



INSTITUT  
POLYTECHNIQUE  
DE PARIS

NNT : 2022IPPAT020

Thèse de doctorat

TELECOM  
Paris



IP PARIS

# Conception et intégration d'un convertisseur analogique-paramètres flexible pour les capteurs intelligents

Thèse de doctorat de l'Institut Polytechnique de Paris  
préparée à Télécom Paris

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (EDIPP)  
Spécialité de doctorat : Électronique et optoélectronique

Thèse présentée et soutenue à Palaiseau, le 7 février 2022, par

**ANTOINE BACK**

Composition du Jury :

Patrick Garda Professeur émérite, Sorbonne Université (LIP6)	Président
Manel Ben-Romdhane Assistant professor, SUP'COM, University of Carthage	Rapporteure
Cyril Lahuec Associate Professor, IMT Atlantique (Lab-STICC)	Rapporteur
Dominique Dallet Professeur, Bordeaux INP (IMS)	Examineur
Michael Pelissier Ingénieur de recherche, Université Grenoble-Alpes (CEA-Leti)	Examineur
Patricia Desgreys Professeure, Télécom Paris (LTCl)	Directeur de thèse
Paul Chollet Maître de conférences, Télécom Paris (LTCl)	Invité
Olivier Fercoq Maître de conférences, Télécom Paris (LTCl)	Invité



# Résumé

Avec le fort développement de l'Internet des Objets (IoT), il devient nécessaire de converger vers de nouveaux capteurs dit intelligents. Ces capteurs doivent permettre d'analyser l'environnement extérieur, comprendre le contexte dans lequel ils sont utilisés et être conscient des besoins utilisateurs. Ils doivent cependant rester petits, fiables, bon marché et avoir une autonomie de plusieurs années. La conversion analogique-paramètre (Analog-to-Feature, A2F) est une nouvelle méthode d'acquisition pensée pour les appareils IoT, et semble être une solution adaptée pour de tels capteurs. Cette conversion consiste à extraire des paramètres directement sur le signal analogique. Une sélection pertinente des paramètres permet d'extraire uniquement l'information nécessaire à une tâche particulière. Le convertisseur proposé est basé sur la technique de l'échantillonnage non-uniforme en ondelettes (NUWS). L'architecture mélange le signal analogique avec des ondelettes paramétrables avant d'intégrer et convertir le signal en données numériques. L'objectif de la thèse est de proposer une méthode pour concevoir un convertisseur A2F générique basé sur le NUWS. Il est ainsi nécessaire de définir les caractéristiques des ondelettes afin d'acquérir une large gamme de signaux basse fréquence (ECG, EMG, EEG, parole...). Cette étape nécessite l'utilisation d'algorithmes de sélection de paramètres et d'algorithmes d'apprentissage automatique pour sélectionner le meilleur ensemble d'ondelettes pour une application donnée et qui doit permettre de définir les spécifications du convertisseur. L'étape de sélection des paramètres doit tenir compte des contraintes de mise en œuvre pour optimiser au mieux la consommation d'énergie. Un algorithme de sélection de paramètres est proposé pour choisir des ondelettes pour une application donnée, afin de maximiser la précision de classification tout en diminuant la consommation d'énergie, grâce à un modèle de consommation réalisé dans une technologie CMOS 0.18  $\mu\text{m}$ .

**Mots-clés :** Convertisseur analogique-paramètre, Sélection de paramètres, Apprentissage automatique, Échantillonnage non-uniforme en ondelettes, Capteurs intelligents.



# Abstract

The Internet of Things (IoT) is currently experiencing huge developments. IoT includes lots of different devices such as Wireless Sensors Networks (WSN) or wearable electronics that rely on wireless communications. These networks need to understand the context in which they are used. This means that the system must know what is happening around it, i.e. sense the environment, and understands the needs of the user. This requires always-on sensing on many sensors while being small, cheap, reliable and having a lifetime of several years. Analog-to-Feature (A2F) conversion is a new acquisition method that was thought for IoT devices. The converter aims at extracting useful features directly on the analog signal. By carefully choosing a set of features, it is possible to acquire only the relevant information for a given task. The proposed converter is based on the Non-Uniform Wavelet Sampling (NUWS) architecture. The architecture mixes the analog signal with tunable wavelets prior to integration and digital conversion. The aim of the thesis is to propose a method to design a generic A2F converter based on the NUWS. It includes the definition of the wavelet parameters in order to acquire a broad range of low frequency signals (ECG, EMG, EEG, speech ...). This step requires the use of feature selection algorithms and machine learning algorithms for selecting the best set of wavelets for a given application and should be used to define the specifications for the converter. The feature selection step must be aware of physical implementation constraints to optimize energy consumption as much as possible. A feature selection algorithm is proposed to choose wavelets for a given application, in order to maximize classification accuracy while decreasing power consumption, through a power model designed in 0.18  $\mu\text{m}$  CMOS technology.

**Keywords:** Analog-to-feature converter, Feature selection, Machine Learning, Non-Uniform Wavelet Sampling, Smart sensors.



