'erreurs avant BER=10^{-10}) et/ou étaient incompatibles avec une implantation semi-parallèle. L'algorithme le plus pertinent était l'algorithme Gallager-B [Gal62], proposé spécifiquement pour le traitement d'informations dures via l'échange de messages binaires entre les éléments VNs $\longleftrightarrow$ CNs. Toutefois, ce processus itératif impliquant l'échange de messages binaires s'est avéré insuffisant en termes de pouvoir de correction. Nous avons donc étudié son évolution détaillée dans [Mit98, Ric01]. Cet algorithme, appelé Gallager-E, augmente la dynamique des messages internes au décodeur. La ternarisation des messages échangés entre les nœuds VN et CN permet de modéliser l'incertitude. Cette transformation du format des données induit une augmentation de la complexité du décodeur, mais améliore aussi drastiquement les performances de décodage, répondant positivement aux contraintes imposées par l'étude. Pour réutiliser en partie l'architecture de décodeur en notre possession, une reformulation de l'algorithme de manière à exécuter un ordonnancement par couches horizontales a été proposée [ci:LeG22a]. Cette reformulation de l'algorithme initial, présentée dans l'algorithme 1, permet de réduire la taille des mémoires stockant les messages échangés, mais aussi d'accélérer la convergence de l'algorithme initial [Mit98, Ric01].

Une des propriétés intéressantes de cette formulation de l'algorithme Gallager-E est sa capacité à converger très rapidement, nécessitant de fait un faible nombre d'itérations. La figure 3.11 compare les performances en termes de FER de l'algorithme de décodage Gallager-E dans ses 2 formulations. Nous observons qu'avec 10 itérations en ordonnancement par couches horizontales (nécessaire pour atteindre des débits $\geq$ 10 Gbps), il est possible d'obtenir des performances de décodage assez proches de celles obtenues lorsque 300 itérations sont exécutées.

Ces résultats nous ont incités à prototyper une architecture matérielle du décodeur sur circuit FPGA. Cette architecture de type semi-parallèle est proche de celle utilisée pour le décodeur LDPC du lien descendant [ci:Pig18a]. Cependant des adaptations ont été nécessaires tant sur les aspects liés à la mémorisation des données que sur le traitement de l'information. Par exemple, les unités de calcul ont été repensées afin d'effectuer les opérations de l'algorithme Gallager-E. L'architecture interne de ces unités est détaillée dans la figure 3.12. Par rapport aux unités de

Algorithm 1 Algorithme de décodage Gallager-E avec un ordonnancement par couches horizontales

▷ Input (received word) : $\boldsymbol{y} = (y_1, y_1, ..., y_N) \in \{0,1\}^N$
init
$t = 0$, $Y_i = 2y_i - 1$ and $V_i = 0$ for $i \in [1, .., N]$
repeat
▷ Horizontal layered scheduling
for all j $\in [1, .., M]$ **do**
$C_j^{(t)} = 1$, $Sum0 = 0$
▷ Stage Sl
for all i $\in$ N(j) **do**
▷ Variable to parity check message $v2c_{ij}^{(t)}$ processing

$$M_{ij} = \begin{cases} V_i, & t = 0 \\ V_i - c2v_{ji}^{(t-1)}, & \text{otherwise} \end{cases}$$

$$v2c_{ij}^{(t)} = sign(M_{ij} + \omega^{(t)} * Y_i)$$

▷ Check node $C_j^{(t)}$ processing

$$C_j^{(t)} = \begin{cases} C_j^{(t)}, & M_{ij} \geqslant 0 \\ -C_j^{(t)}, & \text{otherwise} \end{cases}$$

if $M_{ij} = 0$ **then** $Sum0 = Sum0 + 1$ **end if**
end for
▷ Stage S2
for all i $\in$ N(j) **do**
▷ Check to variable message $c2v_{ji}^{(t+1)}$ processing

$$c2v_{ji}^{(t)} = \begin{cases} 0, & Sum0 \geqslant 2 \\ 0, & Sum0 = 1 \text{ and } M_{ij}^{(t)} \neq 0 \\ C_j^{(t)}, & Sum0 = 1 \text{ and } M_{ij}^{(t)} = 0 \\ C_j^{(t)} \times M_{ij}^{(t)}, & \text{otherwise} \end{cases}$$

▷ Variable node accumulator V_i updating

$$V_i = M_{ij} + c2v_{ji}^{(t+1)}$$

end for
end for
$t = t + 1$
until $t \leq t_{max}$

$$\forall i \in [1, .., N],\ x_i = \begin{cases} y_i, & Y_i + V_i = 0 \\ 0, & Y_i + V_i > 0 \\ 1, & Y_i + V_i < 0 \end{cases}$$

▷ Output (decoded word) : $\boldsymbol{x} = (x_1, x_1, ..., x_N) \in \{0,1\}^N$

traitement du décodeur OMS [ci:Pig18a], des FIFO ont été ajoutées pour pouvoir transmettre à la fois les valeurs du canal, mais aussi les messages entrants à la seconde tranche de calculs. Ce passage d'information est rendu nécessaire pour maintenir une cohérence au niveau du stockage des données dans la mémoire VN, mais aussi pour générer le message de sortie. En effet, l'opération de vote impliquée dans l'algorithme Gallager-E est non inversible.

Les résultats obtenus après synthèse et placement routage sont fournis dans le tableau 3.5 pour les deux codes LDPC considérés lorsque la fréquence est contrainte à 250 MHz. L'utilisation de 7 décodeurs LDPC en parallèle permet d'atteindre un débit de 10 Gbps. Dans cette configuration, 98% des LUT et 47% BRAM à disposition sont utilisés. La diminution d'une itération de décodage permet de réduire le nombre de décodeurs d'une unité, réduisant à 84% et 40% les taux d'utilisation des ressources. Dans l'hypothèse où la fréquence maximale de fonctionnement (307 MHz) pouvait être utilisée, seuls 5 décodeurs permettraient d'atteindre le débit de 10 Gbps.

Une comparaison des caractéristiques matérielles du décodeur Gallager-E avec ses pairs (tableau 3.6) exécutant respectivement un algorithme de décodage de type Gallager-B et Offset Min-Sum

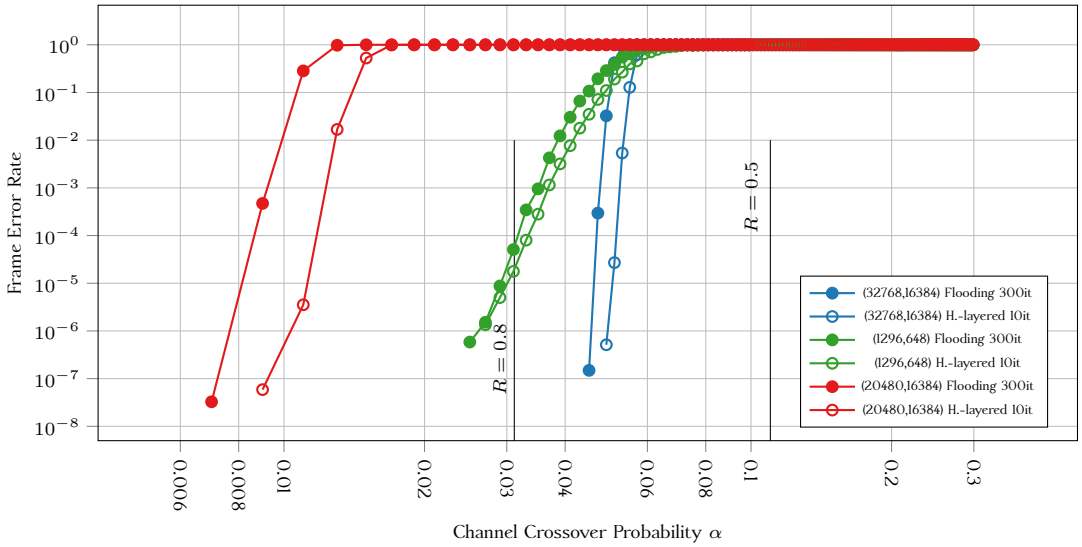


Figure 3.11 – Performances de correction d'erreurs pour différents codes LDPC en fonction de l'ordonnancement des calculs et du nombre d'itérations executées.

permet d'observer le compromis en termes de complexité matérielle qu'offre cette solution. La complexité logique est réduite d'un facteur 3× par rapport au décodeur OMS tandis que la complexité mémoire est 50% inférieure. A l'opposé, le décodeur Gallager-E est 50% plus complexe qu'un décodeur de type Gallager-B en raison de la gestion de l'incertitude durant le processus de décodage. Toutefois, l'utilisation de ce dernier est rendue impossible vu de son piètre pouvoir de correction.

Rappelons que cette étude a été menée dans le cadre d'un projet de recherche commun avec une

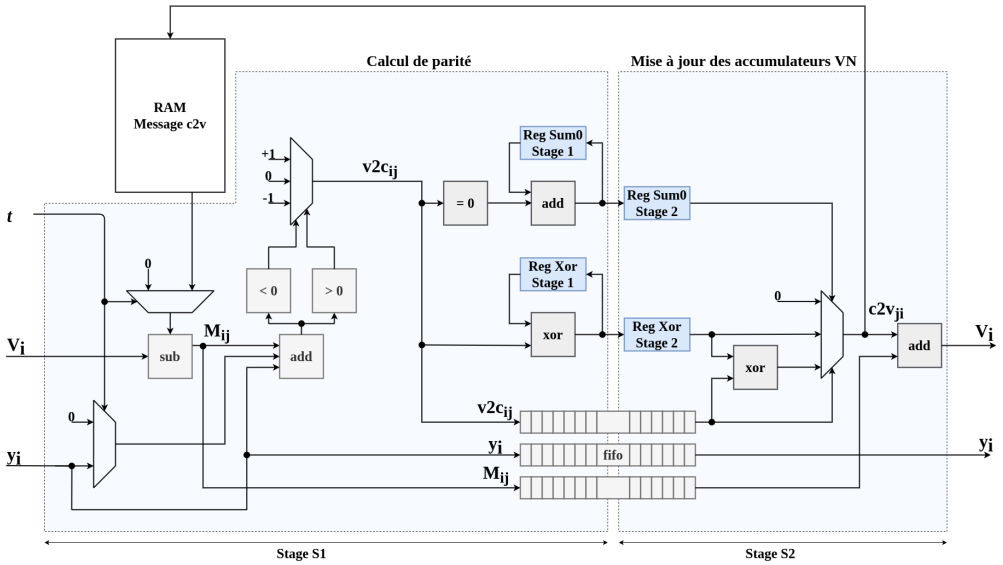


Figure 3.12 – Architecture interne des éléments de calcul développés pour supporter l'algorithme de décodage Gallager-E.

Table 3.5 - Résultats d'implantation post-placement routage des décodeurs matériels - Fréquence de fonctionnement fixée à 250 MHz - 10 itérations de décodage - Cible FPGA Xilinx KU060.

Code	Z	Complexité			Performances	
		LUT	FF	BRAM	débit	latence
(32768, 16384)	256	13979	10573	42	1.53 Gbps	21.42 μs
(20480, 4096)	256	13979	10573	34	1.51 Gbps	13.55 μs

équipe toulousaine d'Airbus Defence and Space en parallèle de 2 projets R&T CNES. Tout au long de l'étude, nous avons eu des réunions d'avancement au cours desquelles nous avons pu confronter nos investigations aux contraintes applicatives. Le travail de recherche effectué durant cette étude a donné lieu à 4 publications en conférences internationales [ci:Pig18a,ci:Pig20a,ci:Pig20b,ci:LeG22a], 1 article dans une revue internationale [ri:Cas19a] et une publication dans une revue internationnalle [ri:LeG23a] ainsi que 2 présentations dans des journées d'étude nationales [pn:Pign19a,pn:Pign19b]. Les descriptions matérielles des décodeurs LPDC (lien montant et descendant) ont été fournies à Airbus afin de permettre leur intégration dans les prototypes de systèmes de communication optique en cours de développement.

3.4 Étude, évaluation et implantation d'algorithmes de décodage innovants

En parallèle de ces travaux visant à implanter efficacement les algorithmes de décodage usuels sur des circuits matériels sous contraintes de performances, j'ai initié des travaux visant à étudier, à amender et à prototyper des approches innovantes du point de vue algorithmique. Ces études se sont déroulées dans le cadre d'une collaboration avec le laboratoire SUP'COM de Tunis. Cette collaboration initiée en 2007 avec le co-encadrement de la thèse de Nadia KHOUJA s'est ainsi poursuivie. Les travaux de recherche conjoints se sont focalisés sur l'étude et l'implantation matérielle de décodeurs LDPC optimaux du point de vue maximum de vraisemblance (ML) respectivement pour le décodage des codes LDPC et sa transposition pour le décodage des codes LDPC convolutifs (LDPC-CC). L'originalité de ces travaux vient de l'utilisation d'un algorithme de décodage basé sur une formulation LP du problème de décodage LDPC [Fel03,Fel05], formulation modifiée en 2013 par S. Barman [Bar13] à l'aide de l'approche ADMM. La possibilité de raffiner cette formulation algorithmique pour concevoir un décodeur matériel temps réel sans altérer les

Table 3.6 - Complexité matérielle des décodeurs LDPC en focntion de l'algorithme de décodage - Code LDPC ($\mathbf{K} = 16384$, $\mathbf{N} = 20480$) - 10 itérations de décodage - Circuit FPGA Xilinx KU060.

	Gallager-B	Gallager-E	OMS
# LUT	7 833 (8%)	13 979 (14%)	49 137 (49%)
# FF	8 379 (8%)	10 573 (11%)	26 919 (27%)
# BRAM	23 (5%)	34 (7%)	73 (14%)
Quantification	4 bits VN 1 bit MSG	5 bits VN 2 bits MSG	8 bits VN 6 bits MSG
Fréquence max.	347 MHz	309 MHz	156 MHz
Débit	2.10 Gbit/s	1.87 Gbit/s	0.94 Gbit/s

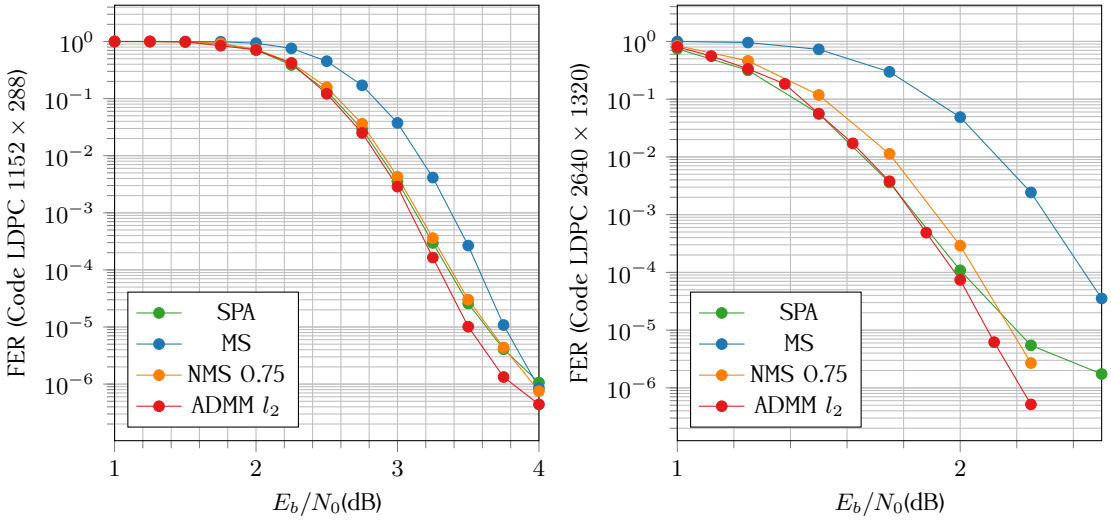


Figure 3.13 – Comparaison des performances de décodage des algorithmes ADMM-l_2, SPA, MS et NMS sur canal AWGN.

performances de décodage était un défi. Ce projet de recherche avec SUP'COM s'est concrétisé au travers de deux thèses menées à bien par Imen DEBBABI et Hayfa BEN THAMER, débutées en 2013 et 2014. J'ai co-encadré ces deux thèses avec Nadia KHOUJA et Fethi TLILI.

Au début des travaux, les activités de recherche dans le domaine du décodage des codes LDPC se focalisaient principalement sur l'implantation matérielle efficace de décodeurs en termes de débit [Mar15], de complexité [Cas11, Liu18] tout en limitant les dégradations de performances vis-à-vis de l'algorithme BP SPA [Mac97]. L'utilisation d'approches de décodage de type LP [Fel03, Fel05]. permettait d'égaler ou d'outrepasser les performances de décodage fournies par l'algorithme BP SPA [Fos99, Che05, Zha05] comme cela a été démontré dans [Liu16]. La formulation originale de cette approche de décodage qui repose sur la programmation linéaire (LP) [Fel03, Fel05] converge assurément sur le mot de code ML (*Maximum Likelihood*). Toutefois, les algorithmes de décodage LP usuels [Fel05, Liu12, Zha12b, Zha13b] possèdent des complexités calculatoires et mémoires trop importantes pour être applicables même pour des codes LDPC courts ($\mathbf{N} < 100$). Toutefois des avancées notables dans la formulation mathématique de l'algorithme LP ont changé la donne. L'utilisation de l'algorithme des directions alternées (*ADMM, Alternating Direction Method of Multipliers*) [Boy11] qui est un algorithme de résolution de problèmes d'optimisation permet de transposer le problème d'optimisation LP sous la forme d'une formulation basée sur une approche par passage de messages (MP) plus traditionnelle. La complexité calculatoire du décodage et les besoins en termes de mémorisation s'en trouvent drastiquement réduits, sans altération des performances de décodage. De plus, l'introduction de facteurs de pénalisation [Jia15] permet d'améliorer le pouvoir de correction des algorithmes de décodage LDPC usuels comme cela est illustré dans la figure 3.13.

Malgré ces avancées, la formulation algorithmique et la complexité des opérations flottantes impliquées dans l'algorithme de décodage rendaient son implantation impossible sous contrainte temps réel. En effet, même si la complexité algorithmique avec l'algorithme BP SPA semblait équivalente à haut niveau, l'opération de projection euclidienne impliquée dans le processus de décodage [Zha12b, Was15] possédait une complexité bien supérieure à celle de l'opération de

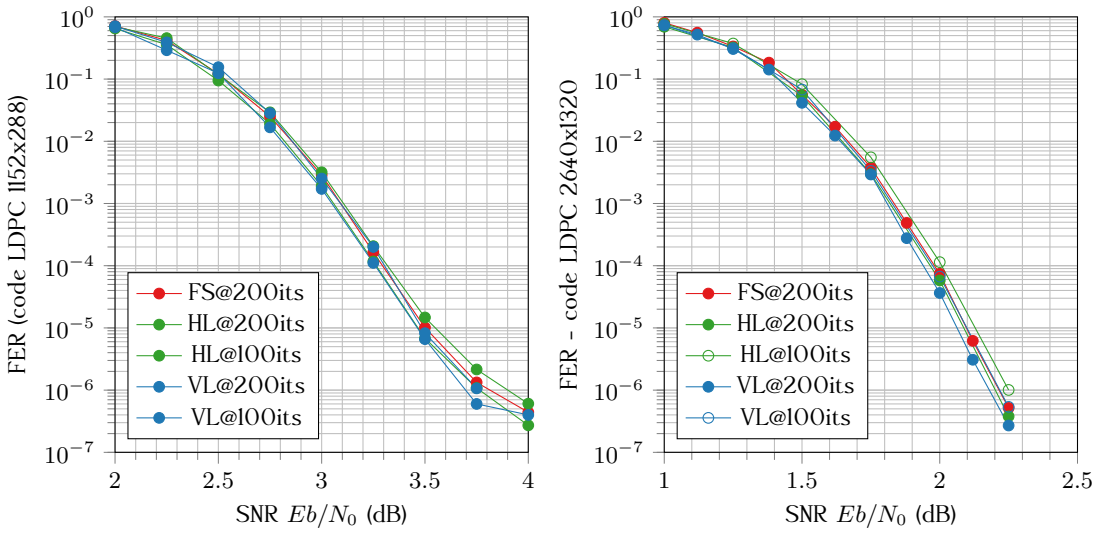


Figure 3.14 – Comparaison des performances de décodage de algorithme ADMM-l_2 en fonction de l'ordonnancement des calculs et du nombre d'itérations de décodage.

tangente hyperbolique inverse (*atanh*) présente dans la formulation BP SPA.

Afin d'améliorer les caractéristiques algorithmiques de la formulation du processus de décodage ADMM proposé dans [Boy11] basé sur un ordonnancement par innondation, nous avons étudié puis proposé plusieurs reformulations de l'algorithme de décodage. L'objectif consistait dans un premier temps à réduire le nombre de projections euclidiennes à calculer afin de réduire la complexité calculatoire. Ces reformulations se sont inspirées des travaux réalisés pour les décodeurs LDPC traditionnels [Hoc04, Gui07a, Sha07]. Deux reformulations de l'algorithme de décodage ADMM basées sur des ordonnancements par couche ont ainsi été développées et comparées [ci:Deb16c]. La figure 3.14 présente les performances de décodage de la formulation initiale par inondation (*FS, Flooding Schedule*) et des reformulations par couches horizontales (*HL, Horizontal Layered schedule*) et verticales (*VL, Vertical Layered schedule*). Nous avons constaté que :

1. le réordonnancement des calculs n'introduisait pas d'apparition de plancher d'erreurs ;
2. les ordonnancements par couche permettent de réduire d'environ 2× le nombre d'itérations de décodage et donc de calculs à iso-performance.

Nous avons décidé de choisir la formulation HL car elle minimisait à la fois le nombre de projections euclidienne (dont la complexité calculatoire est prédominante) et qu'il offrait un coût mémoire équivalent ce qui n'était pas le cas de la formulation VL comme détaillé dans [ri:Deb16a].

Nous avons ensuite entrepris de paralléliser l'implantation de cette formulation algorithmique sur des architectures programmables de type multi-cœurs et many-cœurs. Cela a été fait dans le but de faciliter l'étude des performances du décodeur ADMM dans différents contextes applicatifs. L'idée était aussi de simplifier les étapes de raffinement nécessaires pour aboutir à une implantation en virgule fixe du décodeur. La nature des traitements impliqués dans la projection euclidienne et la nature des données flottantes nous ont amenés à étudier différentes stratégies de parallélisation et à considérer aussi bien des cibles de type CPU et GPU pour

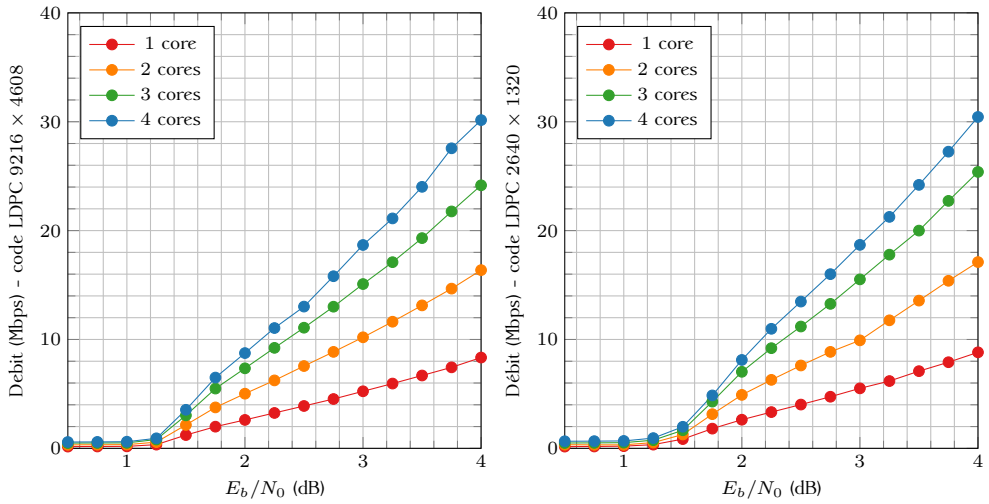


Figure 3.15 – Débits de décodage pour un décodeur ADMM-l_2 avec critère d'arrêt. Résultats mesurés sur un processeur INTEL multi-cœurs SIMD (4 cœurs) pour 100 itération au maximum [ri:Debl7a].

lesquelles nous étions déjà compétents. La complexité d'implantation était principalement liée au calcul de la projection euclidienne dont la complexité calculatoire était importante et difficile à formuler de manière parallèle. En effet, il est nécessaire d'exécuter des accumulations de données, des divisions, des normalisations ainsi que des opérations de tri pour des vecteurs de taille d_c. Différentes stratégies de parallélisation et d'optimisation de l'algorithme de décodage et de l'opération de projection euclidienne ont été étudiées. Les travaux ont permis d'atteindre des débits évoluant de quelques Mbps à plusieurs centaines de Mbps en fonction du SNR sur une cible multi-cœurs INTEL Core-i7 tel que présenté dans la figure 3.15. Des débits supérieurs à 600 Mbps ont été atteints sur un processeur INTEL Xeon possédant 24 cœurs physiques. Les schémas de parallélisation et les optimisations bas niveau appliqués pour les cibles de type CPU et GPU sont décrits de manière exhaustive dans [ci:Debl6a, ri:Debl6b, ri:Debl7a].

Nous avons ensuite employé ces implantations logicielles pour accélérer l'optimisation des paramètres de pénalisation [Jial5] de l'algorithme de décodage en fonction des codes LDPC étudiés et les SNR ciblés. Ces versions accélérées ont aussi servi à l'étude et à la validation des raffinements des formats de quantification internes afin de permettre la conception d'un décodeur matériel. L'identification de formats de quantification efficaces s'est avérée plus complexe que pour les algorithmes de type Min-Sum où seuls deux formats de quantification différents sont nécessaires : un format pour les messages et un format pour les accumulateurs. Dans l'algorithme ADMM, les dynamiques des données et des calculs se sont avérées importantes tout comme l'impact du bruit de quantification sur les performances de décodage. Si l'on exclut l'opération de projection euclidienne pour laquelle les formats de quantification internes proposés dans [Wasl5] ont été retenues, 3 formats de quantification différents ont été identifiés pour les données provenant du canal (accumulateurs), les messages issus des VNs → CNs et les messages CN → VNs. Pour maintenir les performances de décodage (figure 3.16), des largeurs respectives de, 12, 10 et 11 bits se sont avérées nécessaires.

Conscient de la forte complexité calculatoire de l'algorithme de décodage et dans le but de réutiliser la structure de décodeur LDPC à notre disposition [ri:LeGl4b], nous avons fait le choix

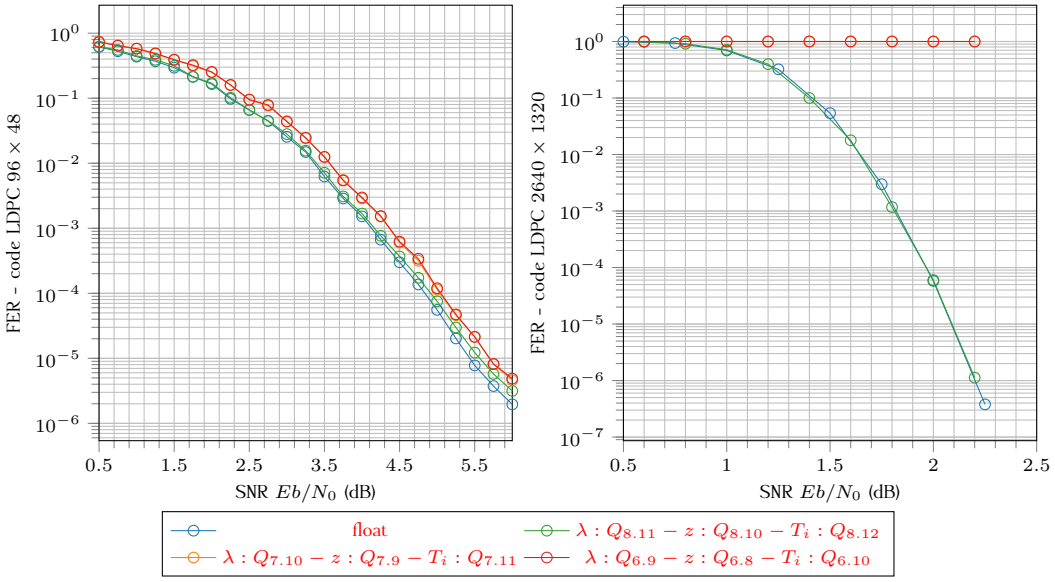


Figure 3.16 - Évaluation de l'impact des différents formats de quantification sur les performances de décodage - Décodeur ADMM-l_2 exécutant au maximum 200 itérations.

de concevoir un décodeur ADMM semi-parallèle. Une représentation schématique de l'architecture matérielle globale est fournie dans la figure 3.17. Le niveau de parallélisme $\mathbf{P}$ est paramétrable en fonction de la structure du code LDPC à traiter. Les unités de calcul, notées CNU_p, permettent de réaliser les traitements liés à l'algorithme de décodage : la génération des messages VN $\rightarrow$ CN, la projection euclidienne sur le polytope et la régénération des messages CN $\rightarrow$ VN et la mise à jour des accumulateurs. Ils sont aussi en charge de la mémorisation des messages CN $\rightarrow$ VN (Z et λ dans le cas de l'algorithme ADMM).