# Remerciements

Tout d’abord, je tiens à remercier ma directrice de thèse, Patricia DESGREYS, et mes encadrants, Paul CHOLLET et Olivier FERCOQ, pour leurs précieux conseils et pour m’avoir accompagné afin de mener à bien ces travaux de thèse, malgré les incertitudes liées au contexte sanitaire. Merci d’avoir cru en moi et de m’avoir offert cette belle opportunité.

Je souhaite également remercier l’ensemble de mon jury de thèse, avec qui j’ai pris plaisir à échanger lors de ma soutenance, pour la bienveillance dont ils ont fait preuve. Je remercie tout d’abord Madame Manel BEN-ROMDHANE et Monsieur Cyril LAHUEC qui ont accepté d’être rapporteurs de cette thèse, mais aussi Messieurs Patrick GARDA, Dominique DALLET et Michael PELISSIER qui ont accepté de participer à mon jury de thèse.

Je remercie également l’ensemble du personnel du département COMELEC de Télécom Paris et plus particulièrement les membres de l’équipe C2S. Merci à Germain, Chadi et Yves pour l’aide qu’ils ont pu m’apporter lors de la réalisation de ces travaux et la rédaction de ce manuscrit. Cela a toujours été un réel plaisir d’échanger avec vous.

Je remercie évidemment mes parents, ma famille et mes proches pour le soutien sans faille dont ils ont fait preuve lors de ces trois années. Merci à vous d’avoir été présent durant les moments difficiles, je n’aurais pas pu réussir sans votre présence.

Un grand merci également à tous les doctorants avec qui j’ai partagé cette aventure, en particulier Arthur, Lyliya, Zülal, Clément et Mirko, pour leurs conseils et leur amitié.





# Table des matières

<b>Résumé</b>	<b>I</b>
<b>Abstrat</b>	<b>III</b>
<b>Remerciements</b>	<b>V</b>
<b>Table des matières</b>	<b>VI</b>
<b>Table des figures</b>	<b>XI</b>
<b>Liste des tableaux</b>	<b>XV</b>
<b>Liste des acronymes</b>	<b>XVII</b>
<b>Publications et communications</b>	<b>XXI</b>
<b>Introduction</b>	<b>1</b>
<b>1 Échantillonnage et traitement de l'information</b>	<b>5</b>
1.1 Introduction sur les capteurs sans-fil . . . . .	5
1.2 Approche classique des capteurs sans fil . . . . .	7
1.2.1 Capteur et conditionnement . . . . .	7
1.2.2 Échantillonnage de Nyquist . . . . .	9
1.2.3 Les architectures des convertisseurs analogiques numériques . . . . .	11
1.2.3.1 CAN Flash . . . . .	12
1.2.3.2 CAN pipeline . . . . .	13
1.2.3.3 CAN à approximation successive . . . . .	14
1.2.3.4 CAN Sigma-Delta ( $\Sigma\Delta$ ) . . . . .	16
1.2.3.5 Conclusion sur les CAN . . . . .	18
1.2.4 Transmission de l'information . . . . .	19
1.2.5 Discussion . . . . .	20
1.3 Échantillonnage compressif : conversion Analogique vers Information . . . . .	21
1.3.1 Principe d'échantillonnage compressif . . . . .	21

1.3.2	Architecture du convertisseur A2I . . . . .	24
1.3.2.1	Échantillonnage non uniforme (NUS) . . . . .	25
1.3.2.2	Démodulateur aléatoire . . . . .	25
1.3.2.3	Convertisseur large bande modulé . . . . .	27
1.3.2.4	Échantillonnage non-uniforme en ondelettes . . . . .	28
1.3.3	Algorithmes de reconstruction . . . . .	30
1.3.3.1	Algorithmes d'optimisation convexe . . . . .	31
1.3.3.2	Algorithmes Gloutons . . . . .	31
1.3.3.3	Comparaison des algorithmes de reconstruction . . . . .	32
1.3.4	Conclusion sur les convertisseurs A2I . . . . .	33
1.3.5	Discussion sur les convertisseurs A2I . . . . .	34
1.4	Algorithmes de classification . . . . .	36
1.4.1	Principe de la classification . . . . .	37
1.4.2	Mise en œuvre d'algorithmes de classification . . . . .	39
1.4.2.1	Arbre de décision et forêt aléatoire . . . . .	39
1.4.2.2	Réseaux de neurones artificiels . . . . .	40
1.4.2.3	Classifieur Bayésien . . . . .	42
1.4.2.4	K plus proches voisins (KNN) . . . . .	43
1.4.2.5	Support Vector Machine (SVM) . . . . .	43
1.4.2.6	Régression logistique . . . . .	46
1.4.3	Conclusion sur les algorithmes de classification . . . . .	47
1.5	Conclusion . . . . .	47
<b>2</b>	<b>Extraction de paramètres par échantillonnage non-uniforme en ondelettes pour la détection d'anomalies d'un signal ECG</b>	<b>49</b>
	Introduction . . . . .	49
2.1	Convertisseur Analogique-Paramètre (A2F) basé sur le NUWS . . . . .	50
2.1.1	Principe du convertisseur analogique-paramètre . . . . .	50
2.1.2	Introduction sur les signaux ECG . . . . .	53
2.1.3	Échantillonnage non-uniforme en ondelettes . . . . .	55
2.1.3.1	Rappels sur la transformée en ondelettes . . . . .	55
2.1.3.2	Dictionnaires d'ondelettes . . . . .	58
2.2	Algorithmes de sélection de paramètres . . . . .	61
2.2.1	Méthodes par filtrage . . . . .	62
2.2.1.1	Gain d'information . . . . .	62
2.2.1.2	Score de Fisher . . . . .	63
2.2.1.3	Relief - Relief-F . . . . .	63
2.2.2	Méthodes par encapsulation . . . . .	65
2.2.2.1	Algorithmes de recherche séquentielle . . . . .	65
2.2.2.2	Algorithme génétique . . . . .	66