L'architecture interne de l'élément de calcul réalisant les traitements associés au nœud de parité est présentée dans la figure 3.18. Il est à noter que contrairement aux travaux antérieurs [Marl1a, Marl5, ci:Pigl8a] concernant les décodeurs de type MS ou Gallager-E, l'unité de calcul se compose de 3 tranches de pipeline. Chacune des tranches nécessite d_c cycles d'horloge pour traiter séquentiellement d_c données. La latence de calcul d'un nœud de parité est donc de $3 \times d_c$ cycles. La tranche de pipeline supplémentaire est dédiée au calcul de la projection euclidienne dont la complexité est ici prédominante.

À partir de la description VHDL complète de l'architecture du décodeur et de la méthodologie développée précédemment [ri:LeGl4b] pour gérer le séquencement des calculs et des accès aux mémoires, plusieurs prototypes sur circuit FPGA ont été développés et évalués. Le circuit FPGA ciblé était un Xilinx Virtex-6 (xc6vlx240t-3ff1156). Une synthèse des complexités matérielles ainsi que les débits atteints par les architectures développées pour quelques codes LDPC usuels est fournie dans le tableau 3.7. Les caractéristiques de décodeurs Min-Sum équivalents sont aussi fournies à titre de comparaison.

Les résultats d'implantations ont démontré qu'il est possible d'atteindre des débits de plusieurs Mbps lorsque les décodeurs ADMM exécutent 100 itérations de décodage et que 22 à 24 unités de calcul sont instanciées. Dans ce cas, le taux d'utilisation du circuit FPGA atteint jusqu'à 60% des *slices* pour le code LDPC avec $\mathbf{N} = 9216$. Ces données démontrent que les avancées faites

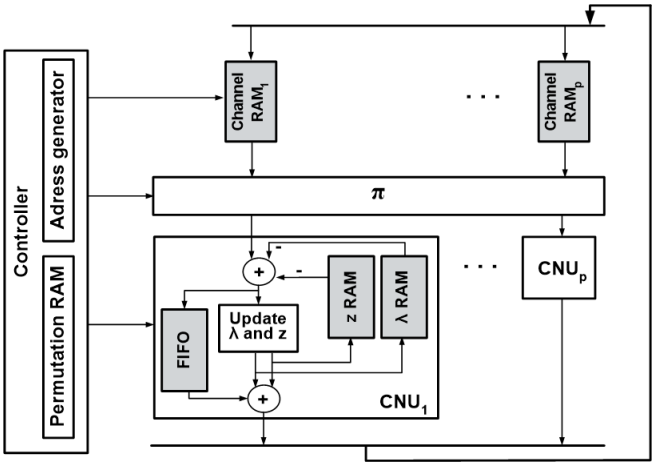


Figure 3.17 – Architecture interne du décodeur ADMM-l_2 implantant un décodage par couches horizontales.

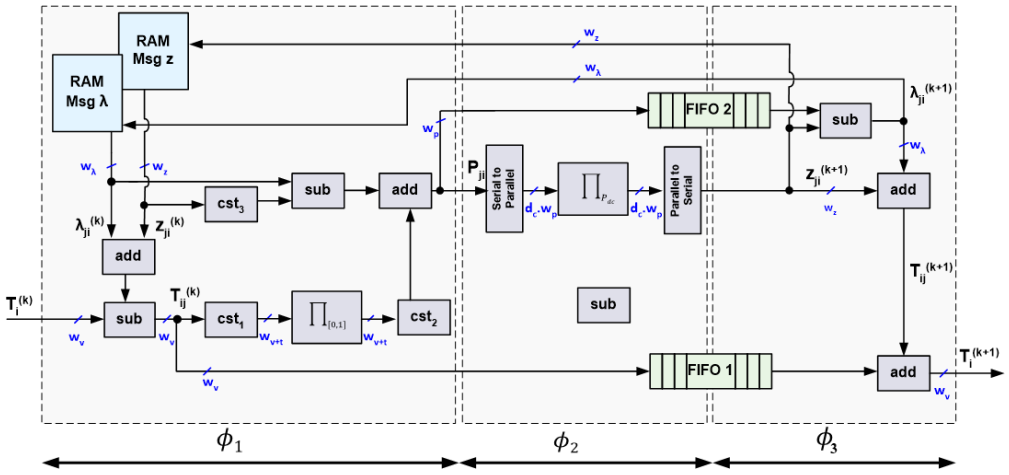


Figure 3.18 – Architecture interne d'un élément de calcul (CNU) du décodeur ADMM-l_2 réalisant un décodage par couches horizontales.

dans le domaine algorithmique ont rendu les algorithmes de décodage de type LP implantables. Cependant, la comparaison avec les décodeurs de type NMS montre qu'un travail supplémentaire est nécessaire pour atteindre des performances équivalentes en termes de débit et de complexité. En effet, les décodeurs de type MS sont pour l'instant 7× moins complexes et possèdent un débit 1.5× supérieur. Ces écarts sont liés à la différence de complexité des unités de calculs (PU). Le PU d'un décodeur MS nécessite 174 LUT (84 *slices*) tandis que celui d'un décodeur ADMM requiert 2213 LUT. (844 *slices*). De même le chemin critique dans l'unité de calcul est 1.5× plus important pour l'algorithme ADMM en raison du coût de la fonction de tri située dans le module de projection euclidienne. Malgré ces écarts de performance, il faut souligner que les décodeurs ADMM offrent un pouvoir de correction supérieur aux décodeurs de type MS.

Les travaux de recherche réalisés dans le cadre de la thèse d'Imen DEBBABI, soutenus le 7 mai 2017, ont fait l'objet de 3 publications dans des journaux internationaux avec comité de lecture [ri:Debl6a,

Table 3.7 - Résultats d'implantation post-placement et routage de décodeurs LDPC implantant les algorithmes ADMM-l_2 et NMS sur un circuit FPGA Xilinx Virtex-6 (ML605). Les débits sont mesurés pour 100 itérations de décodage.

Code LDPC	2640×1320		4000×2000		9216×4608	
Parallélisme (P/M_u)	22/22		22/22		24/32	
Decoder	ADMM-l_2	NMS	ADMM-l_2	NMS	ADMM-l_2	NMS
#LUT	48795 (32%)	7947 (5%)	48451 (32%)	7957 (5%)	67888 (45%)	11650 (12%)
#Flip-Flops	29119 (9%)	5635 (1%)	29149 (9%)	5637 (1%)	43310 (14%)	7960 (2%)
#blocs RAM18k	100 (12%)	78 (9%)	100 (12%)	78 (9%)	180 (21%)	116 (14%)
#DSP48	66 (8%)	– –	66 (8%)	– –	96 (12%)	– –
Fréquence (MHz)	92	157	81	157	84	151
Débit (Mbps)	11	19	5	10	8	13

ri:Deb16b, ri:Deb17a], 4 publications dans des conférences internationales [ci:Deb15, ci:Deb16a, ci:Deb16b, ci:Deb18a] et 2 publications dans des conférences nationales [cn:Deb17b, cn:Deb17c].

Comme l'utilisation du décodage LP pour des codes BCH [Weh16] ou des codes LDPC-NB [Lui14] était en cours d'investigation, nous avons pris le parti avec Nadia KHOUJA et Fethi TLILI d'étendre les travaux d'Imen DEBBABI aux codes LDPC convolutifs (LDPC-CC). Cette famille de codes était majoritairement utilisée dans les applications à haut débit (ex. les liens optiques). Il n'existait aucune étude démontrant ou infirmant la viabilité de l'approche de décodage LP basée sur l'optimisation ADMM pour cette famille de codes. En conséquence, nous avons exploré ce domaine dans les travaux de thèse débutés par Hayfa BEN THAMER en 2015.

Ces travaux ont porté tout d'abord sur la formulation mathématique du problème d'optimisation en vue de vérifier la viabilité de l'approche ADMM pour les codes convolutifs. Des simulations numériques couplées à une étude théorique de complexité algorithmique ont démontré que l'approche ADMM pouvait s'appliquer aux codes LDPC-CC. La formulation du problème LP utilisé dans l'algorithme ADMM ainsi que les premiers résultats en termes de performance de décodage ont été détaillés dans une publication en conférence internationale [ci:Ben17d]. L'étude s'est poursuivie par un raffinement de la formulation mathématique du problème en vue de la conception d'une architecture matérielle de décodeur. Nous avons proposé plusieurs stratégies d'ordonnancement des calculs lors du processus de décodage afin d'évaluer leurs impacts. Cela nous a permis d'identifier les stratégies ayant les meilleurs compromis entre la complexité calculatoire, la complexité mémoire et les performances de décodage. Ces travaux ont été valorisés au travers d'une publication dans une conférence internationale [ci:Ben17b]. En plus des traditionnels ordonnancements par couches horizontales et par couches verticales, un ordonnancement dynamique des calculs de parité a été également proposé. Ce dernier qui étend les travaux présentés dans [Jia17] identifie et priorise l'exécution des nœuds de parité en fonction de leur impact sur le décodage. Cette stratégie permet d'accélérer la convergence et d'éviter l'exécution de projections euclidienne « inutiles » réduisant de fait la complexité calculatoire d'un point de vue théorique. Ces travaux ont donné lieu à une communication dans la conférence

SOFTCOM [ci:Ben17c].

L'architecture matérielle des décodeurs LDPC-CC partageant peu de similitudes avec celles déployées pour les décodeurs LDPC tant du point de vue de l'accès aux mémoires et/ou de la parallélisation des calculs [ci:Ben17a], nous avons décidé d'utiliser les outils de synthèse d'architectures pour concevoir nos architectures de décodeur. Pour ce faire, nous avons exploité l'expertise développée dans la thèse de Yann DELOMIER. Ce choix permettant de réduire fortement le temps de conception vis-à-vis de descriptions au niveau RTL, nous avons pu porter un soin particulier à l'implantation sur circuit FPGA de l'opération de projection euclidienne pour laquelle de nouvelles formulations algorithmiques avaient été proposées [Jia17, Was15, Was17, Wei18]. En effet, comme la projection euclidienne représente, à elle seule, plus de la moitié de la complexité calculatoire du processus de décodage comme en atteste les travaux de thèse d'Imen DEBBABI, il était judicieux de les comparer. Ces travaux d'implantation nous ont permis d'identifier la formulation la plus efficace pour une implantation complète du décodeur. Ces travaux valorisés dans [ci:Ben18b] ont mis en évidence que les dernières contributions dans le domaine de la simplification du calcul de la projection euclidienne s'avéraient être les approches les moins efficaces une fois implantées [Jia17, Was17, Wei18]. À partir de ce constat, plusieurs décodeurs LDPC-CC basés sur l'algorithme ADMM ont été caractérisés et validés fonctionnellement sur circuit FPGA (DIGILENT Nexys 4) [ci:Ben18a]. Les résultats de l'étude démontrent que l'algorithme de décodage ADMM est efficace pour le décodage des codes LDPC-CC. Les niveaux de performance atteints sont obtenus avec une complexité matérielle $1.5\times$ fois supérieure et un débit réduit de moitié. Ces résultats ont été obtenus en comparant des architectures de décodeurs ADMM à des décodeurs NMS obtenus à l'aide de la même approche de conception. Ce résultat plus avantageux que ceux obtenus pour le décodage des codes LDPC dans la thèse Imen DEBABBI est principalement dû aux formats de quantification employés. En effet, ces derniers ont pu être réduits plus fortement, jusqu'à 6 bits (sans pénaliser les performances de décodage).

Les travaux de thèse d'Hayfa BEN THAMER que j'ai co-encadrés avec Nadia KHOUJA et Fethi TLILI ont été soutenus le 15 mars 2019. Les travaux scientifiques ont été valorisés à travers 7 publications dans des conférences internationales [ci:Ben16, ci:Ben17b, ci:Ben17d, ci:Ben17c, ci:Ben17a, ci:Ben18a, ci:Ben18b].

3.5 Étude et prototypage de récepteurs pour l'IoT

Les travaux réalisés sur la période 2010-2019 se sont focalisés sur l'implantation de décodeurs ECC. En effet, ces derniers représentaient une part prépondérante de la complexité des récepteurs. Afin d'élargir le spectre de mes compétences, j'ai participé en 2017 au montage du projet ANR QCSP porté par Emmanuel BOUTILLON, visant à améliorer l'efficacité énergétique de la couche physique des systèmes de communication ciblant l'IoT. Dans ce contexte, le protocole issu du projet de recherche se dispense de préambule de synchronisation des trames transmises, afin d'améliorer l'efficacité spectrale. L'approche est d'autant plus pertinente que le nombre de données à transmettre est faible ($\mathbf{N} < 100$).

Pour supprimer le préambule de synchronisation, l'idée consiste à l'utiliser un type de modulation de type CCSK (*Cyclic Code Shift Keying*) couplée à des ECCs non binaires. La modulation CCSK permet de superposer au sein d'une même trame à la fois des données d'information et de redondance ainsi que les mécanismes permettant la détection et la synchronisation des trames.

Cette nouvelle forme d'onde améliore l'efficacité énergétique des émetteurs au détriment d'une augmentation de la complexité calculatoire des récepteurs. Des études théoriques [Sai22] ont montré l'intérêt d'une telle forme d'onde. Cependant, le prototypage d'un récepteur temps réel incluant les étapes de synchronisation, d'égalisation et de décodage des trames nécessite une sélection et un raffinement des algorithmes en amont des étapes de conception matérielle. C'est autour de ces aspects que se situent mes contributions au projet ANR QCSP. Ces travaux ciblant l'implantation d'un récepteur matériel au sein d'un USRP x310 d'ETTUS sont effectués actuellement par Camille MONIERE, doctorant que je co-encadre depuis 2019 avec Emmanuel BOUTILLON.

À partir des travaux théoriques présentés dans [Sai22], nos travaux se sont portés dans un premier temps sur la modélisation du système de communications dans des langages de bas niveau. Cela a permis de générer l'architecture matérielle grâce à des outils de synthèse HLS. Ces travaux de raffinement algorithmique ont également permis de comparer différentes formulations algorithmiques du récepteur. Cette étude a mis en exergue l'intérêt, au premier abord contre intuitif, d'effectuer les corrélations nécessaires en réception dans le domaine temporel, plutôt que dans le domaine fréquentiel comme traditionnellement. En effet, cela permet de minimiser la complexité algorithmique et d'améliorer les performances de détection comme cela a été démontré dans [ci:Mon21a]. Ces travaux ont permis dans un second temps le développement d'un premier prototype de récepteur dans lequel l'étape de synchronisation était implantée de manière logicielle sur une architecture multi-cœurs. Les performances temps réel obtenues par ce prototype ont permis de démontrer la viabilité de l'approche dans des environnements réels. Le prototype temps-réel, incluant l'émetteur et le récepteur, développé a cette occasion a été présenté durant les journées "*Short packet transmission for wireless communications*" et "Implémentations, outils et applications émergentes pour la radio logicielle" organisées par le GdR ISIS. Les techniques de parallélisation et les optimisations mises en œuvre pour concevoir le récepteur ont donné lieu à une publication en conférence internationale [ci:Mon22a].

La modélisation et le raffinement en virgule fixe du récepteur ainsi que son implantation matérielle sur circuit FPGA via l'utilisation d'outils de synthèse HLS est en cours. Ces travaux permettront d'évaluer et de comparer l'approche proposée avec d'autres formes d'ondes dédiées à l'IoT telle que Lora [Che15, Ben20].

3.6 Bilan des travaux

Dans cette partie est résumé l'ensemble des encadrements réalisés dans le cadre des activités de recherche en relation avec la algorithmes et prototypage matériel de codes correcteurs d'erreurs. Il est à souligner que j'ai délibérément choisi de me focaliser sur un sous ensemble projets de recherche réalisés. En effet, ces derniers me semblaient avoir les constributions les plus notables.

Les encadrements de doctorants et d'étudiants de niveau master 2 dont les travaux sont associés à cet axe de recherche sont récapitulés dans le tableau 3.8. Une synthèse des valorisations scientifiques associées à cet axe de recherche est fournie ci-dessous :

— 5 articles dans des revues internationales à comité de lecture et **2** soumises ;
— 47 articles dans des conférences & workshops internationaux avec actes ;
— 9 articles dans des conférences ou workshops nationaux avec actes ;

Table 3.8 - Liste des encadrements scientifiques relatifs aux travaux relevants du thème *algorithmes et prototypage matériel de codes correcteurs d'erreurs*.

Nom	Statut	Période	Financement	Co-Encadrant(s)	% encadr.
Nadia KHOUJA	PhD	2007-2011	Tunisien	K. GRATI & A. GHAZEL	33%
Imen DEBBABI	PhD	2013-2017	Tunisien	F. TLILI & N. KHOUJA	33%
Thibaud TONNELLIER	PhD	2013-2017	Fonds propres	C. JEGO	33%
Hayfa BEN THAMER	PhD	2014-2019	Tunisien	F. TLILI & N. KHOUJA	33%
Vincent PIGNOLY	PhD	2017-2021	Fonds propres	C. JEGO	50%
Camille MONIERE	PhD	2019-...	ANR	E. BOUTILLON	50%
Antoine SIEBERT	PhD	2021-...	CIFRE	G. FERRE	50%
Oussama AIT SIDI ALI	PhD	2022-...	CIFRE	R. TAJAN	50%
Haïfa FARES	Master 2	2007	Fonds propres	D. DALLET	80%
Maha JEBALIA	Master 2	2008	Fonds propres	D. DALLET	80%
Zied JELASSI	Master 2	2009	Fonds propres	D. DALLET	50%
Mohammed ABID	Master 2	2010	Fonds propres	D. DALLET	50%
Walid SANAA	Master 2	2011	Fonds propres	D. DALLET	50%
Alexis AULERY	Master 2	2012	Fonds propres	D. DALLET	50%
Imen NAJAH	Master 2	2015	Fonds propres	N. KHOUJA	50%
Chaima BOUZOUITA	Master 2	2016	Fonds propres	N. KHOUJA	50%
Vincent PIGNOLY	Master 2	2017	GdR	–	100%
Florian LOUPIAS	Master 2	2020	ANR QCSP	–	100%
Oussama AIT SIDI ALI	Master 2	2021	ANR QCSP	–	100%

— 3 présentations lors de workshops nationaux ou journées du GdR.

L'ensemble des travaux présentés s'est déroulé dans le cadre d'activités partenariales ou de contrats de recherche publics. Les contrats de recherche ayant participé au financement de cet axe de recherche sont listés ci-dessous :

— **Thales-FEC** (Thales Aliena Space) (2013-2016), Étude et évaluation de nouvelles approches pour le décodage des turbocodes, Budget IMS : 50 K€ ;
— **R-S14** (R&T CNES) (2014-2015), Étude et comparaison de nouvelles techniques de codage/décodage canal, Budget IMS : 50 K€ ;
— **Turbo-CPM** (Thales Aliena Space) (2016-2017), Conception matérielle sur FPGA d'un récepteur itératif combinant des modulations à phase continue (CPM) à des codes LDPC, Budget IMS : 75 K€ ;
— **SatCom** (Airbus) (2016-2017), Étude et implémentation de schémas efficaces de réception pour des communications satellitaires, Budget IMS : 50 K€ ;
— **FEC descendant** (R&T CNES) (2018), Étude et comparaison de nouvelles techniques de codage/décodage canal pour les communications optiques descendantes dans le domaine spatial, Budget IMS : 40 K€
— **FEC montant** (R&T CNES) (2019), Étude et comparaison de nouvelles techniques de codage/décodage canal pour les communications optiques montantes dans le domaine spatial, Budget : 20 K€ ;
— **Quasi Cyclic Small Packet** (ANR QCSP) (2019-2022), Définition algorithmique, implantation matérielle et test d'un nouveau schéma de modulation pour l'IoT, Budget IMS : 65 K€ + ½ thèse ;
— **IA-4-demod** (Thales SIX) (2021-2024), Étude de l'intérêt des réseaux de neurones pour la démodulation de signaux numériques (Accompagnement de la thèse d'Antoine SIEBERT),

Budget IMS : 45 K€ ;
— **STC** (Safran) (2022-2025), Étude et mise au point de récepteurs STC optimisés sur plateformes virtuelles HPC (Accompagnement de la thèse d'Oussama AIT SIDI ALI, Budget IMS : 45 K€.

4 Bilan et perspectives de recherche

Ce manuscrit synthétise une partie de mes activités de recherche menées depuis le début de mon doctorat en Octobre 2002. Cette dernière partie dresse un bilan global de mes travaux puis fournit un bilan plus personnel. Enfin, des perspectives à court et à moyen terme sont données.

4.1 Bilan scientifique

4.1.1 Activités de recherche

Les systèmes électroniques embarqués ont pris une place prépondérante dans notre quotidien. Ces systèmes embarqués sont omniprésents dans nos vies, que ce soit dans nos téléphones, nos voitures, notre télévision ou même notre réfrigérateur. Cette omniprésence s'est accompagnée d'une augmentation notable de la complexité des fonctionnalités proposées et des contraintes d'intégration en matière d'efficacité énergétique, de puissance de calcul et de connectivité. Pour pouvoir répondre au mieux aux différents contextes applicatifs, les technologies d'intégration se sont diversifiées tandis que les niveaux d'abstraction s'élevaient. Ces avancées ont permis d'améliorer l'adéquation entre les besoins applicatifs et les supports d'implantation tout en maitrisant l'augmentation des temps de conception.

Historiquement, la conception de circuits dédiés en technologie ASIC était prévalente sur l'utilisation de processeurs ou de circuits de type FPGA. En effet, la faible capacité d'intégration des circuits FPGA et l'inefficacité énergétique des processeurs les discriminaient pour un bon nombre de systèmes embarqués. Toutefois, les solutions développées à partir d'architectures dédiées en technologie ASIC manquaient notablement de flexibilité et nécessitaient de longs cycles de conception.

Par la suite, au début des années 2010, la généralisation des architectures programmables de type HPC (multi-cœurs, many-cœurs) a changé la donne. En effet, ces dernières délivrent des puissances de calculs élevées en même temps que de la flexibilité et des cycles de développement courts. À titre d'exemple, aujourd'hui, les architectures multi-cœurs peuvent intégrer jusqu'à 56/96 cœurs physiques[1] tandis que des architectures many-cœurs dédiées peuvent intégrer jusqu'à 16384 unités

1. INTEL Xeon Platinum 9282 et AMD Zen 4 EPYC 9654.

de calcul en virgule flottante[2]. Ces solutions architecturales se sont naturellement substituées aux circuits ASICs dans de nombreux de domaines pas ou peu contraints énergétiquement. En effet, même si les architectures multi-cœurs et many-cœurs ont investi les datacenters et ont dominé les systèmes de calcul intensif dans les domaines du Big Data, du Deep Learning, des blockchains, ils ne possèdent toujours pas les caractéristiques énergétiques requises. Preuve en est, INTEL qui développe une stack 5G optimisée pour ses cœurs de processeur Xeon recommande d'utiliser un circuit FPGA pour l'encodage et le décodage des codes LDPC afin d'améliorer les caractéristiques du système. L'intégration de circuits FPGA dans les architectures en remplacement des GPU dans les clouds de Microsoft, Amazon ou Facebook pour ne citer qu'eux est liée au même constat.

Les circuits FPGA, longtemps cantonnés à des domaines spécifiques tels que le traitement du signal, la cryptographie, etc. s'imposent aujourd'hui dans de nombreux domaines où l'efficience énergétique, les performances et la flexibilité sont requises. Cela est dû à l'apparition de matrices à haute densité[3] possédant jusqu'à 3466080 ALM et 3456 DSPs qui permettent le développement de systèmes complexes. L'utilisation de circuits FPGA permet grâce à leur structure reconfigurable de tenir les contraintes temps réel en employant de faibles fréquences de fonctionnement minimisant *de facto* la consommation énergétique. Associé à la maturité des outils de synthèse HLS actuels [Catapult-HLS,Intel-HLS,Vitis-HLS] qui raccourcissent les temps de développement et ajoutent de la flexibilité. Ces outils permettent assez simplement le développement d'architectures à partir de descriptions algorithmiques de haut niveau (C, C++, OpenCL) proches de celles utilisées pour programmer les architectures many-cœurs. Ainsi les systèmes composés de processeurs multi-cœurs et de circuits FPGA communicant via des liens PCIe se multiplient [Caul6]. Dans les dernières générations de circuit FPGA, ces derniers intègrent même nativement des cœurs de processeurs ARM ou RISC-V. Cela permet de se passer d'un processeur hôte externe afin d'améliorer l'embarcabilité de ces solutions.

L'ensemble de ces solutions technologiques plus ou moins pertinentes en fonction des domaines et des contraintes applicatives est aujourd'hui à la disposition des concepteurs. Des méthodologies, des outils et des frameworks les ont rendus accessibles. Cependant, les concepteurs sont souvent moins enclins à faire évoluer leurs approches de conception que les solutions technologiques à progresser à cause, des coûts non récurrents de conception.

Les travaux décrits dans le premier chapitre de ce document détaillent mes activités de recherche dans le domaine des méthodologies de conception. Ces travaux visaient à permettre l'accélération du développement d'architectures numériques sur des technologies ASIC et FPGA. Différents travaux portant sur la mise au point d'algorithmes au sein d'outils de synthèse d'architectures (HLS) ont été menés afin d'améliorer la qualité des architectures RTL générées. Ces travaux ont été étendus dans un second temps afin de supporter des contraintes non fonctionnelles facilitant la vérification ou la protection des architectures RTL générées. L'émergence d'outils de synthèse matures au milieu des années 2010 m'a motivé pour étudier les performances obtenues à l'aide de tels outils et à travailler sur la modélisation efficace de décodeurs de codes correcteurs d'erreurs pour le standard de communication 5G 3GPP [5G]. Cela nous a permis de monter en compétences et de démontrer la viabilité de l'approche pour le prototypage rapide et efficace de systèmes de communications numériques.

Un autre élément important lors du développement d'un système est sa capacité à faire évoluer son comportement en fonction de l'environnement dans lequel il évolue. Pour assurer ce type de

2. NVIDIA GeForce RTX 4090.
3. FPGA INTEL Stratix 10 GX

fonctionnalité, les systèmes sont souvent composés d'un ou de plusieurs cœurs de processeurs qui fournissent à la fois flexibilité et efficacité. Dans le domaine de la conception et l'utilisation d'architectures programmables, nous avons présenté dans le Chapitre 2 nos travaux relatifs à la conception de cœurs processeurs flexibles pour les domaines de l'analyse protocolaire et du décodage des codes correcteurs d'erreurs. En parallèle de la conception de cœurs dédiés à un domaine applicatif, nous avons aussi étudié la capacité des architectures multi-cœurs (INTEL, ARM, RISC-V) et many-cœurs (NVIDIA) à pouvoir supporter efficacement l'exécution des algorithmes de décodage des ECC usuels en temps réel. Les niveaux de performance atteints par ces implantations, font encore référence. Ils ont donné lieu à de multiples valorisations industrielles.

Dans le domaine des algorithmes et prototypage matériel de codes correcteurs d'erreurs, nos travaux ont visé à développer de manière conjointe des algorithmes efficaces tout en prenant en considération les contraintes matérielles. Contrairement à mes travaux décrits dans les deux premiers chapitres pour lesquels les algorithmes à implanter étaient considérés comme figés, nous avons ici retravaillé et/ou proposé des algorithmes de décodage de codes ECC afin d'être capable dans un second temps de concevoir des architectures efficaces. Ces travaux réalisés en partie dans le cadre de partenariats industriels ont nécessité une montée en compétence dans le domaine algorithmique pour proposer des solutions architecturales innovantes. La qualité des résultats obtenus a motivé nos deux partenaires industriels à intégrer nos solutions dans leurs produits.

À travers ma participation à l'ensemble de ces travaux, j'ai eu la chance de pouvoir explorer un large spectre de problèmes scientifiques et de solutions technologiques dans le cadre de partenariats et avec des collègues aux compétences complémentaires.

4.1.2 Bilan personnel

Ce document récapitule 20 années de recherche, et plus particulièrement celles postérieures à ma prise de fonction de maître de conférences en 2006. Initiés au sein de l'équipe CSN du laboratoire IMS dirigé par Dominique DALLET. Ces travaux ont évolué lors du recrutement de Christophe JEGO en 2010. Cette évolution thématique au sein de l'équipe a été une expérience enrichissante. En effet, à partir d'une thématique scientifique plutôt bien identifiée, il nous a fallu trouver des financements pour nos activités. Au départ, ces travaux de recherche ont été initiés grâce à des contrats de transfert industriels permettant de couvrir tout ou partie des financements de thèse. Ensuite, le montage de projets collaboratifs et partenariats industriels au travers du rayonnement de l'équipe a fait de l'équipe ce qu'elle est aujourd'hui.

Les travaux développés au cours de ces années sont au croisement des compétences entre la conception de systèmes embarqués, les communications numériques, les architectures de calcul et les outils d'aide à la conception. Le but recherché était d'améliorer l'efficacité au sens large des systèmes développés en combinant l'ensemble de ces compétences. Pour valider la pertinence des travaux que nous avons menés, nous nous sommes basés sur des évaluations théoriques, mais nous avons autant que possible validé les performances par l'implémentation des techniques proposées sur des plateformes matérielles. Ce large spectre d'implantations sur des cibles allant du circuit FPGA basse consommation aux architectures multi-cœurs SIMD ou many-cœurs nous a permis de développer notre domaine d'expertise, de gagner en crédibilité et donc de renforcer et d'élargir nos partenariats afin de poursuivre nos activités.

La qualité des travaux proposés repose en grande partie sur le travail des doctorants et sur la visibilité de l'équipe CSN au niveau national. Ce rayonnement a permis d'attirer de bons étudiants et de collaborer avec de nombreux partenaires académiques et industriels comme cela est détaillé dans le document de synthèse. Une volonté naturelle pour les prochaines années est d'étendre cette visibilité au niveau international.

Pour conclure, l'emploi du temps d'un enseignant-chercheur devrait être réparti équitablement entre les tâches pédagogiques, les activités de recherche tout en includant des responsabilités administratives et collectives. Sur l'intégralité de ma carrière, un investissement conséquent dans toutes les composantes du métier a été réalisé. Cet équilibre, est selon moi, un point essentiel de ce métier épanouissant, mais parfois frustrant. En effet, le temps tout comme la santé est une denrée rare que l'on ne peut malheureusement pas acheter...

4.2 Perspectives de recherche

4.2.1 Activités de recherche

Mes travaux de recherche se focalisent actuellement sur la conception de solutions matérielles, logicielles ou conjointes pour l'implantation de fonctions employées dans les systèmes de communications numériques avec comme objectif l'efficacité calculatoire et énergétique. La démarche repose notamment sur la proposition ou la modification d'algorithmes et d'architectures matérielles en vue d'améliorer leurs adéquations.

Durant les dernières années, j'ai principalement travaillé à l'augmentation des débits de systèmes de communications terrestres ou satellitaires. L'émergence du standard 5G et l'explosion de l'IoT repositionnent au cœur des contraintes stratégiques l'efficience énergétique et la maitrise de la latence dans les systèmes de communications numériques. Ces problématiques modifient fortement les stratégies de parallélisation mise en œuvre pour les architectures programmables, mais aussi le choix des supports d'implantation avec l'utilisation de plateformes plus efficaces énergétiquement telles que les circuits FPGA ou les architectures conjointes de type AMD XILINX Zynq ou INTEL Xeon-FPGA Hybrid Chip. En complément de ces travaux visant à améliorer l'implantation ou le support d'exécution d'algorithmes, il est aussi possible d'œuvrer au niveau algorithmique. Pour ce faire, l'étude de nouvelles formes d'ondes couplées à l'usage parcimonieux de réseaux de neurones peut être une orientation possible pour de futurs travaux de recherche.

Si de nombreuses perspectives émanent directement de mes différents axes de recherche, cette partie détaille ceux qui devraient se développer durant les prochaines années.

4.2.2 Perspectives à court terme

L'intérêt pour les réseaux de neurones profonds s'est étendu du domaine de la classification à l'ensemble des domaines scientifiques. De nombreux travaux universitaires et industriels tentent de remplacer les algorithmes bien établis par des réseaux profonds. Cette tendance est aussi apparue dans le domaine des communications numériques ou les étapes de démodulation ou de décodage canal peuvent être effectuées par des réseaux de neurones au prix pour l'instant d'une complexité calculatoire souvent rédhibitoire. Toutefois, cette transformation permettrait à moyen terme de considérer l'ensemble d'un système de communications numériques sous forme

d'un unique réseau et de faciliter son optimisation et son implantation. Avec Guillaume FERRE nous avons été sollicités par Thales SIX pour étudier la possibilité de remplacer les fonctions d'égalisation et de démodulation utilisées actuellement dans leurs produits par des approches à base de réseaux de neurones. L'amorçage de ces travaux a eu lieu en octobre 2021 lors du recrutement d'Antoine SIEBERT en thèse. Nos objectifs consistent dans un premier temps à identifier et concevoir une structure de réseau de neurones pour réaliser l'étape d'égalisation. Ces résultats permettront dans un second temps de mettre en perspective les performances applicatives obtenues par rapport aux approches traditionnelles. Puis l'approche pourra être étendue à la fonction de démodulation et à leur traitement conjoint.

En parallèle de l'utilisation des réseaux de neurones profonds pour implanter des fonctions de communications numériques, j'ai débuté une action avec Safran Data Systems relative à la virtualisation de récepteurs numériques. Cette action de recherche fait suite au transfert industriel des décodeurs LDPC logiciels réalisé en 2021 pour des liens satellitaires haut débit (DVB-S2, CCSDS). L'objectif des travaux que nous avons débutés en 2022 est de proposer des stratégies de parallélisation et d'implantation pour les algorithmes de démodulation et de synchronisation définis dans la thèse Rami Othman [Oth19]. Le récepteur proposé dans [Oth19] cible des applications de télémesure aéronautique basées sur des modulations de fréquence. L'objectif de la thèse d'Oussama AIT SIDI ALI, que je co-encadre depuis Avril 2022 avec Romain TAJAN, est de proposer des stratégies de parallélisation et des évolutions algorithmiques permettant de transformer les récepteurs actuels utilisant des circuits FPGA vers une solution logicielle à base d'architectures multi-cœurs SIMD. Cette migration vise à maximiser l'agilité et la flexibilité des systèmes de télémesure tout en augmentant les débits grâce à un meilleur passage à l'échelle (Cloud-RAN).

Enfin, contrairement aux deux actions précédentes qui se focalisent sur la proposition de couples algorithmes / architectures efficaces pour le domaine des communications numériques, j'ai décidé de poursuivre mes travaux relatifs aux architectures de processeur. Les travaux actuellement en cours autour du processeur RISC-V se sont limités à enrichir son ISA afin d'augmenter son efficacité pour le décodage des codes ECC. Ces travaux vont se diversifier dans le cadre du projet ANR SEC-V qui a débuté en mars 2022. Dans le cadre de ce projet de recherche collaboratif, nous allons proposer une architecture de processeur RISC-V dotée d'une unité de transformation dynamique du code qui permettra au processeur de s'adapter à son environnement. La finalité première de ces travaux est de permettre une sécurisation du système en lui permettant de faire face à des attaques qu'il pourrait subir. Toutefois, cette adaptation de l'architecture pourrait être utile dans d'autres contextes applicatifs.

4.2.3 Perspectives à moyen terme

L'internet des objets a fait émerger le besoin de posséder des systèmes de communication pour lesquels le débit de transmission est une contrainte secondaire face à l'efficacité spectrale de la forme d'onde et la consommation d'énergie des émetteurs et récepteurs embarqués comme cela a été mis en exergue dans le projet ANR QCSP qui se termine. L'engouement pour l'IoT et les nouveaux usages qui en découlent ont mis en évidence l'impossibilité d'utiliser ces technologies dans les zones blanches. Pour palier à ce manque de relais, la solution envisagée par les acteurs académiques et industriels vise à mettre en place des constellations de nano ou picosatellites placés en orbite basse pour jouer le rôle de stations de base. Même si l'idée est attractive, la

transposition des technologies terrestres n'est pas évidente. Il est par exemple nécessaire : (1) au niveau algorithmique de trouver des formes d'ondes et des algorithmes pour compenser l'effet doppler, (2) de réduire drastiquement la complexité des algorithmes de réception pour rendre leur intégration sur des cibles spatialisables possibles, et (3) de développer des systèmes numériques efficaces et flexibles permettant le support l'exécution en temps réel des applicatifs, mais aussi leur évolution du système durant la vie du satellite. Pour financer ces travaux, la première option consisterait à monter un consortium afin de poursuivre les travaux entrepris dans le cadre du projet ANR QCSP. Une alternative pourrait être d'utiliser une collaboration industrielle comme par exemple EutelSat. En effet, depuis deux ans je travaille en collaboration avec Guillaume FERRE à la parallélisation et à la conception d'un récepteur Lora évolué temps réel [Ben20] sur cible programmable *low-power*, travaux qui intéressent actuellement cette entreprise.

Nos travaux de recherche relatifs à l'enrichissement des cœurs de processeurs RISC-V avec des instructions dédiées au décodage des ECCs vont se terminer. Ils démontrent l'intérêt de l'approche proposée à travers des gains substantiels en termes de latence et d'énergie consommée. Des gains supplémentaires sont obtenus lorsque la largeur du chemin de données (32 bits ou 64 bits) est efficacement utilisée pour traiter de manière vectorielle 4 ou 8 données représentées sur 8 bits. Ces évolutions architecturales ne sont cependant pas suffisantes pour rivaliser avec des processeurs ARM ou INTEL. En effet ces derniers possèdent des architectures superscalaires avec des unités SIMD dédiées. Ces unités possèdent tout d'abord leur propre file de registres. De plus, la représentation des données varie de 128 bits à 512 bits permettant de traiter de 16 à 64 données en parallèle. Pour concurrencer les architectures ARM/INTEL en termes de performances et d'efficacité énergétique pour des applications de type SdR ou Cloud-RAN, il est indispensable de complexifier l'architecture interne des cœurs RISC-V. De nombreuses possibilités architecturales existent : (1) intégrer une unité SIMD spécialisée telle que cela a pu être proposé récemment dans [Ara20] pour les données en virgule flottante, (2) proposer et intégrer un comportement VLIW ou superscalaire, (3) améliorer l'adéquation entre l'unité SIMD et les applicatifs à exécuter grâce à l'usage de la reconfiguration dynamique partielle et/ou l'utilisation d'eFGPA, et (4) développer les flots de compilation associés. Différentes approches sont actuellement disponibles pour décrire un cœur de processeur RISC-V et mener à bien ces travaux. Tout d'abord, il est possible pour maitriser finement l'architecture matérielle et son comportement temporel au cycle près, pour cela on une description au niveau RTL dans différents langages (VHDL, SystemVerilog ou Chisel [Bac12]) peut être privilégiée. A l'opposée pour favoriser la flexibilité et réduire le temps de conception, l'utilisation de descriptions comportementales couplée à l'utilisation d'outils de synthèse HLS peut être aussi mise en œuvre [Rok19, Man20]. Les travaux de recherche proposés, qui sont pluridisciplinaires, pourraient être financés via un projet ANR regroupant par exemple des membres des laboratoires INSA/INRIA, TIMA ou LIRMM.

Enfin, à long terme et pour prendre du recul vis-à-vis des travaux passés, il me semble pertinent de réfléchir à la possibilité de proposer une méthodologie de conception de niveau système pour les applications de télécommunications. L'objectif de cette approche serait d'explorer l'ensemble des solutions technologiques possibles : logicielles, matérielles et conjointes en fonction des contraintes applicatives. En effet, les nouveaux supports d'exécution de type AMD XILINX Zynq ou INTEL Xeon-FPGA Hybrid Chip permettent d'allier le meilleur des deux mondes afin de concevoir des systèmes moins énergivores. Toutefois les choix d'implantation, many-cœurs vs FPGA, sont laissés à l'utilisateur tout comme la sélection des composants à implanter. De nombreuses méthodologies associées ou non à des DSL permettant de spécifier les applicatifs

ont été proposées [Dar14, Sur19, Sha21] dans la littérature pour des systèmes de type flot de données. Cependant peu d'entre elles permettent de considérer l'espace des solutions tel qu'il s'offre à nous aujourd'hui : il existe aujourd'hui une infinité de déclinaisons pour l'implantation logicielle et/ou matérielle d'une fonctionnalité. Ce projet alliant le domaine des architectures pour les télécommunications, de la programmation HPC, de la synthèse HLS et des méthodologies / DSL nécessitera évidemment un consortium pluridisciplinaire national, voire international.

Parcours professionel

Partie II

5 Parcours Professionnel

5.1 Parcours universitaire et professionnel

5.1.1 Parcours universitaire

Décembre 2005

Doctorat en Sciences de l'Ingénieur - Option électronique
Université de Bretagne Sud (UBS), Lorient
Laboratoire LESTER (Laboratoire d'Electronique des Systèmes TEmps Réel)
Titre : « Contributions à la prise en compte des contraintes des applications TDSI dans la synthèse de haut niveau »

Thèse soutenue le 8 décembre 2005 devant le jury composé de :

E. Casseau	Maitre de Conférences HDR	Université de Bretagne Sud
A. M. Fouillart	Ingénieur de R&D	Thales
A. A. Jerraya	Directeur de Recherche CNRS	TIMA, INPG Grenoble
E. Martin	Professeur des Universités,	Université de Bretagne Sud
P. Quinton	Professeur des Universités	Université de Rennes 1
O. Sentieys	Professeur des Universités	Université de Rennes 1

Septembre 2002

DEA Électronique, Option conception et architecture des circuits intégrés,
Université de Bretagne Sud (UBS), Lorient.

Septembre 2002

DESS Mécatronique,
Université de Bretagne Sud (UBS), Lorient.

Septembre 2001

Maîtrise GEII (Génie Electronique et Informatique Industrielle),
IUP - Université de Bretagne Sud (UBS), Lorient.

Septembre 2000

Licence GEII (Génie Electronique et Informatique Industrielle),
IUP - Université de Bretagne Sud (UBS), Lorient.

Septembre 1999

DUT GIM (Génie Industriel et Maintenance),
IUT - Université de Bretagne Sud (UBS), Lorient.

5.1.2 Parcours professionnel

Septembre 2006 → ...

Maître de Conférences hors classe (PEDR classé A - 2012 → 2023),
Ecole ENSEIRB-MATMECA, Bordeaux INP,
Laboratoire IMS, UMR CNRS 5218,
Groupe : *Conception* - Equipe *Circuits et Systèmes Numériques* (CSN),
Thèmes de recherche : Adéquation Algorithme Architecture, HPC, ASIC/FPGA, méthodologies de conception, HPC, codes correcteurs d'erreurs.

Septembre 2005 → Aout 2006

Attaché Temporaire d'Enseignement et de Recherche (ATER),
Ecole d'ingénieurs ENSSAT à Lannion, Université de Rennes 1,
Laboratoire IRISA / INRIA, équipe R2D2,
Thèmes de recherche : Synthèse de haut-niveau, méthodologies de conception.

Octobre 2002 → Septembre 2005

Doctorant (financé par le projet ANR ALIPTA),
Université de Bretagne Sud (UBS), Lorient,
Laboratoire LESTER (laboratoire d'électronique des systèmes temps réel),
Thèmes de recherche : Synthèse de haut-niveau, méthodologies de conception.

Février 2002 → Septembre 2002

Stagiaire de DEA,
Université de Bretagne Sud (UBS), Lorient,
Laboratoire LESTER (laboratoire d'électronique des systèmes temps réel),
Titre : « Evaluation du langage de modélisation SystemC pour l'étude des performances d'un encodeur vidéo h263 ».

5.2 Activités d'enseignements

5.2.1 Activités passées (10/2002 → 08/2006)

Mes activités d'enseignement ont débuté lors de ma thèse en 2002. J'ai enseigné durant les trois années passée en thèse (2002-2005) avec le statut de vacataire au sein de l'Institut Universitaire Professionnalisé (IUP) de l'Université de Bretagne Sud sur le site de Lorient. Ces interventions

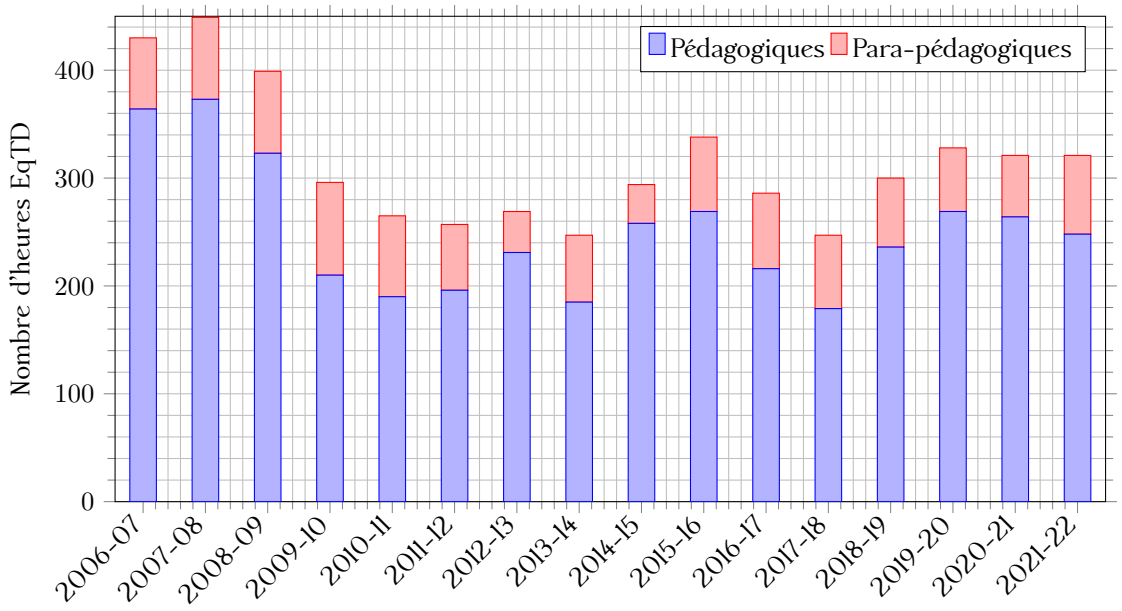


Figure 5.1 - Evolution de la charge d'enseignement exprimée en hEqTD par année.

se sont déroulées au sein des filières GEII (Génie Électrique et Informatique Industrielle) et GSI (Génie des Systèmes Industriels). Le volume horaire global pour l'ensemble de ces trois années représente environ 220 heures équivalent TD. Mes enseignements ont couvert des domaines thématiques variés allant de : l'algorithmie jusqu'aux systèmes de communications numériques en passant par la conception de base de données. Cette expérience riche, tant au niveau du contact avec les équipes pédagogiques qu'avec les étudiants m'a donné envie de devenir maître de conférences.

Suite à l'obtention de mon doctorat, j'ai occupé un poste d'ATER (Attaché Temporaire d'Enseignement et de Recherche) à l'ENSSAT à Lannion. Durant cette année, j'ai eu la charge un service d'enseignement de 192 heures équivalent TD. Ce dernier était réparti sur les 2ème et 3ème années des filières EII (Électronique et Informatique Industrielle) et LSI (Logiciel et Système Informatique). Durant cette période, j'ai eu l'opportunité d'intégrer une équipe pédagogique dynamique avec laquelle j'ai enseigné des modules relatifs à la conception de circuits et systèmes numériques, la programmation de systèmes temps réel et l'architecture des processeurs.

5.2.2 Enseignement à l'ENSEIRB-MATMECA (Sept. 2006 - ...)

Depuis Septembre 2006, j'officie en tant que maître de conférences à l'ENSEIRB-MATMECA au sein du département Électronique. L'ensemble des cours que j'assure se focalise autour de la conception des systèmes numériques et l'informatique industrielle. Les volumes d'enseignement sur la période 2006 à 2022 sont présentés en figure 5.1. Dans cette figure sont dissociées les heures para-pédagogiques (tutorat étudiants, responsabilités, suivi de stages et d'apprentis) des heures d'enseignement.

Dès ma première année au sein de l'ENSEIRB-MATMECA, j'ai été amené à effectuer un service d'enseignement important, associant participation à des séquences pédagogiques existantes, création

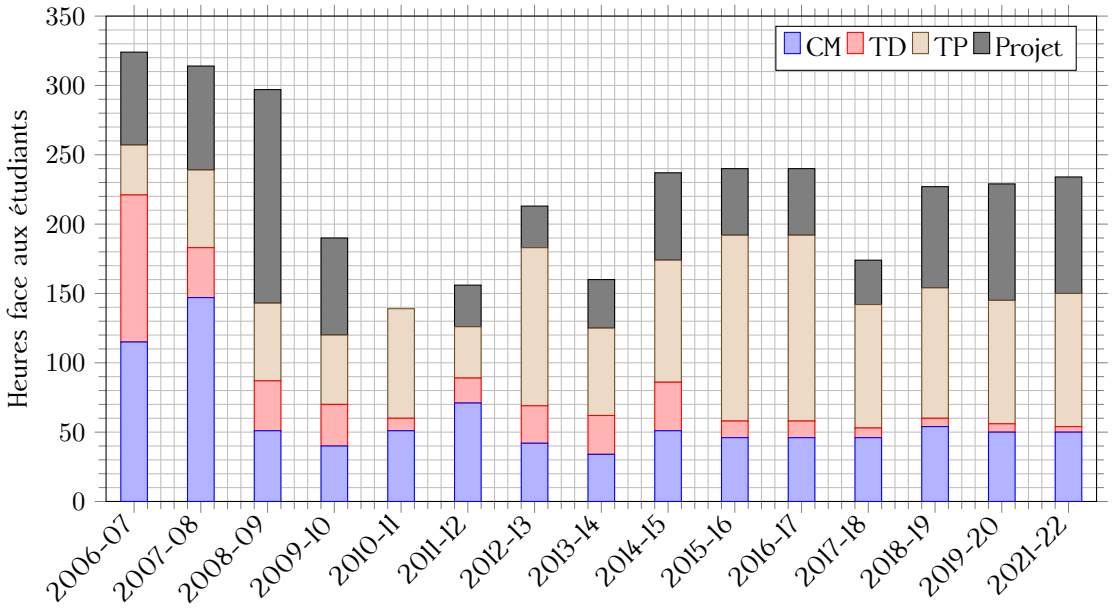


Figure 5.2 – Répartition du présentiel devant les étudiants (heures prodiguées).

de nouveaux enseignements et prise de responsabilités au sein de la filière par apprentissage en *Réseaux et Systèmes d'Informations* (RSI). Cet investissement important, sur la période de 2006 à 2010, consenti en plus de ma participation à la vie du département Électronique était rendu nécessaire par le fort déficit d'encadrement dans les domaines de la conception des systèmes numériques et de l'informatique embarquée. Depuis 2010, grâce au recrutement de nouveaux enseignants statutaires au sein de la filière Électronique de l'ENSEIRB-MATMECA (Christophe JEGO, Camille LEROUX et Jérémie CRENNE) pour soutenir les domaines de la conception des systèmes numériques et de l'informatique embarquée, mon service d'enseignement s'est finalement stabilisé autour de 240h de présentiel devant les étudiants.

La répartition de mon service en fonction de la nature de mes interventions est fournie dans la figure 5.2. Les volumes horaires indiqués dans cette figure sont exprimés en heures face aux étudiants. Il est à noter que l'investissement para-pédagogique (suivi des étudiants en stage ainsi que le suivi d'apprentis) n'est pas indiqué.

Actuellement, j'enseigne exclusivement au sein du département Electronique. J'interviens en 2ème et 3ème années. Mes interventions en 3ème année (niveau Master 2) sont positionnées dans l'option Systèmes Embarqués (SE) qui regroupe les étudiants souhaitant approfondir leurs compétences dans les domaines de la conception et l'utilisation des architectures numériques. La liste des enseignements que je prodiguerai lors de l'année 2022-2023 est fournie dans le tableau 5.1.

Je suis responsable de l'ensemble des modules que j'enseigne. Une partie des séquences pédagogiques que j'ai initiée depuis mon recrutement à l'ENSEIRB-MATMECA a été valorisée aux travers de 3 publications dans la conférence CETSI [cn:Bos08b, cn:LeG08, cn:LeG10] ainsi que 2 articles dans la revue J3EA [rn:LeG10, rn:LeG12].

Table 5.1 - Liste des enseignements préodigués lors de l'année universitaire 2022-2023 au département Electronique de l'ENSEIRB-MATMECA.

Code	Intitulé de l'enseignement	Niveau	Type	Volume (Eq. TD)
EN325a	Conception numérique avancée (modélisation, virgule fixe, HLS)	M2	CM TP	6h 24h
EN325b	Vérification de systèmes numériques (logiciel et matériel)	M2	CM TP	3h 12h
SE301	Calcul HPC pour les systèmes embarqués	M2	EI	32h
PR310	Projet avancé en Système embarqué	M2	Projet	2× 25h
EN201	Synthèse VHDL	M1	CM TD TP	6h 4h 9h
EN202	Projet de conception (VHDL)	M1	TP	30h
EN208	Technologies des circuits numériques	M1	TP	9h
PG108	Programmation en langage C++	M1	CM TD TP	18h 3h 12h
PR214	Projet thématiques	M1	Projet	23h

5.2.3 Responsabilités au sein de l'ENSEIRB-MATMECA

Depuis ma prise de fonction au sein de l'ENSEIRB-MATMECA je participe activement à la vie de l'école. Ainsi, j'ai pris des résponsabilités pédagogiques de manière continue. Dans cette partie, la liste des responsabilités pédagogiques et administratives sont rapportées.

Responsabilités pédagogiques

09/2006 → 08/2008

responsable pédagogique de la 2ème année de formation d'ingénieurs par apprentissage en « Réseaux et Systèmes d'Information ». Une promotion classique compte environ 24 étudiants. Les tâches associées à cette responsabilité couvrent : la sélection des candidats, la gestion des intervenants, les interactions avec le CFAI, la gestion de l'emploi du temps, le suivi des élèves et la gestion des jurys de 2ème année.

09/2010 → 08/2012

responsable pédagogique de la 2ème année de formation d'ingénieurs par apprentissage en « Systèmes Electroniques Embarqués ». Une promotion classique compte environ 24 étudiants. Les tâches associées à cette responsabilité couvrent : la sélection des candidats, la gestion des intervenants, les interactions avec le CFAI, la gestion de l'emploi du temps, le suivi des élèves et la gestion des jurys de 2ème année.

09/2013 → 08/2018

responsable pédagogique de la 2ème année de filière d'ingénieurs en « Électronique ». Durant une année normale, une promotion compte environ 96 étudiants. Les tâches associées à cette responsabilité couvrent : la gestion de l'emploi du temps, la gestion des intervenants, la proposition d'évolutions pédagogiques, le suivi des élèves et la gestion des jurys de 2ème année.

02/2018 → 08/2021

responsable des stages PFE (Projet à Finalité Entreprise) en 3ème année au département Électronique. Les tâches associées à cette responsabilité couvrent : le suivi des étudiants dans leur phase de recherche de stage, l'organisation du suivi des étudiants par l'équipe pédagogique du département, les relations avec les tuteurs industriels et la planification des soutenances.

09/2021 → ...

responsable pédagogique de la 2ème année de filière d'ingénieurs en « Électronique » et membre du Conseil des études de ce département. Durant une année normale, une promotion compte environ 96 étudiants. Les tâches associées à cette responsabilité couvrent : la gestion de l'emploi du temps, la gestion des intervenants, la proposition d'évolutions pédagogiques, le suivi des élèves et la gestion des jurys de 2ème année.

09/2021 → ...

responsable du parcours ingénieur-docteur pour la 2ème année de filière d'ingénieurs en « Électronique ». Cette responsabilité proposer aux étudiants inscrit en 2ème une initiation à la recherche académique et industrielle. Les tâches associées à cette responsabilité couvrent : l'encadrement d'une dizaine d'étudiants, la mise en relation des étudiants avec les EC des domaines choisis et l'organisation et la participation aux soutenances des études bibliographiques.

En parallèle de ces responsabilités pédagogiques, j'assure d'autres tâches d'animation au sein du département électronique. J'ai pour rôle de faire évoluer les thématiques dans lesquelles j'interviens :

09/2007 → ...

responsable de l'Unité d'enseignements EE9SEA depuis 2007, (*Architecture matérielle et conception conjointe*) dans l'option de 3ème année, spécialité Systèmes Embarqués à l'ENSEIRB-MATMECA. Cette UE est composée de 3 modules et représente 6 crédits ECTS.

09/2012 → ...

responsable de l'Unité d'enseignements EE7B (*Electronique Numérique*) pour le 7ème semestre de la filière Électronique à l'ENSEIRB-MATMECA. Cette UE est composée de 4 modules ou d'autres collegues interviennent. Cette UE représente 7 crédits ECTS.

09/2015 → ...

responsable du thème informatique au sein de la filière électronique pour les 3 années de formation. Cette responsabilité vise à maintenir la cohérence pédagogique de l'enseignement de l'informatique et proposer de manière concertée avec l'équipe pédagogique des évolutions. L'ensemble des enseignements concernés représentent environ 30 crédits ECTS.

Responsabilités administratives

En parallèle de ces responsabilités pédagogiques, j'assure des responsabilités administratives au sein de mon établissement. À ce titre, les responsabilités administratives que assure ou que j'ai assuré sont listées ci-dessous :

2008 → 2009

Membre du conseil de perfectionnement de l'ENSEIRB-MATMECA.

2018 → 2022

Membre élu du conseil d'école de l'ENSEIRB-MATMECA.

2022 → 2026

Membre élu du conseil d'école de l'ENSEIRB-MATMECA.

5.3 Thématiques de recherche

Depuis mon arrivée à l'ENSEIRB-MATMECA, j'effectue mon activité de recherche au sein du laboratoire IMS (UMR 5218). Au sein du laboratoire IMS, mes recherches se déroulent au sein de l'équipe Circuits et Systèmes Numériques (CSN) dirigée actuellement par Jean-Baptiste BEGUERET. Mes activités de recherche au sein de l'équipe CSN se sont concentrées autour de l'Adéquation Algorithme Architecture appliquée aux domaines des systèmes de communications numériques. Dans ce domaine applicatif l'augmentation continue de la complexité calculatoire des algorithmes et des applications à intégrer en temps réel nécessite une adéquation forte entre les algorithmes et les architectures numériques servant de support à leurs exécutions pour satisfaire contraintes d'intégrations qu'elles soient de latences ou énergétiques. Afin d'aboutir au développement de solutions efficaces, l'intégration de l'algorithme original est sur des architectures classiques souvent inopportun. En effet, ce dernier nécessite des adaptations ou des transformations plus ou moins importantes pour tirer pleinement partie des fonctionnalités offertes par les architectures numériques de type FPGA, CPU ou GPU. Il est aussi possible de repenser les architectures numériques d'implantation afin de s'adapter au mieux aux besoins applicatifs. Ces deux approches peuvent bénéficier de la conception de méthodes et des outils pour automatiser certaines des étapes de conception afin d'explorer l'espace des solutions et d'offrir de la flexibilité aux concepteurs.