2.2.3	Conclusion sur les algorithmes de sélection de paramètres . . . . .	68
2.3	Sélection de paramètres pour le NUWS . . . . .	68
2.3.1	Comparaison des méthodes de sélection et de différents types d'ondelettes	69
2.3.2	Synchronisation du complexe QRS . . . . .	71
2.3.3	Niveau de quantification . . . . .	77
2.3.4	Reconnaissance d'anomalie . . . . .	78
2.4	Conclusion . . . . .	79
<b>3</b>	<b>Évaluation et optimisation de la consommation d'un convertisseur analogique- paramètre</b>	<b>83</b>
	Introduction . . . . .	83
3.1	Sélection adaptée à l'architecture . . . . .	84
3.2	Modèle de consommation du circuit analogique . . . . .	86
3.2.1	Frontal analogique . . . . .	87
3.2.2	Mixeur analogique . . . . .	88
3.2.3	Intégrateur . . . . .	88
3.2.4	Bilan . . . . .	93
3.3	Modèle de consommation du générateur d'ondelette . . . . .	94
3.3.1	Architecture . . . . .	94
3.3.1.1	Fonctionnement général du générateur . . . . .	94
3.3.1.2	Détails sur l'architecture numérique . . . . .	95
3.3.2	Réalisation du générateur en technologie CMOS . . . . .	97
3.3.3	Réduction de la consommation grâce au Clock Gating . . . . .	98
3.4	Optimisation de la consommation . . . . .	100
3.5	Comparaison de la consommation avec l'état de l'art . . . . .	101
3.6	Au-delà de la conversion analogique-paramètre . . . . .	105
3.7	Conclusion . . . . .	107
	<b>Conclusions et Perspectives</b>	<b>109</b>
	<b>A Code MATLAB pour le SFS</b>	<b>117</b>
	<b>B SFS avec limite d'extracteur</b>	<b>119</b>
	<b>Bibliographie</b>	<b>123</b>



# Table des figures

1.1	Schéma d'un réseau de capteurs corporels . . . . .	6
1.2	Approche classique de capteur sans-fil . . . . .	7
1.3	Pont de Wheatstone en pont complet . . . . .	8
1.4	Exemple d'amplificateur d'instrumentation . . . . .	8
1.5	Principe d'échantillonnage d'un signal . . . . .	9
1.6	Fonction de transfert d'un CAN 3 bits et erreur de quantification . . . . .	10
1.7	Schéma de fonctionnement d'un CAN Flash 2 bits . . . . .	12
1.8	Schéma de fonctionnement d'un CAN Pipeline . . . . .	13
1.9	Schéma de fonctionnement d'un CAN SAR . . . . .	14
1.10	Chronogramme d'une conversion sur 4 bits par une architecture SAR . . . . .	15
1.11	Schéma de fonctionnement d'un CAN $\Sigma\Delta$ 1 bit . . . . .	16
1.12	Effet du sur-échantillonnage sur le niveau du bruit de quantification . . . . .	17
1.13	Effet du Noise Shaping sur le spectre du signal . . . . .	17
1.14	Énergie de conversion (Puissance/Fréquence de Nyquist) en fonction de l'ENOB. Les données sont issues de l'étude de B. Murmann [23]. . . . .	18
1.15	Exemples de topologie de réseau . . . . .	19
1.16	Exemple de signal parcimonieux en fréquence . . . . .	22
1.17	Signal compressible . . . . .	23
1.18	Capteur sans-fil utilisant le CS . . . . .	24
1.19	Architecture d'un échantillonnage non uniforme . . . . .	25
1.20	Architecture du Démodulateur aléatoire . . . . .	26
1.21	Architecture du RMPI . . . . .	27
1.22	Architecture du Convertisseur large bande modulé . . . . .	28
1.23	Schéma de l'architecture d'un échantillonnage non uniforme en ondelettes . . . . .	29
1.24	Architecture A2I implémentée dans [4] . . . . .	35
1.25	Comparaison de la consommation d'énergie entre un capteur utilisant une ap- proche classique et un capteur utilisant le CS . . . . .	35
1.26	Schéma de fonctionnement de la classification supervisée . . . . .	37
1.27	Matrice de confusion binaire . . . . .	38