Mes travaux de recherche ont débuté en octobre 2002 lors de ma thèse à l'Université de Bretagne Sud au sein du laboratoire LESTER. Durant cette dernière j'ai travaillé sur la conception et l'optimisation des algorithmes pour les outils de synthèse d'architectures. Ces algorithmes avaient pour vocation d'automatiser la génération d'architectures numériques sur cibles ASIC/FPGA à partir de descriptions comportementales. Depuis mon recrutement au laboratoire IMS en 2006, mes activités de recherche se sont diversifiées et couvrent maintenant d'autres thèmes de recherche. Comme présenté en introduction de ce mémoire, mes travaux de recherche s'articulent autour de 3 axes thématiques possédant tous pour dénominateur commun l'adéquation Algorithme Architecture :

1. La **conception automatisée d'architectures dédiées** - La conception d'architectures matérielles au niveau RTL est chronophage. Ce temps de conception important limite

l'exploration de l'espace des solutions architecturales ainsi que la capacité d'une solution à être déployée efficacement sur une autre cible technologique. Afin de palier à ce problème, j'ai travaillé sur le développement de méthodologies de conception et leurs enrichissements au travers du support de contraintes non fonctionnelles. Dans un second temps, j'ai évalué les capacités des outils de synthèse HLS industriels de dernière génération à supporter des cas d'usage réels dans les domaines du décodage de codes correcteurs d'erreurs et l'implantation de systèmes de trairement radar.

2. La **conception et l'utilisation d'architectures programmables** - Les systèmes embarqués actuels requièrent des niveaux de performances élevés tout en offrant de la flexibilité afin par exemple de s'adapter à différents contextes applicatifs. Afin de gérer cette problématique liée à la réutilisation et/ou l'adaptation des systèmes numériques, j'ai oeuvré à la conception d'architectures numériques programmables pouvant être enrichies (ASIP) en fonction des besoin applicatifs. Dans un second temps, j'ai tiré parti des nouvelles opportunités offertes par les architectures multi-cœurs SIMD et many-cœurs en proposant des stratégies de parallelisation innovantes et efficaces pour le domaine des systèmes communicants et plus particulièrement pour le décodage de codes correcteurs d'erreurs.

3. La définition d'**algorithmes et le prototypage matériel de codes correcteurs d'erreurs** - Afin d'aboutir à la conception de systèmes numériques efficaces pour un domaine applicatif donné, la prise en considération des contraintes architecturales est nécessaire au point de vue algorithmique et vice-versa. Le troisième thème de recherche traite donc de la définition d'algorithmes conjointement à la proposition d'architectures matérielles efficaces. Le domaine applicatif considéré lors de ces travaux se focalise sur le décodage des codes correcteurs d'erreurs.

Une grande partie de ma production scientifique et mon encadrement doctoral est directement associée à ces axes de recherche.

5.4 Production scientifique

Je ne présente dans cette section qu'une vue synthétique de mon activité de publication depuis le début de ma carrière. La liste détaillée est présentée à la fin de ce document dans le Chapitre *Publications personnelles*. De manière générale, je suis soit le premier auteur des articles, soit le second (ou le troisième) lorsque le premier auteur est un étudiant que j'ai co-encadré.

La figure 5.3 résume le nombre de publications par année et par type. Pour une meilleure lisibilité, seules les communications dans des journaux et des conférences internationales sont présentées. On notera une activité croissante avec un effort particulier sur les revues depuis 2011. Une synthèse des valorisations scientifiques de mes travaux est présentée ci-dessous :

— **30** articles publiés dans des revues internationales à comité de lecture,
— **92** communications dans des conférences internationales avec actes,
— **1** chapitre d'ouvrage avec comité de lecture,
— **4** communications invitées dans des conférences ou workshops internationaux,
— **24** communications dans des conférences nationales avec actes.

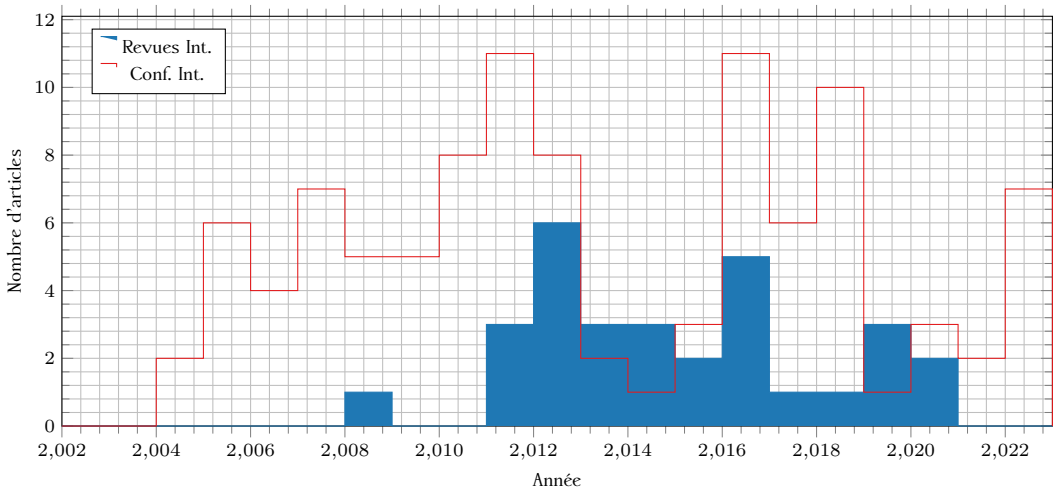


Figure 5.3 - Nombre de publications par année et par type de publications.

5.5 Encadrement doctoral et scientifique

5.5.1 Encadrement de doctorants

Depuis ma prise de fonction en 2006 au laboratoire IMS, au travers de mes activités au sein de l'équipe CSN, j'ai co-encadré **11** doctorants, participé à l'accueil de **2** doctorants invités. Je participe actuellement au co-encadrement de **9** doctorants. Ce chiffre élevé s'explique par le fait que **3** d'entre eux, impactés par la crise liée au COVID 19, finiront avec quelques mois de retard en 2023.

La répartition temporelle des co-encadrements de thèse depuis le début de ma carrière ainsi que le devenir des étudiants sont résumés dans figure 5.4. La liste complète des co-encadrements est, quant à elle, fournie ci-après par ordre chronologique.

Thèses soutenues (total : 11)

Nadia KHOUJA (2007 → 2011)

Optimisation de la chaîne de filtrage dans un récepteur multistandard en termes de consommation de puissance
Thèse soutenue le 26 Février 2011 (débutée en Janvier 2007)
Encadrement à 40% avec Khaled GRATI (40%) et Adel GHAZEL (20%)
Financement : Bourse de thèse tunisienne
Publications : [ri:Grall, ri:Gral2, ci:Kho08, ci:Kho09b, ci:Kho09a, ci:Khol0, ci:Khollb, ci:Kholla],
Situation post-thèse : Maître de conférences à l'ISET'COM (Tunis)

Moez KTHIRI (2009 → 2012)

Étude et implantation d'algorithmes de compression vidéo optimisés H.264/AVC dans un environnement conjoint matériel et logiciel
Thèse soutenue le 30 mars 2012 (débutée en Février 2009)
Encadrement à 40% avec P. Kadionik (40%) et H. Lévi (20%)
Financement : Fonds propres
Publications : [cn:Kthl10a, cn:Kthl10b, ci:Kthl11, ri:Kthl13],
Situation post-thèse : Maître de conférences à l'Université de Carthage (Tunisie)

Aurélien RIBON (2009 → 2012)

Amélioration du processus de vérification des architectures générées à l'aide d'outils de synthèse de haut-niveau
Thèse soutenue le 17 décembre 2012 (débutée en Octobre 2009)
Encadrement à (50%) avec D. Dallet (50%)
Financement : Bourse ministérielle
Publications : [pn:LeG09b, ci:Rib10, cn:Rib10a, cn:Rib10b, ci:LeG10b, ci:LeG10c, ci:Rib11, rn:LeG11],
Situation post-thèse : Ingénieur de recherche chez VASCO Data Security

Willy AUBRY (2008→2012)

Etude et mise en place d'une plateforme multiservices embarquée pour la gestion de flux multimédia et de capteurs réseaux à différents niveaux logiciels et matériels, vers la création d'un réseau domestique Multi-Play
Thèse soutenue le 19 décembre 2012 (débutée en Octobre 2008)
Encadrement à (40%) avec D. Negru (40%), F. Krief (10%), D. Dallet (10%)
Financement : CIFRE avec VioTech
Publications : [ci:Aub11, ci:Aub12a, ci:Aub12b, ci:Aub12c, ri:Duh13],
Situation post-thèse : Ingénieur de recherche (Viotech Communication, Paris)

Jigar SOLANSKI (2010 → 2014)

Approche générative conjointe logicielle - matérielle au développement du support protocolaire d'applications réseau
Thèse soutenue le 27 novembre 2014 (débutée en Octobre 2010)
Encadrement à (33%) avec L. Réveillère (33%) et D. Bromberg (33%)
Financement : Bourse CNRS handicap
Publications : [ri:Mer12, ci:Sol13, ri:LeG16a],
Situation post-thèse : Ingénieur de recherche (Cisco Systems, Bordeaux)

Imen DEBBABI (2013 → 2017)

LP decoding of LDPC codes : from algorithm to implementation
Thèse soutenue le 2 Mai 2017 (débutée en Octobre 2013)
Encadrement à (33%), N. Khouja (33%) et F. Tlili (33%)
Financement : Bourse tunisienne
Publications : [ci:Deb15, ci:Deb16a, ci:Deb16b, ri:Deb16a, ri:Deb16b, ri:Deb17a, cn:Deb17b, cn:Deb17c, ci:Deb18a],
Situation post-thèse : Ingénieur de recherche (Sagemcom Software & Technologies, Tunis)

Thibaud TONNELLIER (2013 → 2017)

Proposition de nouveaux algorithmes de décodage pour les turbocodes et évaluation des architectures matérielles d'implantation
Thèse soutenue le 5 Juillet 2017 (débutée en Novembre 2013)
Encadrement à (33%) avec C. Jégo (33%) et C. Leroux (33%)
Financement : Bordeaux-INP + fonds propres
Publications : [cn:Ton15a, cn:Ton15b, ci:Cas16b, ci:Ton16a, ci:Ton16b, ri:Ton16, ri:Cas19a],
Situation post-thèse : Post-doctorant à McGill (Canada)

Guillaume DELBERGUE (2015 → 2017)

Advances in SystemC/TLM Virtual Platforms : Configuration, Communication and Parallelism
Thèse soutenue le 18 Décembre 2017 (débutée en Février 2015)
Encadrement à (50%) avec C. Jégo (50%)
Financement : CIFRE avec GreenSoC
Publications : [ci:Del15, ci:Del16, cn:Cas17, cn:Del17, pi:Del15, pi:Del16a, pi:Del16b],
Situation post-thèse : Auto-entrepreneur (Hiventive, co-Founder and CEO)

Hayfa THAMEUR (2014 → 2019)

ADMM-LP decoding of LDPC convolutional codes : from algorithm to implementation
Thèse soutenue le 15 Mars 2019 (débutée en Octobre 2014)
Encadrement à (33%), N. Khouja (33%) et F. Tlili (33%)
Financement : Bourse tunisienne
Publications : [ci:Ben16, ci:Ben17b, ci:Ben17d, ci:Ben17c, ci:Ben17a, ci:Ben18a, ci:Ben18b],
Situation post-thèse : Post-doctorante à l'IMEN/UPHF (France)

Figure 5.4 – Répartition temporelle des co-encadrements de thèse depuis ma prise de fonction avec le devenir de l'étudiant apres son doctorat.

Yann DELOMIER (2016 → 2020)

Conception et prototypage de décodeurs de codes correcteurs d'erreurs à partir de modèles comportementaux
Thèse soutenue le 10 Juin 2020 (débutée en Octobre 2016)
Encadrement à (33%) avec J. Crenne (33%) et C. Jégo (33%)
Financement : Bourse ministérielle
Publications : [ci:Del18a, ci:Del18b, ci:Del18c, ci:Del19a, ri:Del20a, ri:Del20b, ci:LeG20a],
Situation post-thèse : ingénieur de recherche à Thales Aliena Space (Toulouse)

Vincent PIGNOLY (2017 → 2021)

Étude de codes LDPC pour applications spatiales optiques et conception des décodeurs associés
Thèse soutenue le 22 Janvier 2021 (débutée en Octobre 2017)
Encadrement à (50%) avec C. Jégo (50%)
Financement : Fonds propres
Publications : [ci:Pig18a, ri:Cas19a, ci:Pig20b, ci:Pig20a, ci:LeG22a, ri:LeG23a],
Situation post-thèse : Ingénieur de recherche à Thales Aliena Space (Toulouse)

Thèses en cours (total : 9)

Mael TOURRES (2019 → ...)

Conception automatisée de circuits numériques pour la couche physique des communications mobiles de 5ème génération,
Thèse débutée en Octobre 2019 (Université de Bretagne Sud),
Encadrement à 33% avec J. Crenne (25%), C. Chavet (25%) et P. Coussy (25%),
Financement : 50% Bordeaux-INP et 50% LabSTICC,
Publication(s) : [cn:Tou21a, cn:Tou21b, ci:Tou21a, ci:Tou22a, un:Tou22].

Camille MONIERE (2019 → ...)

Conception matérielle d'un démodulateur CCSK couplé à des décodeurs canal non-binaires,
Thèse débutée en Octobre 2019 (Université de Bretagne Sud),
Encadrement à 50% avec E. Boutillon (50%),
Financement : Projet ANR QCSP,
Publication(s) : [ci:Mon21a, cn:Mon22a, cn:Mon22b, ci:Mon22a, un:Mon22a].

Logan SAINT-GERMAIN (2020 → ...)

Méthodologie d'implantation de réseaux de neurones sur cible hétérogènes,
Thèse débutée en Janvier 2020,
Encadrement à 25% avec J. Crenne (25%), F. Baldacci (25%) et C. Jégo (25%),
Financement : CIFRE,
Publication(s) : [cn:StG22, ci:StG22a].

Hugues ALMORIN (2020 → ...)

Environnement de modélisation et de conception de systèmes logiciel/matériel pour les systèmes sur puce,
Thèse débutée en Octobre 2020,
Encadrement à 33% avec J. Crenne (33%) et C. Jégo (33%),
Financement : CIFRE,
Publication(s) : [cn:Alm22b, cn:Alm22a, ci:Alm22a].

Clémence GILLET (2020 → ...)

Optimisation de l'implémentation sur FPGA de réseaux de neurones événementiels pour le traitement de signaux biologiques,
Thèse débutée en Octobre 2020,
Encadrement à 33% avec A. Vincent (33%) et S. Saighi (33%),
Financement : Bourse ministérielle,
Publication(s) : [ci:Gil22a].

Denis SHEMONAEV (2020 → ...)

Définition d'un environnement de modélisation et de développement pour la plateforme matérielle NATvision : intégration de fonctions d'apprentissage pour le traitement vidéo,
Thèse débutée en Novembre 2020,
Encadrement à 33% avec J. Crenne (33%) et C. Jégo (33%),
Financement : CIFRE,
Publication(s) : [Ø].

Antoine SIEBERT (2021 → ...)

Contribution à l'implémentation d'algorithmes d'Intelligence Artificielle pour application de radiocommunications dans un contexte embarqué durci,
Thèse débutée en Octobre 2021,
Encadrement à 50% avec G. Ferre (50%),
Financement : CIFRE,
Publication(s) : [Ø].

Oussama AIT SIDI ALI (2022 → ...)

Virtualisation d'un récepteur de télémesure multimission,
Thèse débutée en Mars 2022,
Encadrement à 50% avec R. Tajan (50%),
Financement : CIFRE,
Publication(s) : [Ø].

Thomas NIEDDU (2022 → ...)

Sécurisation d'architecture par la gestion dynamique de microdécodage d'instructions,
Thèse débutée en Octobre 2022,
Encadrement à 50% avec S. Pillement (50%),
Financement : ANR-SEC-V,
Publication(s) : [ci:Pil23].

5.5.2 Encadrements de doctorants invités et de post-doctorants

Accueil et encadrements de doctorant invités (total : 2)

En plus des encadrements traditionnels, j'ai eu la chance de pouvoir accueillir et travailler avec deux doctorants invités. Ces encadrements ont été rendus possible grâce à deux bourses de mobilité que j'ai obtenues. Ces derniers sont listés ci-dessous :

Hassan HARB (2018)

Génération d'architectures matérielles pour le décodage de codes LDPC non binaires,
Période du 04/2018 au 06/2018,
Financement : GdR ISIS (SYMPA-DEC),
Encadré par Emmanuel BOUTILLON,
Publication(s): [ci:Har18a].

Belean BOGDAN (2010 → 2011)

Système de traitement des images génomiques fournies par une expérience micropuce,
Périodes du 01/2010 au 04/2010 et du 01/2011 au 04/2011,
Financement : Bourses de mobilité AUF (Agence Universitaire de la Francophonie),
Encadré par Monica Elena BORDA,
Publication(s): [ci:Bel11, ri:Bel12].

Encadrement de post-doctorants (total : 3)

Au travers des contrats de recherche auxquels j'ai participé, en plus des encadrements de doctorants et de stagaires, j'ai pu aussi co-encadrer 3 post-doctorants donc la liste est fournie ci-dessous :

Guillaume FERRE (2008 → 2009)

Étude et compensation des structures de conversion de données entrelacées,
Période (01/10/2008 au 30/06/2009),
Financement : TIADC (REI DGA),
Encadrement à 50% avec D. Dallet,
Publication(s): [ci:Fer08a, ci:Fer08b],
Situation actuelle : maître de conférences à l'ENSEIRB-MATMECA.

Goulven EYNARD (2009 → 2010)

Modélisation des performances des composants électroniques dans les chaînes de communication à très haut débit,
Période (01/01/2009 au 30/08/2010),
Financement : SCERNE (ANR),
Encadrement à 50% avec D. Dallet,
Publication(s): [ci:Eyn10],
Situation actuelle : ingénieur de recherche à la DGA.

Julien MERCADAL (2011 → 2012)

Adaptation d'un outil de génération d'automates logiciels en langage C pour cibler la génération d'architecture numérique sur FPGA,
Période (01/10/2011 au 30/08/2012),
Financement : laboratoire LaBRI,
Encadrement à 33% avec L. Réveillere et D. Broomberg,
Publication(s): [ri:Mer12],
Situation actuelle : maître de conférences à l'EISTI (École d'ingénieurs) à Cergy.

5.5.3 Encadrement d'étudiants de niveau master 1 & 2

En parallèle de mes activités d'encadrement doctoral, j'ai eu la possibilité d'encadrer des étudiants de niveau master 1 & 2. Ces derniers ont réalisé les développements nécessaires à mes activités de recherche et ont permis d'ouvrir (ou de fermer) de nouvelles voies. À l'heure actuelle, j'ai encadré **18** étudiants de niveau BAC+5 inscrits soit dans un master 2 "recherche" ou en école d'ingénieurs sur des contrats ANR ou sur fonds propres. La répartition temporelle des étudiants ainsi que leur devenir sont résumés dans la figure 5.4. Les informations détaillées concernant les encadrements sont récapitulés ci-dessous. Il est à noter que les **19** étudiants de niveau master 1 encadrés ne sont pas listés dans ce document.

Haïfa FARES (2007)

Étude de l'étage de décimation pour la numérisation d'un signal dans une chaîne de réception multistandard Wifi et WiMax, période du 01/01/2009 au 30/06/2010, établissement d'origine : SUP'COM Tunis, encadrement à 80% avec D. Dallet, publication(s): [ci:Far08], situation post-stage : doctorante à Télécom Bretagne (IMT Atlantique).

Maha JEBALIA (2008)

Conception et réalisation d'un filtre numérique décimateur pour un modulateur Sigma-Delta temps continu passe-bas pour un récepteur multistandard, période du 01/01/2008 au 30/06/2008, établissement d'origine : SUP'COM Tunis, encadrement à 80% avec D. Dallet, publication(s): [ci:Jeb08], situation post-stage : doctorante au laboratoire SUP'COM (Tunis).

Jérémie VIGUIER (2009)

Protection des circuits (IPs) multifonctionnalités à l'aide d'un système de gestion des droits d'utilisation, période du 01/02/2009 au 30/09/2009, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : ingénieur R&D.

Julien BEZINE (2009)

Passerelle matérielle/logicielle pour systèmes embarqués communicants hétérogènes, période du 01/02/2009 au 30/09/2009, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : doctorant au CEA (Paris).

Zied JELASSI (2010)

Étude, conception et implantation d'algorithmes de traitement du signal pour compenser les erreurs d'appariements de CAN entrelacés, période du 01/03/2009 au 30/05/2009, établissement d'origine : SUP'COM Tunis, encadrement à 50% avec D. Dallet, publication(s): [ci:Jell1a], situation post-stage : ingénieur R&D en Tunisie.

Mohammed ABID (2010)

Modélisation des performances des composants électroniques dans les chaînes de communication à très haut débit, période du 01/02/2010 au 30/07/2010, établissement d'origine : SUP'COM Tunis, encadrement à 50% avec D. Dallet, publication(s): [ci:Abi11, ri:Dell1], situation post-stage : ingénieur R&D en Tunisie.

Walid SANAA (2011)

Implantation d'une boucle cartésienne numérique pour la compensation des non linéarités des amplificateurs de puissance, période du 01/04/2011 au 31/12/2011, établissement d'origine : SUP'COM Tunis, encadrement à 50% avec D. Dallet, publication(s): [ci:San12a, ci:San12b, ci:Delt12], situation post-stage : ingénieur R&D en Tunisie.

Alexis AULERY (2012)

Implantation d'une boucle cartésienne numérique pour la compensation des non linéarités d'un Amplificateur de Puissance (PA), période du 01/03/2012 au 31/09/2012, établissement d'origine : ENSEIRB-MATMECA, encadrement à 50% avec D. Dallet, publication(s): [ci:Aul13, ri:Dell5], situation post-stage : doctorant au laboratoire LabSTICC (Lorient).

Guillaume DELBERGUE (2014)

Modélisation de haut-niveau en langage SystemC de calculateurs matériels pour le domaine de l'avionique, période du 01/03/2014 au 31/09/2014, établissement d'origine : ENSSAT (Lannion), encadrement à (30%) avec A. Tak (70%), situation post-stage : doctorant au laboratoire IMS.

Rima TABESSI (2014)

Implantation de filtres numériques multivoies et faibles débits sur FPGA pour des applications biomédicales, période du 01/03/2014 au 31/07/2014, établissement d'origine : ENISo (école d'ingénieurs de Sousse, Tunisie), encadrement à 50% avec Y. Bornat (50%), situation post-stage : ingénieur R&D chez IXEUS Group (Bordeaux).

Imen NAJAH (2015)

Implémentation d'algorithmes de décodage LDPC : Exploration de nouvelles techniques de décodage basées sur BP, période du 01/02/2015 au 31/07/2015, établissement d'origine : SUP'COM Tunis, encadrement à 50% avec N. Khouja (50%), situation post-stage : inconnue.

Chaima BOUZOUITA (2016)

Implantation matérielle sur FPGA de décodeurs LDPC convolutifs, période du 01/02/2016 au 31/12/2016, établissement d'origine : ENISo (école d'ingénieurs de Sousse, Tunisie), encadrement à 50% avec N. Khouja (50%), publication(s): [ci:Ben16], situation post-stage : ingénieur R&D.

Vincent PIGNOLY (2017)

Implantation matérielle sur cible FPGA de décodeurs LDPC non binaire à l'aide de l'outil Vivado HLS, période du 01/02/2017 au 31/07/2017, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : doctorant au laboratoire IMS.

Safouane NOUBIR (2018)

Conception d'architectures numériques sur FPGA pour l'analyse et la stimulation de cellules biologiques vivantes, période du 01/02/2018 au 31/07/2018, établissement d'origine : ENSEIRB-MATMECA, encadrement à 50% avec Y. Bornat (50%), publication(s): [ci:Nou18a, ci:Nou18b], situation post-stage : doctorant à Polytech'Nantes.

Maël TOURRES (2019)

Conception d'un cœur de processeur RISC-V reconfigurable dynamiquement, période du 01/02/2019 au 30/06/2019, établissement d'origine : Université de Nantes, encadrement à 50% avec J. Crenne (50%), situation post-stage : doctorant aux laboratoires IMS & LabSTICC.

Rémy FUMERON (2019)

Génération matérielle à base de modèles pour une application de correction de non linéarités dans les amplificateurs de puissance, période du 01/02/2019 au 31/07/2019, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : doctorant au laboratoire LabSTICC.

Florian LOUPIAS (2020)

Développement d'une plateforme SDR à base d'USRP pour la modulation CCSK, période du 03/02/2019 au 31/07/2019, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : ingénieur R&D.

Clémence GILLET (2020)

Implantation de réseaux de neurones événementiels sur circuit FPGA, période du 03/02/2020 au 31/07/2020, établissement d'origine : ENSEIRB-MATMECA, encadrement à 50% avec A. Vincent (50%), situation post-stage : doctorante au laboratoire IMS.

Oussama AIT SIDI ALI (2021)

Développement d'une plateforme SDR à base d'USRP pour la modulation CCSK, période du 01/02/2021 au 30/07/2021, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : doctorant au laboratoire IMS.

Salma IZOUGGARHAN (2021)

Evaluation des performances d'implantation d'algorithmes de codes correcteurs d'erreurs (LDPC) sur le multi-cœurs *"Kalray's MPPA"*, période du 01/02/2021 au 30/07/2021, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : master spécialisé en management de projets.

Thomas NIEDDU (2022)

Intégration de mécanismes de sécurité CISC dans un processeur RISC-V, période du 01/02/2022 au 30/07/2022, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, situation post-stage : doctorant au laboratoire IMS.

Léa VOLPIN (2022)

Conception et évaluation de systèmes d'émission Lora et QCSP pour l'IoT, période du 01/02/2022 au 30/07/2022, établissement d'origine : ENSEIRB-MATMECA, encadrement à 100%, publication(s): [ci:Vol22a], situation post-stage : ingénieure de recherche au laboratoire IMS.

5.5.4 Encadrement d'ingénieurs d'études

Dans le cadre des projets de recherche financés par des fonds publics ou dans le cadre de contrats de recherche industrielle auxquels j'ai participé, j'ai eu l'oppotunité de co-encadrer **3** ingénieurs d'études :

Simon DESFARGES (2009 → 2012)

Implantation de fonctions matérielles sur FPGA pour le transcodage de vidéos en temps réel
Encadrement à 50% avec D. Négru (50%),
Financement : projet ANR ARDMAHN,
Publication(s): [ci:Aub12a, ci:Aub12b, ci:Aub12c],
Situation post-contrat : ingénieur d'études chez Easymile SAS (Toulouse).

Adrien CASSAGNE (2015 → 2016)

Développement d'un simulateur rapide pour l'évaluation des performances des codes correcteurs d'erreurs
Encadrement à 50% avec D. Barthou (50%),
Financement : Labex CPU,
Publication(s): [ci:Cas15, ci:Cas16a, ci:Cas16b, ci:Cas17, ri:Cas19a],
Situation post-contrat : doctorant au laboratoire IMS.

Olivier HARTMANN (2015 → 2016)

Conception d'un système matériel pour le décodage itératif de codes LDPC couplés à une modulation CPM
Encadrement à 33% avec C. Leroux (33%) et C. Jégo (33%),
Financement : Fonds propres,
Publication(s): [ri:Cas19a],
Situation post-contrat : ingénieur d'études chez BeTomorrow (Bordeaux).

5.6 Rayonnement

5.6.1 Participation à des jurys de thèse

Depuis le début de ma carriere en 2006, j'ai participé à 18 jurys de thèse. Durant ces jurys, mon rôle a été, soit co-encadrant, soit examinateur. Parmis ces 18 jurys, 12 étaient relatifs aux doctorants que j'ai co-encadrés tandis que pour les 5 autres j'étais extérieur à l'équipe d'encadrement. Deux de ces 7 jurys de thèse se sont déroulés sur le site Bordelais, tandis que les 5 derniers se sont déroulés dans d'autres universités. Les jurys de thèse pour lesquels j'ai tenu le rôle d'examinateur sont indiqués ci-dessous.

Jurys de thèse de doctorants non co-encadrés sur le site bordelais (total : 2)

Maher JRIDI (2007)

Etude, Modélisation et Amélioration des Performances des Convertisseurs Analogique Numérique Entrelacés dans le Temps, thèse soutenue le 3 Décembre 2007 à Bordeaux.
Jury composé de : P. Garda (Président), A. Ghazel (Rapporteur), P. Loumeau (Rapporteur), D. Dallet (Directeur), M. Hourdequin (Industriel), Y. Berthoumieu (Examinateur), L. Bossuet (Examinateur), **B. Le Gal (Examinateur)**.

Nicolas MECHOUK (2010)

Etude et conception d'une structure BIST pour Convertisseur Analogique - Numérique, thèse soutenue le 26 Octobre 2010 à Bordeaux.
Jury composé de : C. Pellet (Président), C. Rebai (Rapporteur), P. Loumeau (Rapporteur), D. Dallet (Directeur), L. Bossuet (Examinateur), **B. Le Gal (Examinateur)**, G. Monnerie (Industriel).

Jurys de thèse de doctorants dans d'autres universités françaises (total : 5)

Brice COLOMBIER (2017)

Methods for protecting intellectual property of IP cores designers, thèse soutenue le 19 octobre 2017 à Saint-Etienne.
Jury composé de : L. Torres (Président), P. Coussy (Rapporteur), D. Hély (Rapporteur), L. Bossuet (Directeur), R. Chotin-Avot (Examinatrice), V. Fischer (Examinateur), T. Guneysu (Examinateur), **B. Le Gal (Examinateur)**.

Florian LEMAITRE (2019)

Tracking haute fréquence pour architectures SIMD : optimisation de la reconstruction LHCb, thèse soutenue le 13 Février 2019 à Paris.
Jury composé de : E. Chailloux (Président), A. Cohen (Rapporteur), D. Menard (Rapporteur), L. Lacassagne (Directeur), M. Gouiffes (Examinatrice), **B. Le Gal (Examinateur)**.

Hassan HARB (2019)

Conception d'un décodeur LDPC non binaire très haut débit, thèse soutenue le 8 Novembre 2018 à Lorient.
Jury composé de : O. Berder (Président), E. Boutillon (Directeur), A. Alaeddine (Co-Directeur), A. Burg (Rapporteur), Z. Zhang (Rapporteur), A. Al Ghouwayel (Examinateur), C. Douillard (Examinateur), L. Conde Canencia (Examinateur), **B. Le Gal (Examinateur)**.

Hannah BADIER (2021)

Transient Obfuscation for HLS Security - Application to Cloud security, Watermarking and Hardware Trojan Defense, thèse soutenue le 29 Avril 2021.
Jury composé de : R. Chotin (Rapporteur), L. Bossuet (Rapporteur), R. Leveugle (Examinateur), C. Pilato (Examinateur), **B. Le Gal (Examinateur)**, G. Gogniat (Directeur), J.-L. Le Lann (Directeur), P. Coussy (Co-Directeur).

Hanane Zhor BENMAGHNIA (2022)

Synthèse de Code Virgule Fixe Pour les Réseaux de Neurones, thèse soutenue le 7 Octobre 2022.
Jury composé de : M. El Habib Daho (Rapporteur), C. Michel (Rapporteur), C. Pilato (Examinateur), **B. Le Gal (Examinateur)**, M. Martel (Directeur), Y. Seladji (Co-directrice), X. Thirioux (Co-Encadrant).

5.6.2 Participation à des comités de suivi de thèse

Depuis la mise en place de la réforme du statut des doctorants datant de 2016, j'ai été sollicité pour participer aux comités de suivi de **4** étudiants extérieurs à l'université de Bordeaux :

Mohamed EL BOUAZZATI (2020-2023)

Low-power Software-Defined Baseband Processor for Flexibility and Security,
Directeur(s) de thèse : Guy GOGNIAT,
Ecole doctorale : LabSTICC, Université de Bretagne Sud.

Rémi FURMEON (2019-2022)

Etude des complémentarités des architectures reconfigurables gros grains et des codes correcteurs d'erreurs pour la cryptographie post-quantique,
Directeur(s) de thèse : Phillipe COUSSY,
Ecole doctorale : LabSTICC, Université de Bretagne Sud.

Vinh Hoang SON (2017-2020)

Design of next-generation Tb/s Turbo codes,
Directeur(s) de thèse : Emmanuel BOUTILLON et Charbel ABDEL NOUR,
Ecole doctorale : IMT Atlantique.

Petr DOBIAS (2017-2020)

Towards the Efficient Execution of Applications onto Multi-core Resilient Architectures,
Directeur(s) de thèse : Emmanuel CASSEAU,
Ecole doctorale : INRIA, Université de Rennes 1.

5.6.3 Participation à des comités de sélection

Depuis la création des comités de sélection, je participe régulièrement au processus de recrutement de maîtres de conférences et cela, quasi-exclusivement au sein d'établissements français en dehors du site bordelais. La liste des 10 comités de sélection auxquels j'ai participé depuis 2011 est fournie ci-dessous.

— (2011) - Poste n° 134, ENSEIRB-MATMECA (Electronique), IPB Bordeaux ;
— (2012) - Poste n° 884, IUT GEII, Université Paul Sabatier, Toulouse ;
— (2012) - Poste n° 1211, Département Electronique, Université de Nancy ;
— (2012) - Poste n° 1146, ENSSAT, Université de Rennes 1, Lannion ;
— (2013) - Poste n° 0275, IUT GEII, Université Paul Sabatier, Toulouse ;
— (2013) - Poste n° 1049, IUT MP, Université de Rennes 1, Lannion ;
— (2017) - Poste n° 1758, ENSSAT, Université de Rennes 1, Lannion ;
— (2018) - Poste n° 1843, ETN Polytech, Nantes ;
— (2021) - Poste n° 1644, Université de la Sorbonne, Paris ;
— (2022) - Poste n° 4069, ENSEA, Cergy.

Participation à des comités scientifiques

Participation aux comités d'organisation de workshop internationnaux

— ACM Workshop on Programming Models for SIMD/Vector Processing, WPMVP (2019) ;
— ACM Workshop on Programming Models for SIMD/Vector Processing, WPMVP (2020).

Participation à des comités d'organisation de conférences internationales

— IEEE International Conf. on Electronics Circuits and Systems, Special Session Chair (2011) ;
— IEEE International Conf. on Electronics Circuits and Systems, Special Session Chair (2018).

Participation à des comités techniques (TPC) de conférences internationales

— AICT The Advanced International Conference on Telecommunications (2013, 2014, 2015, 2016) ;
— IEEE International Conference on Electronics Circuits and Systems, ICECS (2006, 2018-2019) ;
— IEEE International New Circuits and Systems Conference (2011) ;
— IEEE GCC Conference and Exhibition for Sustainable Ubiquitous Technology (2011) ;
— IEEE International Symposium on Reconfigurable Communication-centric Systems-on-Chip, ReCoSoC (2012-2013) ;
— ACM Workshop on Programming Models for SIMD/Vector Processing, ACM WPMVP (2018-2019).

Participation à des comités techniques (TPC) de conférences nationales

— Conférence d'informatique en Parallélisme, Architecture et Système, COMPAS (2017-2019).

Relecture d'articles pour des revues internationales

Ci-dessous sont listés les journaux scientifiques d'audience internationale pour lesquels j'ai effectué au moins une relecture par année mentionnée :

— ACM Transaction on Embedded Computing (2015) ;
— China Communications (2018-2020) ;
— ELSEVIER, Microelectronics Journal (2012, 2017) ;
— ELSEVIER, Microprocessors and Microsystems (2020, 2021) ;
— ELSEVIER, Journal of Parallel and Distributed Computing (2017) ;
— ELSEVIER, Journal of Physical Communication (2019) ;
— EURASIP Journal on Advances in Signal Processing (2010) ;
— HINDAWI - International Journal of Reconfigurable Computing (2013) ;
— IEEE Access (2016-2020) ;
— IEEE Communication Letters (2015-2022) ;
— IEEE Embedded System Letters (2016) ;
— IEEE Journal of Solid-State Circuits (2013) ;
— IEEE Journal of Selected Topics in Applied Earth Observations and ... (2017-2018, 2020) ;
— IEEE Transactions on Circuits and Systems I : Regular Papers (2018) ;
— IEEE Transactions on Circuits and Systems II : Express Briefs (2021) ;
— IEEE Transactions on Communications (2020) ;
— IEEE Transactions on Computer-Aided Design (2015-2018) ;
— IEEE Transactions on Parallel and Distributed Systems (2020) ;
— IEEE Transactions on Signal Processing (2014-2015) ;
— IEEE Transactions on VLSI (2015-2017) ;
— IEEE Signal Processing Letters (2015) ;
— IET Computers & Digital Techniques (2013, 2015) ;
— IET Electronics Letters (2016) ;
— IET Information Security (2016-2017) ;
— Inderscience - Journal of High Performance Systems Architecture (2016) ;
— SPRINGER - Journal of Real-Time Image Processing (2012) ;
— SPRINGER - Journal of Signal Processing Systems (2017) ;
— World Scientific - Journal of Circuits, Systems, and Computers (2015).

Relecture d'articles pour des conférences internationales

En parallèle des relectures pour des revues scientifiques je suis sollicité régulièrement pour relire des articles soumis dans des conférences internationales. Ci-dessous est indiqué le nom des conférences pour lesquelles j'ai effectué au moins une relecture par année mentionnée.

— Advanced International Conference on Telecommunications, AICT (2013-2016) ;
— IEEE European Conference on Circuit Theory and Design, ECCTD (2013, 2015) ;
— IEEE Symposium on Reconf. Communication-centric Systems-on-Chip, ReCoSoC (2012-2013) ;
— IEEE Workshop on Signal Processing Systems, SIPS (2012, 2014, 2017-2020) ;

— IEEE International Conference on Design & Architectures for Signal and Image Processing, DASIP (2007, 2009, 2012-2014) ;
— IEEE GCC Conference and Exhibition, GCC (2010-2011) ;
— IEEE European Signal Processing Conference, EUSIPCO (2012),
— IEEE VLSI (2010) ;
— IEEE Computer Society Annual Symposium on VLSI, ISVLSI (2008, 2015, 2018) ;
— IEEE International New Circuits and Systems Conference, NEWCAS (2009, 2017-2019, 2021) ;
— IEEE Design, Automation and Test in Europe, DATE (2018, 2019) ;
— European conference on Parallel and distributed processing, Euro-Par (2009) ;
— IEEE International Conference on Electronics, Circuits, and Systems ICECS (2006-2020) ;
— Advanced International Conference on Telecommunications (AICT 2013-2016) ;
— ACM/IFIP/USENIX Middleware Conference, Middleware (2014) ;
— IEEE International Conference on Control, Measurement and Instrumentation, CMI (2016) ;
— IEEE Symposium on Turbo Codes & Iterative Information Processing, ISTC (2016, 2018) ;
— IEEE International Symposium on signal, Image, Video and Communications, ISIVC (2016).

Comité de lecture de conférences nationales

— Conférence d'informatique en parallélisme, architecture et système, COMPAS (2018, 2019),
— Colloque francophone de traitement du signal et des images, GRETSI (2013, 2017, 2022),
— Journées Nationales du Réseau Doctoral en Micro-nanoélectronique, JNRDM (2007),
— Ph.D Research in Microelectronics and Electronics conference, PRIME (2007).

Responsabilités collectives

Depuis mon recrutement, je me suis investi pour la communauté, localement dans la vie du laboratoire mais aussi au niveau national :

— Membre élu du conseil de laboratoire IMS de (2011-2016),
— Membre élu du CNU en 63ème section (2011-2015),
— Membre élu du conseil d'école de l'ENSEIRB-MATMECA (2018-2022),
— Membre élu du conseil scientifique du laboratoire IMS de (2021-2024).

Activité de vulgarisation scientifique

— Participation aux portes ouvertes de l'école ENSEIRB-MATMECA (2014-2019) ;
— Participation aux salons étudiants pour l'école ENSEIRB-MATMECA (2012-...) ;
— Entre 2014 et 2019, des séminaires de vulgarisation de la recherche ont été mis en place au sein du cursus ingénieurs de l'ENSEIRB-MATMECA durant la 2ème année du cycle d'études (niveau Master 1). L'objectif de ces séminaires était d'inciter les étudiants à intégrer des masters recherche en parallèle de leur 3ème année à l'école. Dans ce cadre, j'ai réalisé chaque année (2014-2019) un séminaire de vulgarisation d'une durée de 2 heures sur le thème de la recherche en électronique numérique.
— Entre 2011 et 2018, j'ai prodigué un séminaire relatif à mes travaux de recherche à l'ENSSAT à Lannion. Ce séminaire recherche d'une durée de 3 heures traitait de la modélisation des performances des systèmes numériques via SystemC. Il était dispensé aux étudiants de 3ème année de la filière Électronique ainsi qu'aux étudiants inscrits en Master recherche.

5.7 Collaborations scientifiques et projets industriels

Durant ma carrière, je me suis aussi impliqué dans des collaborations avec d'autres équipes universitaires et industrielles. Ces collaborations sont essentielles à l'enrichissement et l'approfondissement de réflexions scientifiques. Elles sont par ailleurs indispensables à l'obtention de financement pour les études que nous désirons mener. Ainsi, j'ai participé au montage, à l'animation scientifique et/ou à la réalisation de 10 projets de recherche. Ils peuvent être classifiés en deux types : des projets collaboratifs avec des organismes publics et des contrats de recherche bipartite avec un industriel. Je liste dans la suite de cette section les différents projets de recherche auxquels j'ai participé.

5.7.1 Contrats publics de recherche

ALIPTA (2002-2005)

Valorisation du rôle de la synthèse d'architectures dans les flots de développement systèmes
Type : projet RNTL,
Partenaires : TNI-Valiosys, Thales, Sacet, Turbo-Concept, Lester et Télécom Bretagne,
Budget : 140 K€,
Financement d'une thèse (B. Le Gal),
Implication : participant au projet.

TIADC (2007/2008)

Etude de technique d'entrelacement pour système d'acquisition numérique rapide et Ultra Large Bande
Type : REI DGA,
Partenaires : IMS, DGA,
Budget : 128 K€,
Financement d'un post-doc pendant 12 mois (G. Ferre),
Implication : participant au projet.

SCERNE (2008-2009)

Simulation de Chaînes d'Emission/Réception Nouvelle génération
Type : Programmes d'études amont (PEA DGA),
Partenaires : INRIA, IPSIS, XLIM, IMS, DGA, Thales,
Budget : 108 K€,
Financement d'un post-doctorant durant 18 mois (G. Eynard),
Implication : participant au projet.

CHAMELEON (2008-2010)

Passerelle reconfigurable multi-protocoles et multi-radios
Type : BQR (Bonus Qualité Recherche) ENSEIRB-MATMECA,
Partenaires : IMS, LaBRI,
Budget : 16 K€,
Financement de 2 masters recherche (J. Viguier, J. Bézine),
Implication : porteur du projet.

ARDMAHN (2009-2012)

Architecture Reconfigurable Dynamiquement et Méthodologie pour l'Auto-adaptation en Home Networking
Type : ANR ARPEGE,
Partenaires : LIEN, LEAT, LaBRI, IMS, Thomson, Viotech,
Budget : 50 K€,
Financement d'un ingénieur de recherche durant 12 mois (S. Desfarges),
Implication : responsable scientifique pour IMS.

FEC-CPU (2015-2016)

Accélération des simulateurs des systèmes de communications numériques
Type : LabEx CPU (Bordeaux),
Partenaires : IMS, INRIA,
Budget : 50 K€,
Financement d'un ingénieur de recherche durant 12 mois (A. Cassagne),
Implication : porteur de projet coté IMS.

SIMPADEC-NB (2016-2017)

Simulation et IMplantation PArallèle de DECodeur Non Binaire
Type : GdR ISIS (Appel à projets exploratoires),
Partenaires : IMS, LAB-STICC, Turbo-Concept,
Budget : 7 K€,
Financement d'un stage de niveau master 2,
Implication : porteur du projet.

NAND (2016-2019)

Noise Against Noise Decoder - Amélioration des performances des codes correcteurs d'erreurs par l'introduction de bruit de calcul
Type : ANR,
Partenaires : IMS, LAB-STICC, ETIS, Turbo-Concept, Thales,
Budget : 124 K€,
Financement d'un post-doctorant durant 18 mois,
Implication : participant au projet.

Quasi Cyclic Small Packet (QCSP) (2019-2022)

Improving the performance of error-correcting codes by introducing computational noise
Type : ANR,
Partenaires : IMS, LabSTICC, ETIS, Sequan, Orange,
Budget : 65 K€ + ½ thèse,
Financement d'une ½ thèse, de 6 mois d'ingénieur et de deux stages de master 2,
Implication : porteur du projet coté IMS.

SECURE-V (SECV) (2022-2025)

Increasing security in open-hardware approaches
Type : ANR,
Partenaires : IMS, IETR, LS2N, Thales (TRT), Thales (INVIA),
Budget : 156 K€,
Financement d'une thèse et de deux stages de master 2,
Implication : porteur du projet coté IMS.

5.7.2 Contrats privés de recherche

Cantest-X (2007-2009)

Développement d'un système d'étude et de caractérisation automatique des performances des Convertisseurs Analogiques Numériques
Type : activité de valorisation,
Partenaires : IMS, DGA (CELAR),
Budget : 50 K€,
Implication : responsable scientifique.

Cantest-XI (2010-2012)

Intégration de nouvelles approches de caractérisation & de fonctionnalités d'analyse en temps réel dans l'outil Cantest-X
Type : activité de valorisation,
Partenaires : IMS, DGA (CELAR),
Budget : 20 K€,
Implication : responsable scientifique.

Dect-Pattern (2013)

Développement d'un système de recherche de pattern en temps réel dans des signaux analogiques issus d'une mesure de courant
Type : activité de valorisation,
Partenaires : IMS, SERMA technologie (Pessac),
Budget : 50 K€,
Implication : responsable scientifique.

Thales-FEC (2013-2016)

Etude et évaluation de nouvelles approches pour le décodage des turbocodes
Type : contrat de recherche partenariale,
Partenaires : IMS, Thales Aliena Space,
Budget : 50 K€,
Financement de la thèse de T. Tonnellier (33%),
Implication : participant au projet.

R-S14 (2014-2015)

Etude et comparaison de nouvelles techniques de codage/décodage canal
Type : R&T CNES,
Partenaires : CNES, Labo IMS, TeSA, Thales Alenia Space,
Budget : 50 K€,
Implication : participant au projet.

ModCom (2015-2018)

Modélisation rapides et flexibles des communications dans les SoCs en SystemC
Type : contrat d'accompagnement CIFRE (G. DELBERGUE),
Partenaires : IMS, GreenSoCs,
Budget : 45 K€,
Financement d'une thèse CIFRE sur 3 ans (G. Delbergue),
Implication : responsable scientifique.

Turbo-CPM (2016-2017)

Conception matérielle sur FPGA d'un récepteur itératif combinant des modulations à phase continue (CPM) à des codes LDPC
Type : contrat de recherche partenariale,
Partenaires : IMS, Thales Aliena Space,
Budget : 75 K€,
Financement d'un ingénieur de recherche (O. Hartmann),
Implication : participant au projet.

PRISMA (2017)

Développement de décodeurs LDPC logiciels efficaces pour le standard 5G
Type : contrat de recherche partenariale,
Partenaires : IMS, Prisma technology,
Budget : 20 K€,
Implication : responsable scientifique.

SatCom - Airbus (2016-2017)

Étude de codes LDPC pour applications spatiales optiques et conception des décodeurs matériels associés
Type : contrat de recherche partenariale,
Partenaires : IMS, Airbus Space and Defense,
Budget : 50 K€,
Financement de la thèse de V. Pignoly (50%),
Implication : responsable scientifique.

R&T CNES [FEC descendant] (2018)

Etude et comparaison de nouvelles techniques de codage/décodage canal pour les communication optiques dans le domaine spatial
Type : R&T CNES,
Partenaires : IMS, CNES, Airbus Space and Defense,
Budget : 40 K€,
Financement d'un ingénieur de recherche,
Implication : participant au projet.

R&T CNES [FEC montant] (2019)

Etude et comparaison de nouvelles techniques de codage/décodage canal pour les communication optiques dans le domaine spatial
Type : R&T CNES,
Partenaires : IMS, CNES, Airbus Space and Defense,
Budget : 20 K€,
Implication : participant au projet.

Keysight technology (Keysight) (2019)

Développement de décodeurs LDPC et turbocodes à faible latence efficaces pour les standards 4G LTE et 5G 3GPP
Type : contrat de recherche partenariale,
Partenaires : IMS, Prisma Technology,
Budget : 40 K€,
Implication : porteur du projet.

KRATOS (2021)

Software LDPC decoder for spatial communications (DVB-S2, CCSDS)
Type : contrat de recherche partenariale,
Partenaires : IMS, KRATOS defense,
Budget : 40 K€,
Implication : porteur du projet.

SAFRAN (2021)

Software LDPC decoder for spatial communications (DVB-S2, CCSDS)
Type : contrat de recherche partenariale,
Partenaires : IMS, SAFRAN,
Budget : 30 K€,
Implication : porteur du projet.

Kalray (évaluation) (2021)

Évaluation des caractéristiques de la plateforme MMPA pour le décodage temps réel des codes LDPC du standard 5G 3GPP
Type : contrat de recherche partenariale,
Partenaires : IMS, Kalray,
Budget : 5 K€,
Implication : porteur du projet.

Vastech Technology (2022)

Software LDPC decoder for spatical communications (DVB-S2, CCSDS)
Type : contrat de recherche partenariale,
Partenaires : IMS, Vastech Technology,
Budget : 40 K€,
Implication : porteur du projet.

Kalray (parallélisation) (2022)

Proposition de stratégies de parallélisation efficaces dédiées à la plateforme MMPA en vue du décodage temps réel des codes LDPC du standard 5G 3GPP
Type : contrat de recherche partenariale,
Partenaires : IMS, Kalray,
Budget : 20 K€,
Implication : porteur du projet.

Neural networks for μC (2019-2022)

Définition d'une méthodologie d'optimisation et de déploiement de réseaux de neurones sur cibles microcontrôleurs
Type : contrat d'accompagnement CIFRE (L. SAINT GERMAIN),
Partenaires : IMS, CATIE,
Budget : 45 K€,
Implication : participant au projet.

Multi-view object tracking (2020-2023)

Étude et implantation d'un système de tracking d'objets dans un environnement multi-cameras
Type : contrat d'accompagnement CIFRE (D. SHEMONAEV),
Partenaires : IMS, EMG2,
Budget : 60 K€,
Implication : participant au projet.

HLS for radar processing (2020-2023)

Définition et mise en oeuvre d'un flot de conception pour l'implantation de chaînes de traitement radar à partir de modèles comportementaux
Type : contrat d'accompagnement CIFRE (H. ALMORIN),
Partenaires : IMS, ARELIS,
Budget : 30 K€,
Implication : porteur du projet.

IA-4-demod (2021-2024)

Étude de l'intérêt des réseaux de neurones pour la démodulation de signaux numériques
Type : contrat d'accompagnement CIFRE (A. SIEBERT),
Partenaires : IMS, Thales SIX,
Budget : 45 K€,
Implication : porteur du projet.

STC receiver design (2022-2025)

Étude et mise au point de récepteurs STC optimisés sur plateformes virtuelles HPC
Type : contrat d'accompagnement CIFRE (O. AIT SIDI ALI),
Partenaires : IMS, SAFRAN,
Budget : 45 K€,
Implication : porteur du projet.

5.7.3 Collaborations nationales et internationales

Depuis mon recrutement, je me suis investi afin d'initier et de faire fructifier des coopérations avec des équipes de recherche nationales ou étrangères. Ces efforts ont permis de lancer différentes actions de gré à gré ou au travers de projets structurants. Je détaille ici l'ensemble de ces éléments.

Réseau de recherche international

Durant la période de référence j'ai développé des coopérations internationales avec des équipes de recherche. Les collaborations détaillées ci-dessous sont celles qui ont engendrées des travaux de recherche qui ont été valorisés aux travers de multiples publications scientifiques dans des conférences ou des revues internationales.

— Une collaboration a été initiée et entretenue avec SUPCOM à Tunis. Cette collaboration scientifique débutée en 2010 a permis d'initier 3 thèses (Nadia KHOUJA, Imen DEBBABI, Hayfa THAMEUR) que j'ai co-encadrées ainsi que le co-encadrement de 7 masters recherche (Haifa FARES, Maha JEBALIA, Zied JELASSI, Mohamed ABID, Walid SANAA, Imen NAJAH et Chaima BOUZOUITA). Ces collaborations scientifiques avec les équipes du Pr Adel GHAZEL et du Pr Fethi TLILI ont été fructueuses et ont donné lieu à de multiples publications scientifiques.
— Une collaboration a été aussi initiée en 2010 avec l'université de Cluj-Napoca en Roumanie. Cette collaboration a perduré pendant deux ans grace à des fonds de l'agence universitaire de la francophonie (AUF). Cela a principalement permis de financer deux séjours de 3 mois à Belean BOGDAN.

Réseau de recherche national

D'un point de vue national j'ai initié des collaborations avec d'autres équipes de recherche universitaires. Ces collaborations nationales sont supportées majoritairement par des financements liés à des projets ANR ou sur fond propres. Ces différentes collaborations nationales ont principalement donné lieu aux co-encadrements de thèses et/ou de stages de niveaux master 2.

— Maël TOURRES : stage co-encadré avec Sébastien PILLEMENT (IETR, Nantes) ;
— Camille MONIERE : thèse co-encadrée avec Emmanuel BOUTILLON (LabSTICC, Lorient) ;
— Maël TOURRES : thèse co-encadrée avec Phillipe COUSSY (LabSTICC, Lorient) ;
— Thomas NIEDDU : thèse co-encadrée avec Sébastien PILLEMENT (IETR, Nantes) ;

En plus de ces collaboration de gré-à-gré, je participe aux réseaux de recherche nationaux.

Cette participation est attestée par les présentations réalisées durant des journées GdR ou des séminaires industriels organisés par exemple par le CNES :

— Design of a mixed-signal cartesian feedback loop for a low power Zero-IF WCDMA transmitter. Dans la Journée du GdR SOC/SIP, thématique AMS&RF, sur la correction numérique des imperfections de l'analogique, Télécom ParisTech, Paris, Mai 24 2012.
— Outil de synthèse VHDL à partir d'un langage de haut niveau. Dans le séminaire : "les architectures de traitement du signal embarquées pour le spatial" organisé par le Centre National d'Études Spatiales (CNES), Toulouse, France, Novembre 19 2009.
— Architecture ASIP flexible dédiée aux codes LDPC. Dans les Journées du BarCamp Reconfigurable (GDR SoC/SiP), Carry le Rouet, France, Octobre 2 2012.
— Architecture SoC/SoPC pour les applications réseaux nécessitant des fonctions de parsing. Dans la Journée thématique du GDR SoC-SIP - Nouveaux besoins et nouvelles approches en architecture de systèmes, Télécom ParisTech Paris, Novembre 18 2015.
— Progress in software implementations of ECC decoders. Dans la Journée thématique du GDR ISIS - Energy-efficiency in Error-Correction Coding, Télécom ParisTech, Paris, Juin 8 2016.
— Conception, implémentation et prototypage sur circuit FPGA de décodeurs LDPC pour les communications spatiales. Dans le séminaire : "Codes correcteurs d'erreurs pour les communications spatiales" organisé par le Centre National d'Études Spatiales (CNES), Toulouse, France, Juin 19 2019.
— Conception, implémentation et prototypage sur circuit FPGA de décodeurs LDPC pour les communications optiques spatiales. Dans le séminaire : "Codage, modulation et traitement du signal pour les communications optiques" organisé par le GdR ISIS, Paris, France, Septembre 19 2019.
— Evolution des techniques de parallélisation et des plateformes d'intégration pour la SDR. Dans la Journée thématique du GDR ISIS - Implémentations, outils et applications émergentes pour la radio logicielle, Sorbonne Université, LIP6, Paris, Avril 2022.

Réseau de recherche local

D'un point de vue local j'ai collaboré ou je collabore avec des collègues provenant d'autres laboratoire du site Bordelais afin de mener des activités de recherche transverses :

— Laurent REVEILLERE (Professeur des Universités, LaBRI) ;
— Daniel NEGRU (Maitre de conférences, LaBRI).
— Denis BARTHOU (Professeur des Universités, INRIA) ;

Cette volonté de transdisciplinarité est aussi un moteur pour des collaborations locales au sein du laboratoire IMS au niveau des applications bio-médicales avec les équipes dirigées par Yannick BORNAT et Sylvain SAIGHI ou sur des aspects plus algorihtmiques concernant les communications numériques avec Guillaume FERRE et Romain TAJAN.

Publications Personnelles

Manuscrits

[LeG02a] **B. Le Gal**. Evaluation des performances de SystemC pour la modélisation d'un encodeur h263. Master's thesis, Université de Bretagne Sud, Lorient, France, September 2002.

[LeG05a] **B. Le Gal**. *Contribution à la Prise en Compte des Contraintes des Applications TDSI dans la Synthèse de Haut Niveau*. PhD thesis, Université de Bretagne Sud, Lorient, France, 9 Décembre 2005.

Chapitre d'ouvrage avec comité de lecture

[ri :Del15] N. Deltimple, **B. Le Gal**, C. Rebai, N. Delaunay, A. Aulery, D. Belot, D. Dallet, and E. Kerhervé. *Linearization and Efficiency Enhancement Techniques for Silicon Power Amplifiers*, chapter 4 - Cartesian Feedback with Digital Enhancement for CMOS RF Transmitter, pages 35-53. Elsevier, 2015.

Revues internationales avec comité de lecture

[ri :Bel12] B. Belean, M. Borda, **B. Le Gal**, and R. Terebes. FPGA based system for automatic cDNA microarray image processing. *Computerized Medical Imaging and Graphics journal, Elsevier*, 36(5) :419-429, July 2012.

[ri :Cas11] E. Casseau and **B. Le Gal**. Design of multi-mode application-specific cores based on high-level synthesis. *Integration, the VLSI Journal, Elsevier*, 45(1) :9-21, January 2012.

[ri :Cas19a] A. Cassagne, O. Hartmann, M. Léonardon, C. Leroux, R. Tajan, O. Aumage, D. Barthou, T. Tonnellier, V. Pignoly, **B. Le Gal**, and C. Jégo. AFF3CT : A fast forward error correction toolbox ! *SoftwareX, Elsevier*, November 11 2019.