---

1.28	Exemple d'arbre de décision permettant de savoir s'il est possible de jouer au tennis en fonction des conditions météorologiques . . . . .	39
1.29	Schéma d'un neurone artificiel . . . . .	40
1.30	Réseau de neurones multicouches . . . . .	41
1.31	Principe de fonctionnement d'un classifieur K plus proches voisins . . . . .	44
1.32	Principe de fonctionnement d'un SVM . . . . .	45
2.1	Exemples de convertisseurs A2F . . . . .	50
2.2	Architecture de convertisseur A2F proposée dans [94] . . . . .	51
2.3	Extracteur de paramètres proposé . . . . .	52
2.4	Schéma du cœur humain. Source : [96] . . . . .	53
2.5	Exemple de période de signal ECG . . . . .	54
2.6	Boîte de Heisenberg d'une ondelette . . . . .	56
2.7	Transformée en ondelettes continue . . . . .	57
2.8	Exemple d'ondelettes utilisées . . . . .	59
2.9	Ondelette de Daubechies db4 . . . . .	59
2.10	Exemples d'ondelettes de Haar constituant le dictionnaire utilisé, où chaque ligne représente une ondelette . . . . .	60
2.11	Exemples d'ondelettes de Gabor constituant le dictionnaire utilisé, où chaque ligne représente une ondelette . . . . .	61
2.12	Exemple de croisement entre deux individus codés sur 10 bits avec respectivement 1 point (5) et 2 points (3 et 8) de croisement . . . . .	67
2.13	Précision de classification des méthodes par filtrage en utilisant les ondelettes de Haar . . . . .	69
2.14	Influence de l'algorithme SFS sur les performances de classification en utilisant les ondelettes de Haar . . . . .	70
2.15	Comparaison des performances obtenues avec les ondelettes de Gabor, de Haar et de Daubechies . . . . .	71
2.16	Effet de la désynchronisation du complexe QRS sur les performances de classification . . . . .	72
2.17	Précision de classification lorsque les données d'entraînement sont désynchronisées . . . . .	73
2.18	Comparaison des ondelettes sélectionnées AVEC et SANS variation de la position du complexe QRS . . . . .	74
2.19	Schéma du circuit analogique de détection de complexe QRS . . . . .	75
2.20	Définition de l'intervalle R-R . . . . .	75
2.21	Schéma d'un réseau de neurones récurrents . . . . .	76
2.22	Performances obtenues avec la recherche séquentielle pour différents niveaux de quantification . . . . .	78
2.23	Précision de classification obtenue lors de la reconnaissance du type d'anomalie sur des données synchronisées et non quantifiées . . . . .	79

3.1	Exemples de a) deux ondelettes superposées qui ne peuvent pas être acquises par le même extracteur et b) deux ondelettes non superposées qui peuvent être acquises par le même extracteur . . . . .	84
3.2	Performances obtenues avec la recherche séquentielle optimisée pour 2, 3 et 5 extracteurs . . . . .	85
3.3	Schéma d'un extracteur de paramètre . . . . .	86
3.4	Schéma du frontal d'acquisition proposé dans [80] . . . . .	87
3.5	Schéma et symbole d'une porte de transmission CMOS . . . . .	88
3.6	Schéma du mixeur analogique . . . . .	89
3.7	Réponse en fréquence d'un intégrateur . . . . .	89
3.8	Filtres passe-bas du premier ordre . . . . .	90
3.9	Amplificateur MOS en configuration de paire différentielle . . . . .	90
3.10	Miroir de courant . . . . .	90
3.11	Intégrateur de type $G_m$ -C proposée dans [116] . . . . .	91
3.12	Architecture complète de la cellule $G_m$ -C proposée dans [116] . . . . .	92
3.13	Résultats présentés dans [116] montrant l'évolution de la fréquence de coupure du filtre en fonction des tensions $V_{gc}$ (gauche) et $V_{t2}$ (droite) . . . . .	92
3.14	Schéma matériel d'un extracteur de paramètre . . . . .	93
3.15	Schéma simplifié du fonctionnement du générateur d'ondelettes . . . . .	94
3.16	Exemple de chronogramme obtenu avec le générateur d'ondelettes . . . . .	95
3.17	Schéma d'un diviseur d'horloge . . . . .	96
3.18	Schéma d'un compteur 4 bits . . . . .	96
3.19	Schéma de placement routage d'un générateur d'ondelettes dans la technologie XFAB 0.18 $\mu\text{m}$ . . . . .	97
3.20	Clock Gating basé sur une porte $ET$ . . . . .	98
3.21	Clock gating basé sur un verrou . . . . .	98
3.22	Comparaison de l'évolution de la précision de classification et du coût d'extraction des paramètres lorsque le coût d'extraction est pris en compte lors de la sélection . . . . .	100
3.23	Comparaison de l'énergie requise pour l'acquisition de 10 secondes de signal avec 3 types de capteurs . . . . .	103
3.24	Comparaison de l'énergie requise pour l'acquisition de 10 secondes de signal entre les capteurs de l'état de l'art et les solutions à base de convertisseur A2F . . . . .	105
3.25	Convertisseur A2F avec réseau de neurones analogique . . . . .	106
4.1	Exemple de signal EEG avec l'annotation correspondante . . . . .	112
4.2	Répartition des différentes électrodes au niveau du cuir chevelu. Source : [131] . . . . .	113
4.3	Performances de classification pour la détection d'épisodes épileptiques dans des signaux EEG avec des ondelettes de Haar . . . . .	113
4.4	Sensibilité et spécificité obtenues en utilisant des NN et avec une sélection par FS et SFS . . . . .	114