[ri :Deb16a] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Fast converging ADMM penalized algorithm for LDPC decoding. *IEEE Communication Letters*, 20(4) :648-651, April 2016.

[ri :Deb16b] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Real time LP decoding of LDPC codes for applications requiring high correction performances. *IEEE Wireless Communications Letters*, 5(6) :676-679, December 2016.

[ri :Debl7a] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Multi-core and many-core implementations of ADMM decoding algorithms for LDPC codes. *Journal of Signal Processing Systems (JSPS), Springer*, September 2017.

[ri :Dell1] N. Delaunay, M. Abid, **B. Le Gal**, D. Dallet, C. Rebai, N. Deltimple, E. Kerherve, and D. Belot. Mixed cartesian feedback for zero-IF WCDMA transmitter. *Journal of Analog Integrated Circuits and Signal Processing, Springer*, 73(3) :909–917, December 2012.

[ri :Del20a] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Model-based design of flexible and efficient LDPC decoders on FPGA devices. *Journal of Signal Processing Systems (JSPS), Springer*, 92 :727–745, February 2020.

[ri :Del20b] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Model-based design of hardware SC polar decoders for FPGAs. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 13(2) :27, June 2020.

[ri :Duh13] F. Duhem, F. Muller, W. Aubry, **B. Le Gal**, D. Negru, and P. Lorenzini. Design space exploration for partially reconfigurable architectures in real-time systems. *Journal of Systems Architecture (JSA), Elsevier*, 59(8) :571–581, September 2013.

[ri :Gral1] K. Grati, N. Khouja, **B. Le Gal**, and A. Ghazel. High-level design flow targeting real multistandard circuit designer requirements. *Journal of Communication and Computer, David Publishing*, 8(1) :335–346, May 2011.

[ri :Gral2] K. Grati, N. Khouja, **B. Le Gal**, and A. Ghazel. Power consumption models for decimation FIR filters in multistandard receivers. *VLSI Design journal (Special issue on Flexible Radio Design : Trends and Challenges in Digital Baseband Implementation), Hindawi*, 2012(Article ID 870546) :15, March 2012.

[ri :Kth13] M. Kthiri, **B. Le Gal**, P. Kadionik, and A. Ben Atitallah. A very-high throughput deblocking filter for h264/AVC standard. *Journal of Signal Processing Systems, Springer*, 73(2) :189–199, November 2013.

[ri :LeG08] **B. Le Gal**, E. Casseau, and S. Huet. Dynamic memory access management for high-performance DSP applications using High-Level Synthesis. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 16(11) :1454–1464, November 2008.

[ri :LeGl1a] **B. Le Gal** and Emmanuel Casseau. Latency-sensitive high-level synthesis for multiple word-length DSP design. *Journal on Advances in Signal Processing, EURASIP*, 2011(Article ID 927670) :11, January 2011.

[ri :LeGl1b] **B. Le Gal** and Emmanuel Casseau. Word-length aware DSP hardware design flow based on high-level synthesis. *Journal of Signal Processing Systems, Springer*, 62(3) :341–357, March 2011.

[ri :LeGl2] **B. Le Gal** and L. Bossuet. Automatic low-cost IP watermarking technique based on I/O mark insertions. *Design Automation for Embedded Systems, Springer*, 2012(16) :71–92, May 2012.

[ri :LeGl3] **B. Le Gal** and C. Jego. Softcore processor optimization according to real-application requirements. *IEEE Embedded Systems Letters*, 5(1) :4–7, March 2013.

[ri :LeGl4a] **B. Le Gal** and C. Jégo. GPU-like on-chip system for decoding LDPC codes. *ACM Transactions on Embedded Computing Systems (TECS)*, 13(4) :1–19, Feburary 2014.

[ri :LeGl4b] **B. Le Gal**, C. Jego, and C. Leroux. A flexible NISC-based LDPC decoder. *IEEE Transactions on Signal Processing (TSP)*, 62(10) :2469—2479, May 2014.

[ri :LeGl4c] **B. Le Gal**, C. Jégo, and J. Crenne. A high-throughput efficiency approach for GPU-based LDPC decoding. *IEEE Embedded System Letters*, 6(2) :29–32, June 2014.

[ri :LeGl5a] **B. Le Gal**, C. Leroux, and C. Jego. Multi-Gb/s software decoding of polar codes. *IEEE Transactions on Signal Processing (TSP)*, 63(2) :349-359, January 2015.

[ri :LeGl5c] **B. Le Gal** and C. Jego. High-throughput LDPC decoder on low-power embedded processors. *IEEE Communication Letters*, 19(11) :1861-1864, November 2015.

[ri :LeGl6a] **B. Le Gal** and C. Jego. High-throughput multi-core LDPC decoders based on x86 processor. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 27(5) :1373-1386, May 2016.

[ri :LeGl6b] **B. Le Gal**, L. Reveillere, Y.-D. Bromberg, and J. Solanki. A flexible SoC and its methodology for parser-based applications. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 10(1) :1-23, September 2016.

[ri :LeGl9a] **B. Le Gal**, C. Leroux, and C. Jego. High-performance software implementation of SCAN decoders for polar codes. *Annals of Telecommunications, Springer*, 73(5) :401-412, April 2018.

[ri :LeGl9b] **B. Le Gal** and C. Jego. High-throughput FFT-SPA decoder implementation for non-binary LDPC codes on mulitcore target. *Journal of Signal Processing Systems, Springer*, 92 :37-53, March 2019.

[ri :LeGl9c] **B. Le Gal** and C. Jego. Low-latency and high-throughput software turbo-decoders on multi-core architectures. *Annals of Telecommunications, Springer*, 75 :27-42, August 2019.

[ri :LeG23a] **B. Le Gal**, V. Pignoly, and C. Jégo. High-performance hard-input LDPC decoding on multicore devices for optical communication links. *Journal of System Architecture (JSA), ELSEVIER*, February 2023.

[ri :Mer12] J. Mercadal, L. Reveillere, Y.-D. Bromberg, **B. Le Gal**, T. F. Bissyande, and J. Solanki. Zebra : Building efficient network message parsers for embedded systems. *IEEE Embedded Systems Letters*, 4(3) :69-72, September 2012.

[ri :Ton16] T. Tonnellier, C. Leroux, **B. Le Gal**, C. Jego, B. Gadat, and N. Van Wambeke. Lowering the error floor of turbo codes with CRC verification. *IEEE Wireless Communications Letters*, 5(4) :404-407, August 2016.

Articles dans des revues nationales

[rn :LeG08] **B. Le Gal**, E. Casseau, and C. Andriamisaina. Synthèse de haut niveau en tenant compte de la dynamique des traitements. *Revue des Technique et Science Informatiques, série TSI, Architecture des ordinateurs,*, 27(9-10/2008) :1129-1154, Décembre 2008.

[rn :LeGl0] **B. Le Gal**, L. Bossuet, and M. Grand. Enseignement ludique de la programmation objets à l'aide d'applications de traitement de l'image. *Journal sur l'enseignement des sciences et technologies de l'information et des systèmes (J3EA)*, 9 :6, 2010.

[rn :LeGl1] **B. Le Gal**, A. Ribon, C. Jego, and D. Dallet. La synthèse de haut-niveau, un atout indéniable pour la construction de systèmes complexes ? *Journal l'Actualité Composants, Edité par le Centre Nationnal d'Etudes Spatiales (CNES)*, 36 :4-11, Juillet 2011.

[rn :LeGl2] **B. Le Gal** and D. Dallet. Sensibilisation au développement d'accélérateurs matériels sur FPGA. *Revue sur l'enseignement des sciences et technologies de l'information et des systèmes (J3EA)*, 11(3) :24, Septembre 2012.

[rn :LeGl3] **B. Le Gal** and C. Jégo. Méthodologie d'optimisation des processeurs embarqués : une approche favorisant la réduction de la surface et de la consommation des processeurs embarqués. *Revue des Technique et Science Informatiques (TSI), Architecture des ordinateurs, Architecture des ordinateurs*, 32(6) :725-754, Septembre 2013.

Articles dans des conférences internationales

[ci :Abd05] N. Abdelli, P. Bomel, E. Casseau, A.M. Fouilliart, C. Jégo, P. Kajfasz, **B. Le Gal**, N. Le Héno, and E. Martin. Hardware virtual components compliant with communication system standards. In *Proceedings of the 8th Euromicro Conference on Digital System Design (DSD)*, pages 88-95, Porto, Portugal, August 30 - September 3 2005.

[ci :Abi11] M. Abid, N. Delaunay, **B. Le Gal**, D. Dallet, C. Rebai, N. Deltimple, E. Kerherve, and D. Belot. Mixed cartesian feedback for zero-IF WCDMA transmitter. In *Proceedings of the 2nd IEEE Latin American Symposium on Circuits and Systems (LASCAS)*, pages 1-4, Bogota, Colombia, February 23-25 2011.

[ci :Alm22a] H. Almorin, **B. Le Gal**, J. Crenne, and C. Jego. High-throughput FFT architectures using HLS tools. In *Proceedings of the IEEE International Conference on Circuits and Systems (ICECS)*, pages P-P, Glasgow, UK, October 24-26 2022.

[ci :And06a] C. Andriamisaina, **B. Le Gal**, and E. Casseau. Bit-width optimizations for high-level synthesis of digital signal processing systems. In *Proceedings of IEEE Workshop on Signal Processing Systems (SIPS)*, Banff Alberta, October 2-4 2006.

[ci :And06b] C. Andriamisaina, **B. Le Gal**, and E. Casseau. Bit-width optimizations for high-level synthesis of digital signal processing systems. In *Proceedings of Design of Circuits and Integrated Systems conference (DCIS)*, Barcelona, Spain, November 22-24 2006.

[ci :Aub11] W. Aubry, **B. Le Gal**, D. Negru, and D. Dallet. Adaptation engine evaluation toward next generation home gateways in future media internet context. In *Proceedings of the 18th IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, Beirut, Lebanon, December 11-14 2011.

[ci :Aub12a] W. Aubry, D. Négru, **B. Le Gal**, S. Desfarges, and D. Dallet. A generic hardware video adaptation framework towards content - and context - awareness in future networks. In *Proceedings of the 20th EURASIP European Conference on Signal Processing (EUSIPCO)*, pages 2218-2222, Bucharest, Romania, August 27-31 2012.

[ci :Aub12b] W. Aubry, **B. Le Gal**, D. Negru, S. Desfarges, and D. Dallet. A generic video adaptation FPGA implementation towards content - and context - awareness in future networks. In *Proceedings of the IEEE International Conference on Design & Architectures for Signal and Image Processing (DASIP)*, Karlsruhe, Germany, October 23-25 2012.

[ci :Aub12c] W. Aubry, D. Negru, **B. Le Gal**, S. Desfarges, and D. Dallet. Spatial downsizing impact in the transrating tradeoff for content/context awareness in media network. In *Proceedings of the International Conference on Telecommunications and Multimedia (TEMU)*, pages 158-162, Heraklion, Crete, Greece, July 30 - August 1 2012.

[ci :Aul13] A. Aulery, D. Dallet, **B. Le Gal**, N. Deltimple, D. Belot, and E. Kerherve. Study and analysis of a new implementation of a mixed-signal cartesian feedback for a low power zero-IF WCDMA transmitter. In *Proceedings of the 11th IEEE International*

Conference on New Circuits and Systems (NEWCAS), pages 1-4, Paris, France, June 16-19 2013.

[ci :Belll] B. Belean, M. Borda, and **B. Le Gal**. FPGA technology and parallel computing towards automatic microarray image processing. In *Proceedings of the* 34^{th} *International Conference on Telecommunications and Signal Processing (TSP)*, Budapest, Hungary, August 18-20 2011.

[ci :Benl6] H. Ben Thameur, C. Bouzouita, N. Khouja, **B. Le Gal**, F. Tlili, and C. Jego. Hardware complexity reduction of LDPC-CC decoders based on message-passing approaches. In *Proceedings of the* 17^{th} *international conference on Sciences and Techniques of Automatic control and computer engineering (STA)*, pages 679-684, Sousse, Tunisia, December 19-21 2016.

[ci :Benl7a] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. A survey on decoding schedules of LDPC convolutional codes and associated hardware architectures. In *Proceedings of the* 22^{nd} *IEEE Symposium on Computers and Communication (ISCC)*, Crete, Greece, July 3-6 2017.

[ci :Benl7b] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Low complexity ADMM-LP based decoding strategy for LDPC convolutional codes. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, Lorient, France, October 3-5 2017.

[ci :Benl7c] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Reduced complexity ADMM-based schedules for LP decoding of LDPC convolutional codes. In *Proceedings of the* 25^{th} *International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Croatia, September 21-23 2017.

[ci :Benl7d] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. A LP-based algorithm for decoding LDPC convolutional codes. In *Proceedings of the IEEE International Conference on Internet of Things, Embedded Systems and Communications (IINTEC)*, pages 20-22, Gafsa, Tunisia, October 2017.

[ci :Benl8a] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Implementation aspects of a pipeline ADMM-based LP decoding of LDPC convolutional codes. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, Spain, April 15-18 2018.

[ci :Benl8b] H. Ben Thameur, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Hardware design of euclidean projection modules for ADMM LDPC decoding. In *Proceedings of the* 25^{th} *IEEE International Conference on Electronics Circuits and Systems (ICECS)*, Bordeaux, France, December 2018.

[ci :Cam07a] P. Camino, D. Dallet, B. Quertier, A. Baudry, G. Comoretto, and **B. Le Gal**. A decimation filter for a very large band signal in radioastronomy. In *Proceedings of the* 3^{rd} *Research in Microelectronics and Electronics Conference (PRIME)*, Talence, France, July 2-5 2007.

[ci :Cam07b] P. Camino, D. Dallet, B. Quertier, A. Baudry, **B. Le Gal**, and G. Comoretto. A multi-rate filter for broadband radiotelescopes. In *Proceedings of the* 15^{th} *IMEKO-TC4 International Symposium*, pages 55-60, Romania, September 2007.

[ci :Cas04] E. Casseau, **B. Le Gal**, C. Jégo, N. Le Héno, and E. Martin. Reed-solomon behavioral virtual component for communication systems. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 4, pages 173-176, Vancouver, Canada, May 21-23 2004.

[ci :Cas05] E. Casseau, **B. Le Gal**, P. Bomel, C. Jégo, S. Huet, and E. Martin. C-based rapid prototyping for digital signal processing. In *Proceedings of the EURASIP European Conference on Signal Processing (EUSIPCO)*, Antalya, Turquie, September 4-8 2005.

[ci :Cas07] E. Casseau, S. Khan, **B. Le Gal**, and W. Aubry. Multimode architecture design. In *Proceeding of Design and Architectures for Signal and Image Processing Workshop (DASIP)*, Grenoble, France, November 27-29 2007.

[ci :Cas09] E. Casseau and **B. Le Gal**. High-level synthesis for the design of FPGA-based signal processing systems. In *Proceedings of the 9th international Conference on Systems, Architectures, MOdeling and Simulation (SAMOS)*, pages 25-32, Samos, Greece, July 20-23 2009.

[ci :Cas15] A. Cassagne, **B. Le Gal**, C. Leroux, O. Aumage, and D. Barthou. An efficient, portable and generic library for successive cancellation decoding of polar codes. In *Proceedings of the 28th International Workshop on Languages and Compilers for Parallel Computing (LCPC)*, pages 303-317, Raleigh, NC, USA, September 9-11 2015.

[ci :Cas16a] A. Cassagne, O. Aumage, C. Leroux, D. Barthou, and **B. Le Gal**. Energy consumption analysis of software polar decoders on low power processors. In *Proceedings of the EURASIP European Conference on Signal Processing (EUSIPCO)*, Budapest, Hungary, August 29 - September 2 2016.

[ci :Cas16b] A. Cassagne, T. Tonnellier, C. Leroux, **B. Le Gal**, O. Aumage, and D. Barthou. Beyond Gbps turbo decoder on multi-core CPUs. In *Proceedings of the 9th International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Brest, France, September 5-9 2016.

[ci :Cas17] A. Cassagne, O. Hartmann, Mathieu Leonardon, T. Tonnellier, G. Delbergue, Camille Leroux, Romain Tajan, **B. Le Gal**, Christophe Jego, Olivier Aumage, et al. Fast simulation and prototyping with aff3ct. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SiPS) - Demo Night*, 2017.

[ci :Deb15] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Analysis of ADMM-LP algorithm for LDPC decoding, a first step to hardware implementation. In *Proceedings of the 22th IEEE International Conference on Electronics, Circuits, and Systems, (ICECS)*, pages 356-359, Cairo, Egypth, December 6-9 2015.

[ci :Deb16a] I. Debbabi, N. Khouja, F. Tlili, **B. Le Gal**, and C. Jego. Multicore implementation of LDPC decoders based on ADMM algorihtm. In *Proceedings of the 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 971-975, Shangai, China, March 20-25 2016.

[ci :Deb16b] I. Debbabi, N. Khouja, F. Tlili, **B. Le Gal**, and C. Jego. Evaluation of the hardware complexity of the ADMM approach for LDPC decoding. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Doha, Qatar, April 3-6 2016.

[ci :Deb16c] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jego. Comparison of different schedulings for the ADMM based LDPC decoding. In *Proceedings of the 9th International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Brest France, September 5-9 2016.

[ci :Deb18a] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jégo. ADMM hardware decoder for regular LDPC codes using a NISC-based architecture. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, Spain, April 15-18 2018.

[ci :Del15] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. Beyond qbox : development of virtual platforms based on QEMU and SystemC TLM-2.0. In *Proceedings of the Forum on Specification and Design Languages (FDL)*, Barcelona, Spain, September 2015.

[ci :Del16] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. QBox : an industrial solution for virtual platform simulation using QEMU and SystemC TLM-2.0. In *Proceedings of the European Congress Embedded Real Time Software And Systems (ERTS)*, Toulouse, France, January 2016.

[ci :Del18a] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Model-based design of efficient LDPC decoder architectures. In *Proceedings of the* 10^{th} *International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Hong Kong, China, December 2018.

[ci :Del18b] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Fast design of reliable, flexible and high-speed AWGN architectures with high level synthesis. In *Proceedings of the* 25^{th} *IEEE International Conference on Electronics Circuits and Systems (ICECS)*, Bordeaux, France, December 2018.

[ci :Del18c] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. From multicore LDPC decoder implementations to FPGA decoder architectures : a case of study. In *Proceedings of the* 25^{th} *IEEE International Conference on Electronics Circuits and Systems (ICECS)*, Bordeaux, France, December 2018.

[ci :Del19a] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Generation of efficient self-adaptive hardware polar decoders using high-level synthesis. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, Nanjing, China, October 2019.

[ci :Eyn10] G .Eynard, N. Lewis, D. Dallet, and **B. Le Gal**. Efficient phase noise modeling of a PLL-based frequency synthesizer. In *Proceedings of the* 4^{th} *International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pages 1-4, Limassol, Cyprus, March 3-5 2010.

[ci :Far08] H. Fares, C. Rebai, **B. Le Gal**, and D. Dallet. Optimized decimation structure for complex bandpass $\sigma\delta$ modulator in wideband receiver. In *Proceedings of the IMEKO Workshop on ADC and DAC Modelling and Testing.*, Florence, Italy, September 22-24 2008.

[ci :Fer08a] G. Ferre, **B. Le Gal**, L. Bossuet, D. Dallet, J. Maher, and P. Colucci. Orthogonal correction implementation for time interleaved analog-to-digital converters : real-time application. In *Proceedings of the* 16^{th} *EURASIP European Conference on Signal Processing (EUSIPCO)*, Lausanne, Switzerland, August 25-29 2008.

[ci :Fer08b] G. Ferre, M. Jridi, L. Bossuet, **B. Le Gal**, and D. Dallet. A new orthogonal online digital calibration for time-interleaved analog-to-digital converters. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 576-579, May 18-21 2008.

[ci :Gil22a] C. Gillet, A. F. Vincent, **B. Le Gal**, and S. Saighi. Flexible design methodology for spike encoding implementation on FPGA. In *Proceedgins of the IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Taipei, Taiwan, October 13-15 2022.

[ci :Gra09] M. Grand, L. Bossuet, **B. Le Gal**, D. Dallet, and G. Gogniat. A reconfigurable crypto sub system for the software communication architecture. In *Proceedings of the* 28^{th}

IEEE conference on Military communications (MILCOM), pages 2708–2714, Boston, Massachusetts, USA, October 18–21 2009.

[ci :Gral0] M. Grand, L. Bossuet, **B. Le Gal**, D. Dallet, and G. Gogniat. A multi-core AES cryptoprocessor for multi-channel SDR. In *Proceedings of the Military Communications and Information Systems Conference (MCC)*, pages 1–7, Wroclaw, Poland, September 27 - October 1 2010.

[ci :Gralla] M. Grand, L. Bossuet, **B. Le Gal**, D. Dallet, and G. Gogniat. A reconfigurable multi-core cryptoprocessor for multi-channel communication systems. In *Proceedings of the 18th Reconfigurable Architectures Workshop (RAW)*, pages 204–211, Anchorage, USA, May 16–17 2011.

[ci :Grallb] M. Grand, L. Bossuet, **B. Le Gal**, G. Gogniat, and D. Dallet. Design and implementation of a multi-core crypto-processor for software defined radios. In *Proceedings of the 7th international conference on Reconfigurable computing : architectures, tools and applications (ARC)*, pages 29–40, Belfast, United Kingdom, March 2011.

[ci :Har18a] H. Harb, E. Boutillon, and **B. Le Gal**. Real-time evaluation of NB-LDPC codes thanks to HLS-based hardware emulation. In *Proceedings of the IEEE Conference on Design and Architectures for Signal and Image Processing (DASIP)*, Porto, Portugal, October 2018.

[ci :Hue04] S. Huet, P. Bomel, E. Casseau, **B. Le Gal**, and O. Pasquier. Electronic system level to HW/SW design flow. In *Proceedings of the Global Signal Processing conference (GSPx)*, Santa Clara, CA, USA, September 27-30 2004.

[ci :Jeb08] M. Jebalia, C. Rebai, **B. Le Gal**, and D. Dallet. Optimized decimation filter architecture for 5th order $sigma\delta$ converter in GSM/UMTS/WiMAX radio receiver. In *Proceedings of the IEEE International conference on Signals, Circuits & Systems (SCS)*, Hammamet, Tunisia, November 7-9 2008.

[ci :Jella] Z. Jelassi, D. Dallet, C. Rebai, **B. Le Gal**, and C. Jego. An LCF-ADC model using co-simulation technique. In *Proceedings of the International Workshop on ADC Modeling, Testing and Data Converter Analysis and Design (IMEKO-IWADC)*, Orvieto, Italy, June 30 - July 1 2011.

[ci :Jri07a] M. Jridi, L. Bossuet, **B. Le Gal**, and D. Dallet. New adaptive calibration method for time interleaved analog to digital converters. In *Proceedings of the 50th MWSCAS and 5th IEEE International New Circuits and Systems Conference (NEWCAS) Conference*, pages 932–935, Montreal, Canada, August 5–8 2007.

[ci :Jri07b] M. Jridi, L. Bossuet, **B. Le Gal**, and D. Dallet. Mismatch error compensation method for time-interleaved A/D converters. In *Proceedings of the 12th International Workshop on ADC Modeling and Testing*, September 2007.

[ci :Kho08] N. Khouja, K. Grati, A. Ghazel, and **B. Le Gal**. Low power design of a decimation filter in multi-standard receiver. In *Proceedings of the 3rd International Conference on Design and Technology of Integrated Systems in Nanoscale Era (DTIS)*, pages 1–5, Tozeur, Tunisia, March 25-28 2008.

[ci :Kho09a] N. Khouja, B. Le Gal, K. Grati, and A. Ghazel. Experiments on designing low power decimation filter for multistandard receiver on heterogeneous targets. In *Proceedings of the 17th EURASIP European Conference on Signal Processing (EUSIPCO)*, pages 1284–1288, Glasgow, Scotland, August 24-28 2009.

[ci :Kho09b] N. Khouja, K. Grati, A. Ghazel, and **B. Le Gal**. Design and implementation of a reconfigurable decimation and channel selection filter for GSM and UMTS radio standards. In *Proceedings of the IEEE conference on Wireless Communications & Networking Conference (WCNC)*, pages 323-328, Budapest, Hungary, April 5-8 2009.

[ci :Kho10] N. Khouja, K. Grati, A. Ghazel, and **B. Le Gal**. Evaluation of FIR filters implementations for low power multistandard receivers. In *Proceedings of the 5th International Conference on Design & Technology of Integrated Systems (DTIS)*, pages 1-5, Hammamet, Tunisia, March 23-25 2010.

[ci :Khoa] N. Khouja, K. Grati, **B. Le Gal**, and A. Ghazel. Power consumption modeling of decimation fir filters for multistandard receivers". In *Proceedings of the 6th IEEE International Conference on Design & Technology of Integrated Systems in nanoscale era (DTIS)*, Athens, Greece, April 6-8 2011.

[ci :Khob] N. Khouja, K. Grati, **B. Le Gal**, and A. Ghazel. Power consumption estimation models for FIR decimation filters in multi-standard receivers. In *Proceedings of the 6th IEEE Grids, Clusters and Clouds Conference and Exhibition (GCC)*, pages 617-620, Dubai, United Arab Emirates, February 19-22 2011.

[ci :Kthi] M. Kthiri, P. Kadionik, **B. Le Gal**, A. Ben Atitallah, and H. Lévi. Performance analysis and evaluation of Xenomai with H.264/AVC decoder. In *Proceedings of the 23rd IEEE International Conference on Microelectronics (ICM), Special session on Reliability of Systems on Chip (SoC)*, Hammamet, Tunisia, December 19-22 2011.

[ci :LeG04] **B. Le Gal**, E. Casseau, P. Bomel, C. Jégo, N. Le Héno, and E. Martin. High level synthesis assisted rapid prototyping for digital signal processing. In *Proceeding of 16th International IEEE Conference on Microelectronics (ICM)*, pages 746-749, Tunis, Tunisia, December 6-8 2004.

[ci :LeG05a] **B. Le Gal**, E. Casseau, S. Huet, and E. Martin. Pipelined memory controllers for DSP applications handling unpredictable data accesses. In *Proceedings of the IEEE Computer Society Annual Symposium on VLSI : New Frontiers in VLSI Design (ISVLSI)*, pages 268—269, Tampa, Florida, USA, May 11-12 2005.

[ci :LeG05b] **B. Le Gal**, E. Casseau, and E. Martin. Bounded budgeted parallel architecture versus control dominated architecture for hazard data-signal processors synthesis. In *Proceedings of the SPIE Conference, Microtechnologies for the New Millennium (SPIE)*, volume 5837, pages 100—111, Seville, Spain, May 9-11 2005.

[ci :LeG05c] **B. Le Gal**, E. Casseau, and E. Martin. Pipelined memory controllers for DSP real time applications handling unpredictable data accesses. In *Proceedings of the EURASIP European Conference on Signal Processing (EUSIPCO)*, Antalya, Turkey, September 4-8 2005.

[ci :LeG05d] **B. Le Gal**, E. Casseau, C. Andriamisaina, and E. Martin. Dynamic memory access management and address computation for dataflow applications. In *Proceedings of the IFIP International Conference on Very Large Scale Integration (VLSI-SOC)*, pages 152—157, Perth, Australia, October 17-19 2005.

[ci :LeG06a] **B. Le Gal**, C. Andriamisaina, and E. Casseau. Bit-width aware high-level synthesis for digital signal processing systems. In *Proceedings of IEEE System-On-Chip Conference (SOC)*, pages 175-178, Austin Texas, September 24-27 2006.

[ci :LeG06b] **B. Le Gal** and E. Casseau. IP generation targeting multiple bit-width standards. In *Proceeding of 13th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 784-787, Nice, France, December 10-13 2006.

[ci :LeG07a] **B. Le Gal**, L. Bossuet, S. Khan, and E. Casseau. HLS design flow for the synthesis of multimode systems under multiple constraints. In *Proceeding of 14th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 314 -317, Marrakech, Morocco, December 11-14 2007.

[ci :LeG07b] **B. Le Gal**, L. Bossuet, and D. Dallet. Mathematical functions based watermarking for IPP. In *Proceedings of the 14th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 310-313, Marrakech, Morocco, December 11-14 2007.

[ci :LeG09] **B. Le Gal** and E. Casseau. Automated multimode system design for high performance DSP applications. In *Proceedings of the 17th EURASIP European Conference on Signal Processing (EUSIPCO)*, pages 1289—1293, Glasgow, Scotland, August 24-28 2009.

[ci :LeG10a] **B. Le Gal**, L. Bossuet, and D. Dallet. Using digital signal processor singularities to minimize ROM based controller area. In *Proceedings of the 8th IEEE International New Circuits and Systems Conference (NEWCAS)*, pages 29-32, Montreal, Canada, June 20-23 2010.

[ci :LeG10b] **B. Le Gal**, A. Ribon, L. Bossuet, and D. Dallet. Area optimization of ROM-based controllers dedicated to digital signal processing applications. In *Proceedings of the 18th EURASIP European Conference on Signal Processing (EUSIPCO)*, pages 547-551, Aalborg, Denmark, August 23-27 2010.