# Liste des tableaux

1.1	Exemples de capteurs [16] . . . . .	6
1.2	Caractéristiques de CAN Flash . . . . .	12
1.3	Caractéristiques de CAN Pipeline . . . . .	14
1.4	Caractéristiques de CAN SAR . . . . .	15
1.5	Caractéristiques de CAN $\Sigma\Delta$ . . . . .	17
1.6	Comparaison des différentes architectures de CAN . . . . .	18
1.7	Caractéristiques de différents systèmes de transmissions sans-fil . . . . .	20
1.8	Tableau montrant la complexité et le nombre de mesures minimal pour différents algorithmes de reconstruction [3, 73] . . . . .	33
1.9	Avantages et inconvénients des différentes architectures de convertisseur A2I . . . . .	34
1.10	Avantages et inconvénients des différents modèles de classification . . . . .	47
2.1	Caractéristiques classiques du complexe PQRST pour un rythme cardiaque de 60 battements par minutes [97] . . . . .	54
2.2	Exemples de scores de Fisher pour deux paramètres extraits avec des ondelettes de Haar . . . . .	63
2.3	Avantages et inconvénients des différentes méthodes de sélection de paramètres . . . . .	68
2.4	Caractéristiques de différents circuits pour la détection de complexes QRS. . . . .	74
2.5	Comparaison de l'erreur de prédiction des intervalles R-R pour les trois approches testées . . . . .	76
3.1	Consommation des différents éléments du circuit [80] . . . . .	87
3.2	Exemples de filtres passe-bas à très faible fréquence de coupure . . . . .	91
3.3	Consommation du circuit front-end analogique . . . . .	93
3.4	Correspondance entre les signaux numériques générés et les valeurs théoriques des ondelettes . . . . .	95
3.5	Comparaison des résultats de synthèse, de surface et de consommation <i>AVEC</i> et <i>SANS</i> Clock Gating . . . . .	99
3.6	Tableau regroupant les données considérées pour l'estimation de la consommation des différents types de capteurs . . . . .	102



# Liste des acronymes

<b>A2F</b>	Analog-to-Feature
<b>A2I</b>	Analog-to-Information
<b>AOP</b>	Amplificateur Opérationnel
<b>BASN</b>	Body Area Sensor Network
<b>BLE</b>	Bluetooth Low Energy
<b>BP</b>	Basis Pursuit
<b>BPDN</b>	Basis Pursuit De-Noising
<b>CAN</b>	Convertisseur Analogique/Numérique
<b>CMOS</b>	Complementary Metal Oxyde Semiconductor
<b>CMTNN</b>	Complementary Neural Network
<b>CNA</b>	Convertisseur Numérique/Analogique
<b>CoSaMP</b>	Compressive Sampling Matching Pursuit
<b>CS</b>	Échantillonnage compressif
<b>DFT</b>	Discrete Fourier Transform
<b>DWT</b>	Discrete Wavelet Transform
<b>ECG</b>	Électrocardiogramme
<b>EEG</b>	Électroencéphalogramme
<b>EMG</b>	Électromyogramme
<b>ENOB</b>	Effective Number Of Bits
<b>FN</b>	Faux Négatif
<b>FNN</b>	Falsity Neural Network

<b>FoM</b>	Figure of Merite
<b>FP</b>	Faux Positif
<b>FPGA</b>	Field Programmable Gate Arrays
<b>FS</b>	Fisher Score
<b>HBC</b>	Human Body Communications
<b>IG</b>	Information Gain
<b>IoT</b>	Internet of Things
<b>LNA</b>	Low Noise Amplifier
<b>LPF</b>	Low Pass Filter
<b>LSB</b>	Least Significant Bit
<b>ML</b>	Machine Learning
<b>MP</b>	Matching Pursuit
<b>MSB</b>	Most Significant Bit
<b>NAR</b>	Nonlinear AutoRegressive network
<b>NN</b>	Neural network
<b>NUS</b>	Non-Uniform Sampling
<b>NUWS</b>	Non-Uniform Wavelet Sampling
<b>OMP</b>	Orthogonal Matching Pursuit
<b>OSR</b>	OverSampling Ratio
<b>PGA</b>	Programmable Gain Amplifier
<b>PSD</b>	Power Spectral Density
<b>RD</b>	Random Demodulator
<b>RIP</b>	Restricted Isometry Property
<b>RMPI</b>	Random Modulator Pre-Integrator
<b>RNN</b>	Recurrent Neural Network
<b>ROMP</b>	Regularized Orthogonal Matching Pursuit
<b>S&amp;H</b>	Sample and Hold (échantillonneur bloqueur)

<b>SAR</b>	Successive Approximation Register
<b>SFS</b>	Sequential Forward Search
<b>SMOTE</b>	Synthetic Minority Oversampling Technique
<b>SNDR</b>	Signal to Noise and Distortion Ratio
<b>SNR</b>	Signal to Noise Ratio
<b>SVM</b>	Support Vector Machine
<b>TNN</b>	Truth Neural Network
<b>TRMC</b>	Taux de Réjection de Mode Commun
<b>UWB</b>	Ultra Wideband
<b>VN</b>	Vrai Négatif
<b>VP</b>	Vrai Positif
<b>WBAN</b>	Wireless Body Area Network
<b>WPAN</b>	Wireless Personal Area Network
<b>WSN</b>	Wireless sensor networks
<b>ZR</b>	Zero Rule classifier



# Publications et communications

## Revues

[MicroJ21] (Soumis) **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.

Power-aware feature selection for optimized Analog-to-Feature converter

In *Microelectronics Journal*

## Conférences internationales

[AVIC21] **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.

Analog-to-feature converter optimization through power-aware feature selection.

In *International Conference on Analog VLSI Circuits*, Octobre 2021, Bordeaux, France.

[NEWCAS20] **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.

Feature selection algorithms for flexible analog-to-feature converter.

In *18th IEEE International New Circuits and Systems Conference (NEWCAS)*, Juin 2020, Montréal, Canada.

## Conférences nationales

[IEEE-CAS19] **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.

Feature Selection Algorithms for the Design of a Flexible Analog-To-Feature Converter.

In *Technical workshop, sponsored by IEEE-CAS and GDR SOC2, on Integrated machine-learning hardware for near-sensor computing applications*, IEMN - Villeneuve d'Ascq, France, Novembre 2019.

[JDSE19] **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.

Design of a flexible analog-to-feature converter for smart acquisition of biological signals.

In *Junior Conference on Data Science and Engineering*, Septembre 2019, Paris-Saclay, France.

[GDR-SoC2-19] **Antoine BACK**, Paul CHOLLET, Olivier FERCOQ et Patricia DESGREYS.  
Conception d'un convertisseur analogique-paramètres pour l'acquisition intelligente de signaux biologiques.

In *Colloque du GDR SOC2* Juin 2019, Montpellier, France.



# Introduction

Dans le contexte des capteurs sans fil et plus généralement, dans le domaine de l'internet des objets (IoT), la consommation énergétique des systèmes de communication sans fil, destinés à la transmission des données du capteur vers l'agrégateur, représente une part significative de la consommation énergétique totale des capteurs. Cela peut en effet représenter jusqu'à 96% du budget énergétique total comme montré dans [1]. C'est donc le principal levier d'économie d'énergie pour augmenter leur autonomie. Une première solution consiste à améliorer l'efficacité énergétique des systèmes de transmission. Une seconde solution consiste à proposer de nouvelles manières de transmettre les données de façon intelligente comme l'échantillonnage compressif (CS) [2]. Le CS est une technique bien connue qui permet de réduire le nombre d'échantillons acquis, par rapport aux approches classiques fonctionnant au taux de Nyquist, et donc de réduire la quantité de données à transmettre. Le but du CS est de faire l'acquisition de données compressées et ensuite de reconstruire, à partir de ces données, le signal d'origine grâce à un algorithme de reconstruction. Le problème est que ces algorithmes sont complexes [3] et que le taux de compression est limité [4].

Des travaux antérieurs montrent le gain d'énergie obtenu grâce à la réduction de la quantité de données via l'extraction de paramètres dans le domaine analogique [5]. Les convertisseurs analogique-paramètre (Analog-to-Feature, A2F) sont des systèmes d'acquisition pensée pour l'IoT, dans le but d'augmenter l'autonomie de ces objets. Comme pour le CS, le but des convertisseurs A2F est de diminuer le nombre d'échantillons acquis en extrayant seulement les informations utiles du signal. Mais contrairement au CS, le but ici n'est pas de reconstruire le signal, mais d'utiliser les informations extraites comme données d'entrée d'algorithmes de Machine Learning (ML), par exemple, pour réaliser de la détection d'activité vocale ou bien pour détecter des anomalies dans un électrocardiogramme (ECG), un électroencéphalogramme (EEG) ou un électromyogramme (EMG). Les solutions actuelles de convertisseurs A2F sont conçues pour des applications spécifiques telles que la détection d'activité vocale, comme dans [6, 7]. L'objectif principal lors de la conception d'un convertisseur analogique-paramètre est donc de définir, pour une application donnée, ce qu'est une information utile et comment l'extraire. Dans [6, 7] l'extraction de paramètres consiste à extraire le contenu énergétique de plusieurs bandes de fréquences dans le domaine des fréquences audibles.

Les travaux menés dans cette thèse font suite aux travaux menés dans [8] : l'objectif ici est donc de proposer une architecture et une méthode afin de concevoir un convertisseur A2F générique, utilisable pour différents types de signaux basse fréquence (ECG, EEG, voix, ...). Pour cela, il est possible d'utiliser l'échantillonnage non-uniforme en ondelettes (NUWS), qui est une méthode issue du CS, pour réaliser l'extraction de paramètres. Le NUWS offre de nombreux degrés de liberté, ce qui permet d'en faire une méthode d'extraction générique, mais redondante. Des algorithmes de sélection de paramètres comme le score de Fisher (FS), le gain d'information (IG) ou la recherche séquentielle (SFS), permettent donc de déterminer, pour une application donnée, quelles doivent être les informations à extraire en vue de la classification. Lorsque les caractéristiques des ondelettes à utiliser, pour obtenir des informations pertinentes sur le signal à traiter, sont déterminées, une architecture matérielle pourra être proposée et ainsi obtenir un modèle de consommation énergétique qui permettra d'évaluer le gain de consommation par rapport à une approche classique de capteur sans fil. L'objectif de cette thèse est donc de proposer un modèle de consommation énergétique d'un convertisseur analogique-paramètre pour l'acquisition de signaux de faible fréquence et ainsi de montrer l'intérêt de cette technologie dans le domaine des capteurs connectés.