[ci :LeG10c] **B. Le Gal**, A. Ribon, L. Bossuet, and D. Dallet. Reducing and smoothing power consumption of ROM-based controller implementations. In *Proceedings of the 23rd Symposium on Integrated Circuits and System Design (SBCCI)*, pages 8-13, Sao Paulo, Brazil, September 6-9 2010.

[ci :LeG11] **B. Le Gal** and L. Bossuet. Automatic HLS based low-cost IP watermarking. In *Proceedings of the 9th IEEE International New Circuits and Systems Conference (NEWCAS)*, pages 490-493, Bordeaux, France, June 26-29 2011.

[ci :LeG12a] **B. Le Gal** and C. Jego. Improving architecture efficiency of softcore processors. In *Proceedings of the International Conference on Embedded Real Time Software and Systems (ERTS)*, pages 1-7, Toulouse, France, February 1-3 2012.

[ci :LeG12b] **B. Le Gal** and C. Jego. Design of an ASIP LDPC decoder compliant with digital communication standards. In *Proceedings of the IEEE Workshop on Signal Processing Systems (SIPS)*, pages 19-24, Québec City, Québec, Canada, October 17-19 2012.

[ci :LeG12c] **B. Le Gal** and C. Jego. FPGA prototyping of an ASIP LDPC decoder for the DVB-T2 standard. In *Proceedings of the IEEE International Conference on Design & Architectures for Signal and Image Processing (DASIP)*, pages 1-2, Karlsruhe, Germany, October 23-25 2012.

[ci :LeG14] **B. Le Gal**, C. Leroux, and C. Jego. Software polar decoder on an embedded processor. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, pages 1-6, Belfast, UK, October 20-22 2014.

[ci :LeG16a] **B. Le Gal**, C. Leroux, and C. Jégo. Memory reduction techniques for successive cancellation decoding of polar codes. In *Proceedings of the 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 986-990, Shangai, China, March 20-25 2016.

[ci :LeG16b] **B. Le Gal**, C. Leroux, and C. Jégo. A scalable 3-phase polar decoder. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCASS)*, pages 417-420, Montreal, Canada, May 22-25 2016.

[ci :LeGl7a] **B. Le Gal** and C. Jego. Low-latency software LDPC decoders. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, Lorient, France, October 3-5 2017.

[ci :LeGl7b] **B. Le Gal**, C. Leroux, and C. Jego. Successive cancellation decoder for very long polar codes. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, Lorient, France, October 3-5 2017.

[ci :LeG20a] **B. Le Gal**, Y. Delomier, C. Leroux, and C. Jego. Low-latency sorter architecture for polar codes successive-cancellation-list decoding. In *Proceedings of the IEEE International Workshop on Signal Processing Systems (SIPS)*, Coimbra, Portugal, October 20-22 2020.

[ci :LeG22a] **B. Le Gal**, V. Pignoly, and C. Jégo. High-performance Gallager-E decoders for hard input LDPC decoding on multi-core devices. In *Proceedings of the International Workshop on Design and Architecture for Signal and Image Processing (DASIP)*, pages 3-15, Budapest, Hungary, June 2022.

[ci :Mec10] N. Mechouk, D. Dallet, L. Bossuet, and **B. Le Gal**. A low-area filter bank design methodology for on-chip ADC testing. In *Proceedings of the 17th IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pages 718-721, Athens, Greece, December 12-15 2010.

[ci :Mon21a] C. Moniere, K. Saied, **B. Le Gal**, and E. Boutillon. Time sliding window for the detection of CCSK frames. In *Proceedings of the International Workshop on Signal Processing Systems (SIPS)*, Coimbra, Portugal, October 19-21 2021.

[ci :Mon22a] C. Moniere, **B. Le Gal**, and E. Boutillon. Efficient software and hardware implementations of a QCSP communication system. In *Proceedings of the International Workshop on Design and Architecture for Signal and Image Processing (DASIP)*, pages 29-41, Budapest, Hungary, June 2022.

[ci :Nou18a] S. Noubir, Y. Bornat, and **B. Le Gal**. Real-time analysis of living biological cell activity. In *Proceedings of the IEEE Conference on Design and Architectures for Signal and Image Processing (DASIP)*, Porto, Portugal, October 2018.

[ci :Nou18b] S. Noubir, Y. Bornat, and **B. Le Gal**. A scalable and efficient digital signal processing system for real time biological spike detection. In *Proceedings of the 25th IEEE International Conference on Electronics Circuits and Systems (ICECS)*, Bordeaux, France, December 2018.

[ci :Pig18a] V. Pignoly, **B. Le Gal**, C. Jego, and B. Gadat. High data rate and flexible hardware QC-LDPC decoder for satellite optical communications. In *Proceedings of the 10th International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Hong Kong, China, December 2018.

[ci :Pig20a] V. Pignoly, **B. Le Gal**, C. Jégo, and B. Gadat. High speed LDPC decoding for optical space link. In *Proceedings of the IEEE International Conference on Electronics Circuits and Systems (ICECS)*, Glasgow, Scotland, November 23-25 2020.

[ci :Pig20b] V. Pignoly, **B. Le Gal**, C. Jégo, B. Gadat, and L. Barthe. Fair comparison of hardware and software LDPC decoder implementations for SDR space links. In *Proceedings of the IEEE International Conference on Electronics Circuits and Systems Conference (ICECS)*, Glasgow, Scotland, November 23-25 2020.

[ci :Pil23] S. Pillement, M. Mendez Real, J. Pottier, T. Nieddu, **B. Le Gal**, S. Faucou, J.L. Bechennec, M. Briday, S. Girbal, J. Le Rhun4, O. Gilles, D. Gracia Perez, A. Sintzoff,

and J.R. Coulon. Securing a RISC-V architecture : A dynamic approach. In *Proceedgins of the IEEE Design, Automation and Test in Europe Conference and Exhibition (DATE)*, Antwerpen, Belgium, April 2023.

[ci :Rib10] A. Ribon, **B. Le Gal**, L. Bossuet, and D. Dallet. Boolean assertion high-level synthesis for architecture simulation enhancement. In *Proceedings of the System, Software, SoC and Silicon Debug Conference (S4D)*, Southampton, UK, September 2010.

[ci :Rib11] A. Ribon, **B. Le Gal**, C. Jego, and D. Dallet. Assertion support in high-level synthesis design flow. In *Proceedings of the IEEE Forum on Specification and Design Languages (FDL)*, pages 1-8, Oldenburg, Germany, September 13-15 2011.

[ci :San12a] W. Sanaa, N. Delaunay, **B. Le Gal**, D. Dallet, C. Rebai, N. Deltimple, E. Kerherve, and D. Belot. Design of a mixed-signal cartesian feedback loop for a low power Zero-IF WCDMA transmitter. In *Proceedings of the 3rd IEEE Latin American Symposium on Circuits and Systems*, (LASCAS'12), Playa del Carmen, Mexico, February 29 - March 2 2012.

[ci :San12b] W. Sanaa, **B. Le Gal**, D. Dallet, C. Rebai, N. Deltimple, E. Kerherve, and D. Belot. New digital predistortion design based on mixed-signal cartesian feedback training for 3G homodyne transmitter. In *Proceedings of the 10th IEEE International New Circuits And Systems conference (NEWCAS)*, Montreal, Canada, June 17-20 2012.

[ci :Sol13] J. Solanki, L. Réveillère, Y.-D. Bromberg, **B. Le Gal**, and T. F. Bissyandé. Improving the performance of message parsers for embedded systems. In *Proceeding of the 28th ACM Symposium on Applied Computing (SAC)*, pages 1505-1510, Coimbra, Portugal, March 18-22 2013.

[ci :StG22a] L. Saint Germain, **B. Le Gal**, J. Crenne, F. Baldacci, C. Jego, and S. Loty. Methodology to adapt neural network on constrained device at topology level. In *Proceedings of the IEEE Workshop on Signal Processing Systems (SIPS)*, Rennes, France, November 2-4 2022.

[ci :Ton16a] T. Tonnellier, C. Leroux, **B. Le Gal**, C. Jego, B. Gadat, and N. Van Wambeke. Lowering the error floor of double-binary turbo codes : the flip and check algorithm. In *Proceedings of the 9th International Symposium on Turbo Codes & Iterative Information Processing (ISTC'16)*, Brest France, September 5-9 2016.

[ci :Ton16b] T. Tonnellier, C. Leroux, **B. Le Gal**, C. Jego, B. Gadat, and N. Van Wambeke. Hardware architecture for lowering the error floor of LTE turbo codes. In *Proceedings of the Conference on Design & Architectures for Signal & Image Processing (DASIP)*, Rennes, France, October 12-14 2016.

[ci :Tou21a] M. Tourres, C. Chavet, **B. Le Gal**, J. Crenne, and P. Coussy. Extended RISC-V hardware architecture for future digital communication systems. In *the Proceedings of the International IEEE 5G World Forum (5GWF)*, Montreal, Canada, October 13-15 2021.

[ci :Tou22a] M. Tourres, C. Chavet, **B. Le Gal**, J. Crenne, and P. Coussy. Specialized scalar and SIMD instructions for error correction codes decoding on RISC-V processor. In *Proceedings of the Spring RISC-V Week*, Paris, France, May 2022.

[ci :Vol22a] L. Volpin, **B. Le Gal**, and G. Ferre. Efficient lora-like transmitter stacks for sdr applications. In *Proceedings of the IEEE International Conference on Circuits and Systems (ICECS)*, pages P-P, Glasgow, UK, October 24-26 2022.

Articles dans des conférences internationales (sans actes)

[ci :Bos08] L. Bossuet and **B. Le Gal**. Cryptarchi web site : A collaborative web site for teaching hardware security. In *International CryptArchi Workshop*, Trégastel, France, June 1-4 2008.

[ci :LeG07d] **B. Le Gal**, L. Bossuet, and D. Dallet. Intellectual property protection using watermarking. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (Gdr SoC-SiP)*, Paris, France, June 13-15 2007.

Articles dans des conférences nationales

[cn :Alm22a] H. Almorin, **B. Le Gal**, Jérémie Crenne, Christophe Jego, and Vincent Kissel. Conception d'architectures de FFT pour FPGA à base de modèles comportementaux. In *Proceedings of the Colloque National du GDR SoC²*, Strasbourg, France, June 2022.

[cn :Alm22b] H. Almorin, **B. Le Gal**, Jérémie Crenne, Christophe Jego, and Vincent Kissel. Conception d'architectures de FFT pour FPGA à base de modèles comportementaux. In *Proceedings of the COMPAS conference*, Amien, France, July 2022.

[cn :Bos08a] L. Bossuet and **B. Le Gal**. Cryptarchi web site : A collaborative web site for teaching hardware security. In *International CryptArchi Workshop*, Trégastel, France, June 1-4 2008.

[cn :Bos08b] L. Bossuet and **B. Le Gal**. Site internet collaboratif pour l'enseignement de la sécurité matérielle. In *les actes du 7ème Colloque sur l'Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS)*, Bruxelles, Belgique, October 27-29 2008.

[cn :Cas17] A. Cassagne, M. Leonardon, O. Hartmann, T. Tonnellier, G. Delbergue, V. Giraud, C. Leroux, R. Tajan, **B. Le Gal**, C. Jego, O. Aumage, and D. Barthou. AFF3CT : Un environnement de simulation pour le codage de canal. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (GdR SoC-SiP)*, Bordeaux, France, 14-16 Juin 2017.

[cn :Deb17b] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jégo. Décodage par programmation linéaire des codes LDPC. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (GdR SoC-SiP)*, Bordeaux, France, 14-16 Juin 2017.

[cn :Deb17c] I. Debbabi, **B. Le Gal**, N. Khouja, F. Tlili, and C. Jégo. Architecture matérielle pour un décodeur par programmation linéaire des codes LDPC. In *les actes de la Conférence GRETSI (Groupe de Recherche et d'Etudes du Traitement du Signal et des Images)*, Juan-les-Pins, France, September 5-8 2017.

[cn :Del17] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. Plateformes virtuelles SystemC/TLM : configuration, communication et parallelisation. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (GdR SoC-SiP)*, Bordeaux, France, 14-16 Juin 2017.

[cn :Del18a] Y. Delomier, **B. Le Gal**, J. Crenne, and C. Jego. Génération d'architectures de décodeur LDPC à l'aide d'un outil de synthèse de haut niveau. In *les actes de la Conférence COMPAS (Conférence d'informatique en Parallélisme, Architecture et Système)*, Toulouse, France, Juin 2018.

[cn :Jri07] M. Jridi, L. Bossuet, **B. Le Gal**, and D. Dallet. Les convertisseurs analogique numérique entrelacés dans le temps : théorie et nouvelles tendances des méthodes de correction d'erreurs. In *les actes de la 10ème édition des Journées Nationales du réseau Doctoral en Microélectronique*, (JNRDM'07), pages 1-4, Lille, France, May 14-15 2007.

[cn :Kth10a] M. Kthiri, P. Kadionik, **B. Le Gal**, H. Lévi, and B. A. Ahmed. Evaluation des performances de xenomai avec un décodeur H.264/AVC. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (Gdr SoC-SiP)*, Lyon, France, June 15-17 2010.

[cn :Kth10b] M. Kthiri, P. Kadionik, B. A. Ahmed, **B. Le Gal**, and H. Lévi. Implémentation matérielle du filtre anti-bloc pour la norme H.264/AVC. In *Dans les actes da la Conférence GRETSI (Groupe de Recherche et d'Etudes du Traitement du Signal et des Images)*, Bordeaux, France, September 5-8 2010.

[cn :LeG05] **B. Le Gal**, E. Casseau, C. Andriamisaina, S. Huet, and E. Martin. Séquenceur mémoire pour les applications multimédia temps réel gérant des séquences d'accès indéterministes. In *les actes da la Conférence GRETSI (Groupe de Recherche et d'Etudes du Traitement du Signal et des Images)*, pages 823-826, Louvain-la-Neuve, Belgique, September 6-9 2005.

[cn :LeG07a] **B. Le Gal**, L. Bossuet, and D. Dallet. Intellectual property protection using watermarking. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (Gdr SoC-SiP)*, Paris, France, June 13-15 2007.

[cn :LeG08] **B. Le Gal** and L. Bossuet. Travaux pratiques de gestion de périphériques d'une carte FPGA Xilinx Starter-Kit Spartan3-E à l'aide de picoprocesseur (picoBlaze). In *les actes du 7ème Colloque sur l'Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS)*, Bruxelles, Belgique, October 27-29 2008.

[cn :LeG10] **B. Le Gal**, L. Bossuet, and M. Grand. Enseignement ludique de la programmation objets à l'aide des applications de traitement de l'image. In *les actes du 8ème Colloque sur l'Enseignement des Technologies et des Sciences de l'Information et des Systèmes*, (CETSIS'10), Grenoble, France,, March 8-10 2010.

[cn :LeG15a] **B. Le Gal**, C. Leroux, and C. Jégo. Réduction de la complexité mémoire pour le décodage à annulation successive des codes polaires. In *Dans les actes de la Conférence GRETSI (Groupe de Recherche et d'Etudes du Traitement du Signal et des Images)*, Lyon, France, 8-11 Septembre 2015.

[cn :Mec08] N. Mechouk, L. Bossuet, **B. Le Gal**, and D. Dallet. Unité de gestion des modes d'un IP multimode. In *les actes de la 11ème édition des Journées Nationales du réseau Doctoral en Microélectronique (JNRDM)*, Bordeaux, France, Mai 2008.

[cn :Mon22a] C. Monière, **B. Le Gal**, and Emmanuel Boutillon. Real-time QCSP communication system prototyping. In *Proceedings of the Colloque National du GDR SoC²*, Strasbourg, France, June 2022.

[cn :Mon22b] C. Monière, **B. Le Gal**, and Emmanuel Boutillon. Implémentations logicielles et matérielles efficientes d'une chaine de communications QCSP. In *Proceedings of the COMPAS conference*, Amien, France, July 2022.

[cn :Rib10a] A. Ribon, **B. Le Gal**, L. Bossuet, and D. Dallet. Gestion des assertions comportementales dans un flot de synthèse de haut-niveau. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package*, (Gdr SoC-SiP), Paris, France, June 9-11 2010.

[cn :Rib10b] A. Ribon, **B. Le Gal**, D. Dallet, and L. Bossuet. Génération de moniteurs matériels dans un processus de synthèse de haut-niveau. In *les actes de la 13ème édition des Journées Nationales du réseau Doctoral en Microélectronique (JNRDM)*, Montpellier, France,, Juin 7-9 2010.

[cn :StG22] L. Saint-Germain, **B. Le Gal**, Jeremie Crenne, Fabien Baldacci, Christophe Jego, and Sébastien Loty. Methodology for neural network adaptation for low-end hardware processors. In *Proceedings of the COMPAS conference*, Amien, France, July 2022.

[cn :Ton15a] T. Tonnellier, C. Leroux, **B. Le Gal**, C. Jego, B. Gadat, and C. Poulliat. L'algorithme self-corrected max-log-map pour le décodage des turbo codes. In *Dans les actes de la 18ème édition des Journées Nationales du réseau Doctoral en Microélectronique (JNRDM)*, Bordeaux, France, Mai 2015.

[cn :Ton15b] T. Tonnellier, C. Leroux, **B. Le Gal**, C. Jego, B. Gadat, and C. Poulliat. Extension du principe self-corrected de l'information extrinsèque au décodage itératif de turbo codes. In *Dans les actes de la Conférence GRETSI (Groupe de Recherche et d'Etudes du Traitement du Signal et des Images)*, Lyon, France, 8-11 Septembre 2015.

[cn :Ton17] T. Tonnellier, C. Leroux, **B. Le Gal**, and C. Jego. Correction des erreurs résiduelles lors du processus de turbo décodage : algorithme et architecture. In *les actes du Groupement de Recherche System-on-Chip, System-in-Package (GdR SoC-SiP)*, Bordeaux, France, 14-16 Juin 2017.

[cn :Tou21a] M. Tourres, C. Chavet, P. Coussy, **B. Le Gal**, and J. Crenne. Architecture programmable pour les systèmes de communications numériques. In *les actes du Groupement de Recherche System-on-Chip (GdR SoC2)*, Rennes, France, Juin 2021.

[cn :Tou21b] M. Tourres, **B. Le Gal**, J. Crenne, C. Chavet, and P. Coussy. Architecture matérielle programmable optimisée pour les systèmes de communications numérique. In *les actes de la Conférence francophone d'informatique en Parallélisme, Architecture et Système (COMPAS)*, Lyon, France, Juillet 2021.

Présentations invitées, audience internationale

[ci :Delt12] N. Deltimple, N. Delaunay, W. Sanaa, **B. Le Gal**, C. Rebai, D. Dallet, D. Belot, and E. Kerherve. Cartesian feedback with digital enhancement applied to fully integrated cmos rf transmitter. In *Proceedings of the International Microwave Symposium, Workshop WSD : RF & mmW PAs : Linearization and Power Challenges (IMS)*, Montreal, Canada, 2012.

[pi :Del15] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. System-level modeling for today and tomorrow with SystemC (part of the talk). In *the Design and Verification Conference & Exhibition Europe (DVCON Europe)*,, Munich, Germany, November 2015.

[pi :Del16a] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. Analysis of tlm-2.0 and it's applicability to non memory mapped interfaces. In *the Workshop on the Evolution of SystemC-related Standards (SystemC Evolution Day 2016), Intel Campeon*, Munich, Germany, May 3 2016.

[pi :Del16b] G. Delbergue, M. Burton, **B. Le Gal**, and C. Jego. A SystemC asynchronous wait mechanism enabling multi-threading and multi-simulator support. In *In the Workshop*

on the Evolution of SystemC-related Standards (SystemC Evolution Day 2016), Intel Campeon, Munich, Germany, May 3 2016.

Présentations invitées, audience nationale

[pn :Aull2] A. Aulery, W. Sanaa, D. Dallet, **B. Le Gal**, N. Deltimple, D. Belot, and E. Kerherve. Design of a mixed-signal cartesian feedback loop for a low power Zero-IF WCDMA transmitter. In *la Journée du GdR SOC/SIP, thématique AMS&RF, sur la correction numérique des imperfections de l'analogique*, Télécom ParisTech, Paris, Mai 24 2012.

[pn :LeG09a] **B. Le Gal**. Introduction à la synthèse de haut niveau. In *les actes de l'école thématique d'Architectures des systèmes matériels enfouis et méthodes de conception associées*, Pleumeur-Bodou (Cotes-d'Armor), France, Mars 30 - Avril 3 2009.

[pn :LeG09b] **B. Le Gal**. Outil de synthèse VHDL à partir d'un langage de haut niveau. In *le séminaire : "les architectures de traitement du signal embarquées pour le spatial" organisé par le Centre National d'Etudes Spatiales (CNES)*, Toulouse, France, Novembre 19 2009.

[pn :LeG12] **B. Le Gal** and C. Jégo. Architecture ASIP flexible dédiée aux codes LDPC. In *les Journées du BarCamp Reconfigurable (GDR SoC/SiP)*, Carry le Rouet, France, Octobre 2 2012.

[pn :LeG15] **B. Le Gal**, D. Bromberg, and L. Réveillère. Architecture SoC/SoPC pour les applications réseaux nécessitant des fonctions de parsing. In *la Journée thématique du GDR SoC-SIP - Nouveaux besoins et nouvelles approches en architecture de systèmes*, Télécom ParisTech Paris, Novembre 18 2015.

[pn :LeG16] **B. Le Gal**, C. Leroux, and C. Jégo. Progress in software implementations of ECC decoders. In *la Journée thématique du GDR ISIS - Energy-efficiency in Error-Correction Coding*, Télécom ParisTech, Paris, Juin 8 2016.

[pn :LeG22] **B. Le Gal**. Evolution des techniques de parallélisation et des plateformes d'intégration pour la SDR. In *la Journée thématique du GDR ISIS - Implémentations, outils et applications émergentes pour la radio logicielle*, Sorbonne Université, LIP6, Paris, Avril 7 2022.

[pn :Mon22] C. Moniere, **B. Le Gal**, and E. Boutillon. Démonstrateur temps réel de transmission de trames courtes QCSP sur la bande ISM. In *la Journée thématique du GDR ISIS - Implémentations, outils et applications émergentes pour la radio logicielle*, Télécom ParisTech, Paris, Avril 7 2022.

[pn :Pign19a] V. Pignoly, **B. Le Gal**, C. Jégo, and B. Gadat. Conception, implémentation et prototypage sur circuit FPGA de décodeurs LDPC pour les communications spatiales. In *le séminaire : "Codes correcteurs d'erreurs pour les communications spatiales" organisé par le Centre National d'Etudes Spatiales (CNES)*, Toulouse, France, Juin 19 2019.

[pn :Pign19b] V. Pignoly, **B. Le Gal**, C. Jégo, and B. Gadat. Conception, implémentation et prototypage sur circuit FPGA de décodeurs LDPC pour les communications optiques spatiales. In *le séminaire : "Codage, modulation et traitement du signal pour les communications optiques" organisé par le GdR ISIS*, Paris, France, Septembre 19 2019.

Articles de revues internationales (soumis)

[un :Mon22a] C. Moniere, **B. Le Gal**, and E. Boutillon. Efficient implementations of a QCSP communication system. *Journal of System Architecture (JSA), ELSEVIER*, pages P-P, October 2022.

[un :Tou22] M. Tourres, C. Chavet, **B. Le Gal**, J. Crenne, and P. Coussy. Specialized scalar and SIMD instructions for error correction codes decoding on RISC-V processors. *IEEE Open Journal of the Computer Society (OJCS)*, September 2022.

Bibliographie

[4G-LTE] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); LTE physical layer; General description (version 17)*, March 2022.

[5G] 3GPP. *5G; NR; Multiplexing and channel coding (TS 38.212 version 15.2.0)*, July 2018.

[Akm14] A. Akmalkhodzhaev and A. Kozlov. New iterative turbo code list decoder. In *Proceedings of the International Symposium on Problems of Redundancy in Information and Control Systems*, pages 15-18., St. Petersburg, Russia, June 2014.

[Ala11] A. Alamdar-Yazdi and F. R. Kschischang. A simplified successive-cancellation decoder for polar codes. *IEEE Communications Letters*, 2011.

[All08] M. Alles, T. Vogt, and N. Wehn. FlexiChaP : A reconfigurable ASIP for convolutional, turbo, and LDPC code decoding. In *Proceedings of the 5th International Symposium on Turbo Codes and Related Topics*, sept. 2008.

[Ama09] Erick Amador, Renaud Pacalet, and Vincent Rezard. Optimum LDPC decoder : A memory architecture problem. In *Proceedgins of the ACM/IEEE Design Automation Conference (DAC)*, pages 891-896, 2009.

[And14] J. Andrade and et al. Combining flexibility with low power : Dataflow and wide-pipeline LDPC decoding engines in the Gbit/s era. In *2014 IEEE 25th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pages 264-269, June 2014.

[And16] J. Andrade, G. Falcao, V. Silva, and L. Sousa. A survey on programmable LDPC decoders. *IEEE Access*, 4 :6704-6718, July 2016.

[And96] J. D. Andersen. Turbo codes extended with outer BCH code. 32(22) :2059-2060, October 1996.

[Ara20] Matheus Cavalcante, Fabian Schuiki, Florian Zaruba, Michael Schaffner, and Luca Benini. Ara : A 1-GHz+ scalable and energy-efficient RISC-V vector processor with multiprecision floating-point support in 22-nm fd-soi. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 28(2) :530-543, 2020.

[Ari08] E. Arikan. A performance comparison of polar codes and reed-muller codes. *IEEE Communications Letters*, 12(6) :447-449, June 2008.

[Ari09] E. Arikan. Channel polarization : A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55(7) :3051-3073, 2009.

[Bac12] Jonathan Bachrach, Huy Vo, Brian Richards, Yunsup Lee, Andrew Waterman, Rimas Avižienis, John Wawrzynek, and Krste Asanovi. Chisel : Constructing

hardware in a scala embedded language. In *Proceedings of the 49th Annual Design Automation Conference*, DAC '12, pages 1216–1225, New York, NY, USA, 2012. Association for Computing Machinery.

[Bar13] Siddharth Barman, Xishuo Liu, Stark C. Draper, and Benjamin Recht. Decomposition methods for large scale lp decoding. *IEEE Transactions on Information Theory*, 59(12) :7870–7886, 2013.

[Ben10] Yosi Ben-Asher, Danny Meisler, and Nadav Rotem. Reducing memory constraints in modulo scheduling synthesis for fpgas. *ACM Transactions on Reconfigurable Technology and Systems*, 3(3) :15 :1–15 :19, 2010.

[Ben20] Mohamed Amine Ben Temim, Guillaume Ferré, and Romain Tajan. A novel approach to enhance the robustness of LoRa-Like PHY layer to synchronization errors. In *Proceedgins of the IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Taipei, Taiwan, December 2020.

[Ben65] V. E. Benes. *Mathematical theory of connecting networks and telephone traffic*. Academic Press, New York, 1965.

[Ber93] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding : Turbo-codes,. In *Procroceedings of the International Conference on Communications (ICC)*, pages 1064–1070, Geneva, Switzerland, May 1993.

[Bha07] J. Bhadra, M-S. Abadir, L-C. Wang, and S. Ray. A survey of hybrid techniques for functional verification. *IEEE Design and Test of Computers*, 24 :112–122, 2007.

[bluebook] Michael Fingeroff. *High-Level Synthesis Blue Book*. Xlibris Corporation, 2010.

[Bo19] Maxence Bouvier, Alexandre Valentian, Thomas Mesquida, Francois Rummens, Marina Reyboz, Elisa Vianello, and Edith Beigne. Spiking Neural Networks Hardware Implementations and Challenges : A Survey. *Journal of Emerging Technologies in Computing Systems*, 15(2), 2019.

[Boh15] R. G. Bohorquez, C. A. Nour, , and C. Douillard. Precoding techniques for turbo codes. In *Proceedings of the European Wireless Conference*, Budapest, Hungary, May 2015.

[Bor07] J. Bortolami. Leveraging system models for rtl functional verification. http ://www.eetimes.com, 2007.

[Boy11] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1), 2011.

[Bre15] Matthew F. Brejza, Liang Li, Robert G. Maunder, Bashir M. Al-Hashimi, Claude Berrou, and Lajos Hanzo. 20 years of turbo coding and energy-aware design guidelines for energy-constrained wireless applications. *IEEE Communications Surveys & Tutorials*, 18(1) :8–28, June 2015.

[Can12] A. Canis, J. Choi, M. Aldham, V. Zhang, A. Kammoona, T. Czajkowski, S. Brown, and J.H. Anderson. Legup : An open source high-level synthesis tool for fpga-based processor/accelerator systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 1(1) :1–25, May 2012.

[Cas11] J. Castiñeira Moreira, M. Rabini, C. González, C. Gayoso, and L. Arnone. Fpga implementation of two very low complexity ldpc decoders. In *Proceedings of the Southern Conference on Programmable Logic (SPL)*, pages 7–12, 2011.

[Catapult-HLS] Mentor Graphics Corporation. *Catapult High-Level Synthesis - Datasheet*, 2017.

[Caul6] A. Caulfield et al. A cloud-scale acceleration architecture. In *Proceedings of the 49th Annual IEEE/ACM International Symposium on Microarchitecture*, October 2016.

[Cha00] R. Chapman and T.S. Durrani. IP protection of DSP algorithms for system on chip implementation. *IEEE Transactions on Signal Processing*, 48(3) :854–861, 2000.

[Cha10a] Y.-L. Chang and al. High-throughput GPU-based LDPC decoding. In *Proceedings of the SPIE Satellite Data Compression, Communications and Processing conference*, volume 7810, 2010.