Tout d'abord, le Chapitre 1 présente l'état de l'art des capteurs sans-fil. Comme dans [8], ce chapitre permet de faire un tour d'horizon des outils et techniques utilisés dans le domaine des capteurs sans fils. Différentes techniques d'échantillonnage seront abordées, la méthode classique au taux de Nyquist et l'échantillonnage compressif. Nous verrons également les outils utilisés pour mettre en place ces deux techniques. Différents protocoles de communication et algorithmes de classification/régression seront également présentés. Ce Chapitre permet d'abord d'identifier les leviers d'action afin de réduire la consommation des capteurs sans-fil, cela permet notamment de mettre en lumière et de quantifier la différence de consommation entre les processus d'acquisition et de transmission. Il permet également d'identifier les verrous proposés par la technique du CS qui nécessite des algorithmes de reconstruction complexes afin d'obtenir une reconstruction fidèle du signal original. Pour finir, ce Chapitre permet de présenter différents modèles de classification dont l'utilisation est centrale dans le concept de convertisseur A2F. En effet, l'objectif du CS étant d'acquérir la même quantité d'information avec moins d'échantillons qu'avec une technique classique d'échantillonnage, le taux de compression obtenus est limité. L'utilisation d'algorithmes d'apprentissage automatique permet de réduire cette contrainte en ne faisant l'acquisition que de l'information nécessaire à la réalisation de la tâche souhaitée.

Le premier objectif du Chapitre 2 est tout d'abord de présenter le concept de convertisseur A2F, dont l'objectif est d'extraire des informations du signal de manière analogique pour ensuite les utiliser comme données d'entrée d'algorithmes de classification, de présenter également le principe d'échantillonnage non-uniforme en ondelettes et finalement, de présenter différentes méthodes de sélections de paramètres, dont l'utilisation est nécessaire afin de réduire au maximum la quantité de données. Ces méthodes sont alors testées dans le cadre d'une

détection d'anomalie dans des signaux ECG. Par rapport à l'étude menée dans [8], cette étude explore de manière plus approfondie le processus de sélection de paramètres dans le cadre d'une détection d'anomalie dans des signaux ECG. De nouvelles pistes sont étudiées, comme l'utilisation des machines à supports de vecteurs (SVM) pour la classification, ou encore l'utilisation du gain d'information comme méthode de sélection. Les algorithmes génétiques sont également considérés comme méthode de sélection, mais ont été jugés moins appropriés que la recherche séquentielle pour minimiser le nombre de paramètres sélectionnés. L'utilisation généralisée de cette dernière, combinée au gain d'information utilisé comme méthode de pré-sélection, a montré de meilleurs résultats, lorsque l'on considère la précision et le nombre de paramètres, par rapport aux résultats obtenus dans [8]. Une étude est également menée sur l'effet sur les performances de classification, d'une mauvaise synchronisation des signaux ECG, et plus particulièrement de l'onde R, par rapport à la fenêtre d'analyse. Celle-ci est menée en utilisant les résultats de sélections de la combinaison IG et SFS, plutôt qu'avec le seul score de Fisher, comme c'est le cas dans [8]. De plus, celle-ci est réalisée pour des écarts de synchronisation beaucoup plus grands et bien plus réalistes. Différents modèles de prédiction de l'intervalle R-R sont ainsi proposés, comme les réseaux de neurones récurrents, afin d'estimer l'erreur obtenue pour cette tâche. Les résultats obtenus grâce à ces modèles montrent que les valeurs choisies dans l'étude de [8] n'étaient pas réalistes. Une nouvelle évaluation du niveau de quantification nécessaire des paramètres est également réalisée, la sélection des paramètres, basée sur le SFS, est réalisée sur les données quantifiées afin de prendre en compte l'éventuelle perte d'information, engendrée par le processus de quantification, ce qui n'était pas le cas dans [8] puisque les paramètres étaient sélectionnés en amont sans tenir compte de la quantification. Une dernière étude est réalisée afin d'exploiter la reconfigurabilité du convertisseur A2F proposé : celle-ci a pour but de sélectionner de nouveaux paramètres permettant de distinguer deux catégories d'anomalies en se limitant aux seules données anormales de la base.