[Che02b] J. Chen and M.P.C. Fossorier. Density evolution for two improved BP-based decoding algorithms of LDPC codes. *IEEE Communications Letters*, 6(5) :208–210, 2002.

[Che05] Jinghu Chen, A. Dholakia, E. Eleftheriou, and M.P.C. Fossorier and ; Xiao-Yu Hu. Reduced-complexity decoding of ldpc codes. *IEEE Transactions on Communications*, 53(8) :1288–1299, August 2005.

[Che15] Aleksandra Checko, Henrik L. Christiansen, Ying Yan, Lara Scolari, Georgios Kardaras, Michael S. Berger, and Lars Dittmann. Cloud RAN for mobile networks - a technology overview. *IEEE Communications Surveys & Tutorials*, 17(1) :405–426, September 2015.

[Con00] G.A. Constantinides, P.Y.K. Cheung, and W. Luk. Optimal datapath allocation for multiple-wordlength systems. *Electronics Letters*, 36(17) :1508–1509, August 2000.

[Con01] G.A. Constantinides, P.Y.K. Cheung, and W. Luk. Heuristic datapath allocation for multiple wordlength systems. In *Proceedings of the Design, Automation and Test in Europe (DATE)*, pages 791–796, Munich, Germany, March 2001.

[Con05] J. Cong, Yiping Fan, Guoling Han, Yizhou Lin, Junjuan Xu, Zhiru Zhang, and Xu Cheng. Bitwidth-aware scheduling and binding in high-level synthesis. In *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 856–861, Shanghai, China, January 2005.

[Cou08] Philippe Coussy, Cyrille Chavet, Pierre Bomel, Dominique Heller, Eric Senn, and Eric Martin. *High-Level Synthesis*, chapter GAUT : A High-Level Synthesis Tool for DSP Applications. Springer, Dordrecht, 2008.

[Dar14] Mickael Dardaillon. *Compilation d'applications flot de données paramétriques pour MPSoC dédiés à la radio logicielle*. PhD thesis, École doctorale en Informatique et Mathématiques de Lyon, Lyon, INSA, November 2014.

[Div05] D. Divsalar, C. Jones, S. Dolinar, and J. Thorpe. Protograph based ldpc codes with minimum distance linearly growing with block size. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, volume 3, 2005.

[Div06] D. Divsalar, S. Dolinar, and C. Jones. Construction of protograph LDPC codes with linear minimum distance. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 664–668, 2006.

[DVB-RCS] ETSI. *Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems (ETSI EN 301 790)* :, May 2009.

[DVB-S2] European Standard (Telecommunications series), ETSI EN 302 307 V1.2.1. *Digital Video Broadcasting (DVB). Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications (DVB-S2).*, August 2009.

[Ell00] J. P. Elliott. *Understanding Behavioral Synthesis. A Practical Guide to High-Level Design.* Kluwer Academic Publishers, 2000.

[eXC13] Y Explorations (YXI). *eXCite C to RTL Behavioral Synthesis 4.1(a)*, 2010.

[Fal08] G. Falcao, V. Silva, L. Sousa, and J. Marinho. High coded data rate and multicodeword WiMAX LDPC decoding on the Cell/BE. *Electronics Letters*, 44, 2008.

[Fal09a] G. Falcao, V. Silva, J. Marinho, and L. Sousa. Ldpc decoders for the wimax (ieee 802.16e) based on multicore architectures. *in WIMAX New Developments. Upena D Dalal and Y P Kosta (Ed.)*, 2009.

[Fal11a] G. Falcao, J. Andrade, V. Silva, and L. Sousa. GPU-based DVB-S2 LDPC decoder with high throughput and fast error floor detection. *IET Electronics Letters*, 47(9) :542–543, April 2011.

[Fal11b] G. Falcao, L. Sousa, and V. Silva. Massively LDPC decoding on multicore architectures. *IEEE Transactions on Parallel and Distributed Systems*, 22(2) :309–322, 2011.

[Fal12a] Gabriel Falcao, Vitor Silva, Leonel Sousa, and Joao Andrade. Portable LDPC decoding on multicores using OpenCL. *IEEE Signal Processing Magazine*, 29(4) :81–88, 2012.

[Fal14] J. Andrade, G. Falcao, , and V. Silva. Flexible design of wide-pipeline based WiMAX QC-LDPC decoder architectures on FPGAs using high-level synthesis. *IET Electronics Letters*, 50, 2014.

[Fan03] Y.C. Fan and H.W. Tsao. Watermarking for intellectual property protection. *IEEE Electronics Letters*, 39(18) :1316–1318, 2003.

[Fel03] J. Feldman. *Decoding Error-Correcting Codes via Linear Programming.* PhD thesis, Massachussets Institute of Technology, 2003.

[Fel05] J. Feldman, M.J. Wainwright, and D.R. Karger. Using linear programming to decode binary linear codes. *IEEE Transactions on Information Theory*, 51(3) :954–972, March 2005.

[Fos03] Krolnik A. Foster H., Lacey D. *Assertion-Based Design.* Kluwer Academic Publishers, 2nd ed. edition, 2003.

[Fos98] M. P. C. Fossorier and S. Lin. Soft-input soft-output decoding of linear block codes based on ordered statistics,. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, pages 2828–2833, Sydney, Australia, November 1998.

[Fos99] M.P.C. Fossorier, M. Mihaljevic, and H. Imai. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation. *IEEE Transactions on Communications*, 47(5) :673–680, May 1999.

[Gal62] R. Gallager. Low-density parity-check codes. *IRE Transactions on Information Theory*, 8(1) :21–28, 1962.

[Garz16] R. Garzon Bohorquez, C. Abdel Nour, and C. Douillard. Improving turbo codes for 5G with parity puncture-constrained interleavers. In *Proceedings of the International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Brest, France, September 2016.

[Garz18] R. Garzon-Bohorquez, R. Klaimi, C. Abdel Nour, and C. Douillard. Mitigating correlation problems in turbo decoders. In *Proceedings of the International Symposium on Turbo Codes & Iterative Information Processing (ISTC)*, Hong Kong, China, December 2018.

[Gen10] G. Gentile, M. Rovini, and L. Fanucci. A multi-standard flexible turbo/ldpc decoder via asic design. In *Proceedings of the 6th International Symposium on Turbo Codes and Iterative Information Processing*, (ISTC), pages 294-298, Brest, France, 6-10 September 2010.

[Gia13] P. Giard, G. Sarkis, C. Thibeault, and W.J. Gross. A fast software polar decoder. *arXiv :1306.6311*, June 2013.

[Gia14a] P. Giard, G. Sarkis, C. Thibeault, and W.J. Gross. Fast software polar decoders. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7555-7559, May 2014.

[Gra19] Eugene Grayver. Scaling the fast x86 DVB-S2 decoder to 1 Gbps. In *Proceedgins of the IEEE Aerospace Conference*, pages 1-9, 2019.

[Gro12a] Stefan Gronroos, Kristian Nybom, and Jerker Bjorkqvist. Efficient GPU and CPU-based LDPC decoders for long codewords. *Journal of Analog Integrated Circuits and Signal Processing*, 73 :583—595, 2012.

[Gui07a] F. Guilloud, E. Boutillon, J. Tousch, and J.-L. Danger. Generic description and synthesis of LDPC decoders. *IEEE Transactions on Communications*, 55(11) :2084-2091, 2007.

[Hai16] Peter Hailes, Lei Xu, Robert G. Maunder, Bashir M. Al-Hashimi, and Lajos Hanzo. A survey of FPGA-based LDPC decoders. *IEEE Communications Surveys & Tutorials*, 18(2) :1098-1122, 2016.

[Har81] P.G. Harrison. Efficient table-driven implementation of the finite state machine. *Journal of Systems and Software*, 2(3) :201-211, 1981.

[Hoc04] D. Hocevar. A reduced complexity decoder architecture via layered decoding of LDPC codes. In *Proceedings of the IEEE Workshop on Signal Processing Systems (SiPS)*, pages 107-112, October 2004.

[Hon99] I. Hong and M. potkonjak. Behavioral synthesis techniques for intellectual property protection. In *Proceedings of the 36th ACM/IEEE Design Automation Conference*, pages 849-854, June 1999.

[Intel-HLS] intel. *Intel High Level Synthesis Compiler Pro Edition - Reference Manual (v22.2)*, June 2022.

[Jai03] A.K. Jain, L. Yuan, P.R. Pari, and Gang Qu. Zero overhead watermarking technique for FPGA designs. In *Proceedings of the 13th ACM Great Lakes symposium on VLSI*, pages 147-152, 2003.

[Jil1a] H. Ji, J. Cho, and W. Sung. Memory access optimized implementation of cyclic and quasi-cyclic LDPC codes on a GPGPU. *Journal of Signal Processing Systems*, 64 :149-159, 2011.

[Jia15] X. Jiao, H. Wei, J. Mu, and C. Chen. Improved admm penalized decoder for irregular low-density parity-check codes. *IEEE Communications Letters,*, 19(6) :913-916, June 2015.

[Jia17] X. Jiao, J. Mu, and H. Wei. Reduced complexity node-wise scheduling of admm decoding for ldpc codes. *IEEE Communications Letters*, 21(3) :472–475, March 2017.

[Jin00] Hui Jin, Aamod Khandekar, and Robert McEliece. Irregular repeat-accumulate codes. In *Proceedings of the 2nd International Symposium on Turbo codes and related topics (ISTC)*, pages 1–8, 2000.

[Kan12a] Soonyoung Kang and Jaekyun Moon. Parallel LDPC decoder implementation on GPU based on unbalanced memory coalescing. In *Proceedings of the IEEE Conference on Communications*, pages 3692–3697, 2012.

[Kir98] D. Kirovski, Y. Y. Hwang, M. Potkonjak, and J. Cong. Intellectual property protection by watermarking combinational logic synthesis solutions. In *Proceedings of the IEEE/ACM international conference on Computer-aided design (ICCAD)*, pages 194–198, San Jose, CA, USA, November 1998.

[Kou01] Y. Kou, S. Lin, and M. P. C. Fossorier. Low-density parity-check codes based on finite geometries : a rediscovery and new results. In *IEEE Transactions on Information Theory*, volume 47, pages 2711–2736. PIGN83, 2001.

[Kum98] K.-I. Kum and W. Sung. Word-length optimization for high-level synthesis of digital signal processing systems. In *Proceedings of the IEEE Workshop on Signal Processing Systems (SIPS)*, pages 569–578, Cambridge, MA, USA, October 1998.

[Kun09] Gupta R. Kundu S., Lerner S. High-level verification. *IPSJ Transactions on System LSI Design Methodology*, 2(131–144), 2009.

[Lac99] John Lach, William H. Mangione-Smith, and Miodrag Potkonjak. Robust fpga intellectual property protection through multiple small watermarks. In *Proceedings of the 36th Annual ACM/IEEE Design Automation Conference (DAC)*, pages 831–836, June 1999.

[Lee06] D.-U. Lee, A.A. Gaffar, R.C.C. Cheung, O. Mencer, W. Luk, and G.A. Constantinides. Accuracy-guaranteed bit-width optimization. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(10) :1990–2000, 2006.

[Leo19] Mathieu Léonardon, Adrien Cassagne, Camille Leroux, Christophe Jégo, Louis-Philippe Hamelin, and Yvon Savaria. Fast and flexible software polar list decoders. *Journal of Signal Processing Systems*, 91(8) :937–952, August 2019.

[Leon3] Gaisler Research. *GRLIB IP Library User's Manual*, 2010.

[LeR12a] C. Leroux, A. Raymond, G. Sarkis, and W. Gross. A semi-parallel successive-cancellation decoder for polar codes. *Signal Processing, IEEE Transactions on*, 2012.

[LeR12b] C. Leroux, A. J. Raymond, G. Sarkis, I. Tal, A. Vardy, and W. J. Gross. Hardware implementation of successive-cancellation decoders for polar codes. *Journal of Signal Processing Systems*, 69(3) :305–315, 2012.

[Lil3a] Rongchun Li and al. A multi-standard efficient column-layered LDPC decoder for software defined radio on GPUs. In *Proceedings of the SPAWC Workshop*, pages 724–728, 2013.

[lil4] Rongchun Li, Yong Dou, Jiaqing Xu, Xin Niu, and Shice Ni. An efficient parallel SOVA-based turbo decoder for software defined radio on GPU. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 97(5) :1027–1036, 2014.

[Lin14] Y. Lin and W. Niu. High throughput LDPC decoder on GPU. *IEEE Communications Letters*, 18(2) :344–347, February 2014.

[Liu12] X. Liu, S. Draper, , and B. Recht. Suppressing pseudocodewords by penalizing the objective of lp decoding,. In *Proceedings of the IEEE Information Theory Workshop (ITW)*, pages 367–371, September 2012.

[liu13] Chengjun Liu, Zhisong Bie, Canfeng Chen, and Xianjun Jiao. A parallel LTE turbo decoder on GPU. In *Proceedgins of the 15th IEEE International Conference on Communication Technology Conference (ICCT)*, pages 609–614, November 2013.

[Liu16] Xishuo Liu and Stark C. Draper. The admm penalized decoder for ldpc codes. *IEEE Transactions on Information Theory*, 62(6) :2966–2984, June 2016.

[Liu18] Jian-Cheng Liu, Huan-Chun Wang, Chung-An Shen, and Jih-Wei Lee. Low-complexity LDPC decoder for 5G URLLC. In *Proceedings of the IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia)*, pages 43–46, 2018.

[Lui14] Xishuo Liu and Stark C. Draper. Admm decoding of non-binary LDPC codes in f2m. In *2014 IEEE International Symposium on Information Theory*, pages 2449–2453, June 2014.

[Mac97] D.J.C. MacKay and R.M. Neal. Near shannon limit performance of low density parity check codes. *Electronics Letters*, 6(3) :457–458, March 1997.

[Man20] Paolo Mantovani, Robert Margelli, Davide Giri, and Luca P. Carloni. HL5 : A 32-bit RISC-V processor designed with high-level synthesis. In *2020 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–8, 2020.

[Mar09] Cedric Marchand, Jean-Baptiste Dore, Laura Conde-Canencia, and Emmanuel Boutillon. Conflict resolution for pipelined layered LDPC decoders. In *Proceedgins of the IEEE Workshop on Signal Processing Systems (SiPS)*, pages 220–225, 2009.

[Mar10] Cédric Marchand. *Implementation of an LDPC decoder for the DVB-S2, -T2 and -C2 standards*. PhD thesis, Université de Bretagne Sud, Janvier 2010.

[Mar11a] C. Marchand, L. Conde-Canencia, and E. Boutillon. Architecture and finite precision optimization for layered LDPC decoders. *Journal of Signal Processing Systems*, 65, 2011.

[Mar15] C. Marchand and E. Boutillon. LDPC decoder architecture for DVB-S2 and DVB-S2X standards. In *Proceedings of the IEEE Workshop On Signal Processing Systems (SiPS)*, pages 1–5, 2015.

[Mar09] Grant Martin and Gary Smith. High-level synthesis : Past, present, and future. *IEEE Design & Test of Computers*, 26(4) :18–25, July 2009.

[Mau16] Robert Maunder. *Survey of ASIC implementations of LDPC decoders*. University of Southampton, 2016.

[Mey92] B. Meyer. Applying "design by contract". *Computer IEEE*, 25 :40–51, 1992.

[Mig22] Farhad Mighani, Michael Sharp, , Mike McNamara, and Dave Pursley. *Using High-Level Synthesis to Design and Verify 802.11ah Baseband IP*. Adapt-IP, 2022.

[Mit98] Michael Mitzenmacher. A note on low density parity check codes for erasures and errors. Technical report, COMPACT, Systems Research Center, December 1998.

[Mul06] O. Muller, A. Baghdadi, and M. Jezequel. Exploring parallel processing levels for convolutional turbo decoding. In *Proceedings of the International Conference on Information & Communication Technologies*, volume 2, pages 2353–2358, April 2006.

[Mur11] Purushotham Murugappa, Rachid Al-Khayat, Amer Baghdadi, and Michel Jezequel. A flexible high throughput multi-ASIP architecture for LDPC and turbo decoding. In *Proceedgins of the Design, Automation & Test in Europe Conference (DATE)*, 2011.

[Myu05] Seho Myung, Kyeongcheol Yang, and Jaeyoel Kim. Quasi-cyclic LDPC codes for fast encoding. *IEEE Transactions on Information Theory*, 51(8) :2894–2901, 2005.

[Nae08] F. Naessens, B. Bougard, S. Bressinck, L. Hollevoet, P. Raghavan, L. Van der Perre, and F. Catthoor. A unified instruction set programmable architecture for multi-standard advanced forward error correction. In *Proceedings of the IEEE Workshop on Signal Processing Systems*, (SiPS), pages 31–36, October 2008.

[Nar97] K. R. Narayanan and G. L. Stuber. Selective serial concatenation of turbo codes. 1(5) :136–139, September 1997.

[Nar98] K. R. Narayanan and G. L. Stuber. List decoding of turbo codes. 46(6) :754–762, June 1998.

[Nay01] A. Nayak, M. Haldar, A. Choudhary, and P. Banerjee. Precision and error analysis of matlab applications during automated hardware synthesis for fpgas. In *Proceedings Design, Automation and Test in Europe. Conference and Exhibition 2001*, pages 722–728, 2001.

[Oli01] A. Oliveira. Techniques for the creation of digital watermarks in sequential circuits designs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 20(9) :1101–1117, 2001.

[Orange] Consultative Committee for Space Data Systems (CCSDS). *CCSDS 231.1-O-1 - ORANGE BOOK - SHORT BLOCK LENGTH LDPC CODES FOR TC SYNCHRONIZATION AND CHANNEL CODING*. Washington, DC, USA, orange book edition, April 2015.

[Oth19] Rami Othman. *Study of reception techniques for aeronautical telemetry modulations*. PhD thesis, CentraleSupélec, Rennes, January 2019.

[Oul06] Y. Ould-Cheikh-Mouhamedou, S. Crozier, K. Gracie, P. Guinand, and P. Kabal. A method for lowering turbo code error flare using correction impulses and repeated decoding,. In *Proceedings of the International Symposium on Turbo Codes and Related Topics (ISTC)*, Munich, Germany, April 2006.

[Oul07] Y. Ould-Cheikh-Mouhamedou and S. Crozier. Improving the error rate performance of turbo codes using the forced symbol method. 11(7) :616–618, July 2007.

[Ped13] Volnei A. Pedroni. *Finite State Machines in Hardware : Theory and Design (with VHDL and SystemVerilog)*. The MIT Press, December 2013.

[Pfl13] S. Pfletschinger and M. Navarro. Enhanced turbo decoding for error floor reduction. In *Proceedings of the International Conference on Systems, Communication and Coding (SCC)*, Munich, Germany, January 2013.

[Plasma] Steve Rhoads. Plasma 32-bit softcore. Technical report, www.plasmacpu.no-ip.org, 2011.

[Ras14] O. A. Rasheed, P. Ivaniš, and B. Vasi. Fault-tolerant probabilistic gradient-descent bit flipping decoder. *IEEE Communications Letters*, 18(9) :1487–1490, 2014.

[Ras99] A. Rashid, J. Asher, W.H. Mangione-Smith, and M. Potkonjak. Hierarchical watermarking for protection of dsp filter cores. In *Proceedings of the IEEE Custom Integrated Circuits Conference*, pages 39–42, May 1999.

[Rei01] A. C. Reid, T. A. Gulliver, and D. P. Taylor. Convergence and errors in turbo-decoding. *IEEE Transactions on Communications*, 49(12) :2045–2051, December 2001.

[Res05] M. Reshadi, B. Gorjiara, and D. Gajski. Utilizing horizontal and vertical parallelism with a no-instruction-set compiler for custom datapaths. In *Proceedgins of the IEEE International Conference on Computer Design : VLSI in Computers and Processors (ICCD)*, pages 69–74, 2005.

[Ric01] T. J. Richardson and R. L. Urbanke. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Transactions on Information Theory*, 47 :599–618, 2001.

[Roh16] Si-Dong Roh, Keol Cho, and Ki-Seok Chung. Implementation of an LDPC decoder on a heterogeneous FPGA-CPU platform using SDSoC. In *Proc. of TENCON*, 2016.

[Rok19] Simon Rokicki, Davide Pala, Joseph Paturel, and Olivier Sentieys. What you simulate is what you synthesize : Designing a processor core from C++ specifications. In *Proceeding of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–8, Westminster, CO, USA, November 2019.

[Ros11] E. Rosnes and A. G. i Amat. Performance analysis of 3-d turbo codes. In *IEEE Transactions on Information Theory*, volume 57, pages 3707–3720, June 2011.

[Rov07] M. Rovini, G. Gentile, F. Rossi, and L. Fanucci. A scalable decoder architecture for IEEE 802.11n LDPC codes. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, pages 3270–3274, November 2007.

[Sad97] J. S. Sadowsky. A maximum likelihood decoding algorithm for turbo codes. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, pages 929–933, Phoenix, USA, November 1997.

[Sai22] Kassem Saied. *Quasi Cyclic Short Packet (QCSP) transmission for Internet of Things*. PhD thesis, Université de Bretagne Sud, March 2022.

[Sar16a] G. Sarkis, P. Giard, A. Vardy, C. Thibeault, and W. J. Gross. Fast list decoders for polar codes. *IEEE Journal on Selected Areas in Communications - Special Issue on Recent Advances In Capacity Approaching Codes*, 34(2) :318–328, February 2016.

[Sav08] V. Savin. Self-corrected Min-Sum decoding of LDPC codes. In *Proceedings of the IEEE International Symposium on Information Theory*, pages 146–150, July 2008.

[Sch13a] E. Scheiber and et al. Implementation of an LDPC decoder for IEEE 802.11n using Vivado high-level synthesis. In *Proc. of ESPCO*, 2013.

[Sha07] E. Sharon, S. Litsyn, and J. Goldberger. Efficient serial message-passing schedules for ldpc decoding. *IEEE Transactions on Information Theory, vol., pp.,.*, 53(11) :4076–4091, November 2007.

[Sha21] Arash Shafiei. *RDF : un modèle de calcul flot de données reconfigurable*. PhD thesis, Université Grenoble Alpes, Grenoble, December 2021.

[Sha99] R. Y. Shao, S. Lin, , and M. P. C. Fossorier. Two simple stopping criteria for turbo decoding. *IEEE Transactions on Communications*, 47(8) :1117–1120, August 1999.

[Sim16] Osvaldo Simeone, Andreas Maeder, Mugen Peng, Onur Sahin, and Wei Yu. Cloud radio access network : Virtualizing wireless access for dense heterogeneous systems. *Journal of Communications and Networks*, 18(2) :135–149, 2016.

[SIMD] INTEL corp. *Intel Architecture Instruction Set Extensions and Future Features - Programming Reference*, June 2022.

[SPARCv8] SPARC International Inc. *The SPARC Architecture Manual*, ver. 8, 1999.

[Ste00] Mark Stephenson, Jonathan Babb, and Saman Amarasinghe. Bidwidth analysis with application to silicon compilation. In *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 108–120, 2000.

[STM21] STMicroelectronics. *STMicroelectronics quickly brings automotive image signal processing to market with High-Level Synthesis*, 2021.

[Sun06] G. Sun, Z. Gao, and Y. Xu. A watermarking system forIP protection by buffer insertion technique. In *Proceedings of the 7th International Symposium on Quality Electronic Design (ISQED)*, pages 671–675, San Jose, CA, March 2006.

[Sur19] Leonardo Suriano, Florian Arrestier, Alfonso Rodríguez, Julien Heulot, Karol Desnos, Maxime Pelcat, and Eduardo de la Torre. DAMHSE : programming heterogeneous MPSoCs with hardware acceleration using dataflow-based design space exploration and automated rapid prototyping. *Microprocessors and Microsystems, ELSEVIER*, 71, November 2019.

[Tak01] O. Y. Takeshita, O. M. Collins, P. C. Massey, and D. J. Costello. On the frame-error rate of concatenated turbo codes. *IEEE Transactions on Communications*, 49(4) :602–608, April 2001.

[Tal03] Sriraman Tallam and Rajiv Gupta. Bitwidth aware global register allocation. In *Proceedings of the ACM Symposium on Principles of Programming Languages*, pages 85—96, 2003.

[Tal15] Ido Tal and Alexander Vardy. List decoding of polar codes. *IEEE Transactions on Information Theory*, 61(5) :2213–2226, March 2015.

[Tan81] R. Tanner. A recursive approach to low complexity codes. *IEEE Transactions on Information Theory*, 27(5) :533–547, 1981.

[Thu06] Adrian D. Thurston. Parsing computer languages with an automaton compiled from a single regular expression. In Oscar H. Ibarra and Hsu-Chun Yen, editors, *Implementation and Application of Automata*, pages 285–286, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[Ton11] S. Tong, H. Zheng, and B. Bai. Precoded turbo code within 0.1 dB of shannon limit. *Electronics Letters*, 47(8) :521–522, April 2011.

[Tor00] I. Torunoglu and E. Charbon. Watermarking-based copyright protection of sequential functions. *IEEE Journal of Solid-State Circuits*, 35(3) :434–440, 2000.

[Una17] B. Unal, F. Ghaffari, A. Akoglu, D. Declercq, and B. Vasi. Analysis and implementation of resource efficient probabilistic Gallager B LDPC decoder. *Proceedings of the 15th IEEE International New Circuits and Systems Conference (NEWCAS)*, pages 333–336, 2017.

[Vitis-HLS] AMD XILINX. *Vitis High-Level Synthesis User Guide - UG1399 (v2022.1)*, June 2022.

[VivadoHLS] Xilinx. Vivado design suite user guide : High-level synthesis (ug902), May 2021.

[Wad10] T. Wadayama, K. Nakamura, M. Yagita, Y. Funahashi, S. Usami, and I. Takumi. Gradient descent bit flipping algorithms for decoding LDPC codes. *IEEE Transactions on Communications*, 58(6) :1610–1614, 2010.

[Wan11a] Guohui Wang, Michael Wu, Yang Sun, and Joseph R. Cavallaro. GPU accelerated scalable parallel decoding of LDPC codes. In *Proceedings of the IEEE Asilomar Conference on Signals, Systems, and Computers*, pages 2053–2057, 2011.

[Wan11c] G. Wang, M. Wu, Y. Sun, and J. R. Cavallaro. A massively parallel implementation of QC-LDPC decoder on GPU. In *Proceedings of the Symposium on Application Specific Processors*, pages 82–85, 2011.

[Wan13] Guohui Wang, Michael Wu, Bei Yin, and Joseph R. Cavallaro. High throughput low latency LDPC decoding on GPU for SDR systems. In *Proceedings of the IEEE GlobalSIP Conference*, pages 1258–1261, 2013.

[Was15] Mitchell Wasson and Stark C. Draper. Hardware based projection onto the parity polytope and probability simplex. In *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, Grove, CA, USA, November 2015.

[Was17] Mitchell Wasson, Mario Milicevic, Stark C. Draper, and Glenn Gulak. Hardware-based linear programming decoding via the alternating direction method of multipliers. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1143–1147, 2017.

[Weh16] Florian Gensheimer, Stefan Ruzika, Stefan Scholl, and Norbert Wehn. ADMM versus simplex algorithm for LP decoding. In *Proceedings of the International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Brest, France, October 2016.

[Wei13] Y. Wei, M. Jiang, B. Xia, W. Chen, , and Y. Yang. A CRC-aided hybrid decoding algorithm for turbo codes. *IEEE Wireless Communications Letters*, 2(5) :471–474, October 2013.

[Wei18] Haoyuan Wei and Amir H. Banihashemi. An iterative check polytope projection algorithm for admm-based lp decoding of ldpc codes. *IEEE Communications Letters*, 22(1) :29–32, 2018.

[Wu15] Jun Wu, Zhifeng Zhang, Yu Hong, and Yonggang Wen. Cloud radio access network (c-ran) : a primer. *IEEE Network*, 29(1) :35–41, 2015.

[Xia13] Jiao Xianjun, Chen Canfeng, P. Jaaskelainen, V. Guzma, and H. Berg. A 122mb/s turbo decoder using a mid-range GPU. In *IWCMC proc.*, pages 1090–1094, July 2013.

[Xu12] Y. Xu, M. Jiang, M. Ding, and Y. Yang. An efficient OSD-aided iterative decoding algorithm for LTE turbo codes. In *Proceedings of the International Conference on Wireless Communications Signal Processing (WCSP)*, Huangshan, China, October 2012.

[Zha05] Jianguang Zhao, F. Zarkeshvari, and A.H. Banihashemi. On implementation of min-sum algorithm and its modifications for decoding low-density parity-check (LDPC) codes. *IEEE Transactions on Communications*, 53(4) :549–554, April 2005.

[Zha09] Xiaojun Zhang, Yinghong Tian, Jianming Cui, Yuyin Xu, and Zongsheng Lai. An multi-rate LDPC decoder based on ASIP for DMB-TH. In *Proceedgins of the 8th IEEE International Conference on ASIC*, pages 995–998, 2009.

[Zhal2b] X. Zhang and P. Siegel. Adaptive cut generation algorithm for improved linear programming decoding of binary linear codes. *IEEE Transactions on Information Theory*, 58 :6581–6594, June 2012.

[Zhal3b] X. Zhang and P. H.Siegela. Efficient iterative lp decoding of LDPC codes with alternating direction method of multipliers. In *Proceedgins of the IEEE International Symposium on Information Theory (ISIT)*, July 2013.