Le Chapitre 3 se focalise sur l'estimation et l'optimisation de la consommation d'un capteur sans fils utilisant un convertisseur A2F basé sur le NUWS. Une première méthode pour optimiser la consommation est proposée dans [8]. L'objectif de cette méthode consiste à sélectionner des paramètres de façon à ce qu'ils puissent être extraits avec un unique extracteur, or on constate que l'utilisation d'un unique extracteur limite la précision. Une généralisation de cette méthode est réalisée grâce à l'adaptation de l'algorithme SFS auquel un nombre maximal d'extracteur est fourni. Deux nouveaux processus de sélections sont ainsi réalisés avec un nombre maximal d'extracteur fixé à trois et cinq, permettant ainsi de constater que trois extracteurs sont suffisants pour obtenir une précision équivalente à ce qui a pu être obtenu dans le Chapitre 2. Une seconde approche est proposée afin d'optimiser la consommation du convertisseur A2F. Jusqu'à présent, la sélection par SFS ne s'effectuait qu'en fonction de la précision de classification. Cette nouvelle approche consiste donc à sélectionner les paramètres en fonction d'un nouveau critère, tenant compte de la précision de classification et l'énergie nécessaire pour extraire le paramètre. Un critère est donc proposé afin de maximiser la précision tout en

minimisant la consommation. La mise en place de cette approche nécessite la réalisation d'un modèle de consommation de convertisseur A2F afin d'estimer le coût d'extraction de chaque ondelette. Ce modèle est réalisé à partir de données de consommation d'éléments provenant de l'état de l'art et de la réalisation d'un circuit numérique pour la génération d'ondelettes. Des travaux ont également été menés sur la réalisation de ce modèle de consommation, au niveau transistor, pour la globalité de l'architecture, mais faute de temps et d'expérience, ces derniers n'ont pas abouti. Ce qui nous a poussé à utiliser des données de l'état de l'art pour modéliser la partie analogique du circuit. La consommation du circuit numérique pour la génération des ondelettes est estimée avec les outils CADENCE® pour la technologie XFAB CMOS 0.18  $\mu\text{m}$ . Le coût énergétique de chaque ondelette peut ainsi être mesuré, permettant une sélection favorisant les paramètres nécessitant peu d'énergie. Ces travaux ont permis d'identifier de nouveaux paramètres permettant l'obtention d'une précision similaire à ce qui avait pu être obtenu dans les précédentes étapes, soit environ 98% de précision, tout en divisant par trois la quantité d'énergie nécessaire pour l'extraction des paramètres. Pour finir, une comparaison entre les trois types de capteurs (Nyquist, A2I et A2F) est réalisée pour montrer l'intérêt des convertisseurs A2F dans le domaine des capteurs sans fil. Pour le traitement et la transmission de dix secondes de signal ECG, l'utilisation d'un convertisseur A2F permettrait de diviser la quantité d'énergie nécessaire par 15.3, par rapport à l'approche classique et par 3.7 par rapport au capteur utilisant un convertisseur A2I. Dans le cas où le capteur transmet les paramètres et non le résultat de classification.

Nous terminerons par une conclusion et des perspectives visant à accroître davantage le gain de consommation du convertisseur A2F. Pour résumer, les contributions apportées dans cette thèse sont :

- un affinage de la sélection des paramètres permettant d'obtenir de meilleures performances avec moins de paramètres ;
- une évaluation plus réaliste des effets d'une mauvaise synchronisation des signaux ECG sur la précision de classification ;
- un modèle de prédiction de l'intervalle R-R permettant la synchronisation ;
- l'étude de la généricité du système avec l'étude d'une autre application pour la catégorisation d'erreurs ;
- la réalisation d'un modèle de consommation basé sur des données de l'état de l'art et de la réalisation d'un circuit numérique personnalisé ;
- la proposition d'une méthode de sélection permettant l'optimisation de la consommation d'un convertisseur A2F basé sur le NUWS.

# Chapitre 1

## Échantillonnage et traitement de l'information

### 1.1 Introduction sur les capteurs sans-fil

L'intérêt pour les réseaux de capteurs sans fil (Wireless Sensors Network, WSN) est de plus en plus important. Ils sont devenus un des domaines les plus étudiés au cours des dernières années. Un WSN peut être défini comme un groupe de petits appareils, appelés capteurs, qui travaillent en coopération pour recueillir et communiquer, via des outils de communication sans-fil, des informations recueillies sur leur environnement. Les données collectées sont envoyées vers un agrégateur qui traite les données localement ou, est connecté à d'autres réseaux, par exemple Internet.

Avec les progrès technologiques, les capteurs deviennent de plus en plus intelligents, petits et moins chers, ce qui participe à leur démocratisation. Des milliards de capteurs sans fil sont ainsi utilisés dans de nombreuses applications telles que l'industrie, l'agriculture [9, 10], l'urbanisme [11, 12], l'environnement [13, 14, 15], la santé et la sécurité. En cas de catastrophe naturelle, les capteurs permettraient de les anticiper. Dans le domaine de la santé, ils peuvent aider à surveiller la santé d'un patient.

Ces différentes applications leur imposent plusieurs contraintes de conception : la taille du capteur ou du réseau, la topologie du réseau, la portée de la communication, la consommation, la capacité de traitement ou le stockage. Les réseaux de capteurs corporels (Body Area Sensors Network, BASN) constituent un bon exemple d'application ayant une forte contrainte sur la consommation des capteurs. La Figure 1.1 illustre un exemple de réseaux de capteurs corporels. Les capteurs permettant de mesurer l'activité cardiaque (ECG), musculaire (EMG) ou cérébrale (EEG), où les accéléromètres et les gyroscopes sont des exemples de capteurs pouvant être utilisés dans ce type de réseau. Le Tableau 1.1 présente quelques exemples de capteurs pour les BASN [16], ainsi que les signaux mesurés et leur fréquence d'acquisition, qui dépend des caractéristiques de chacun des signaux.

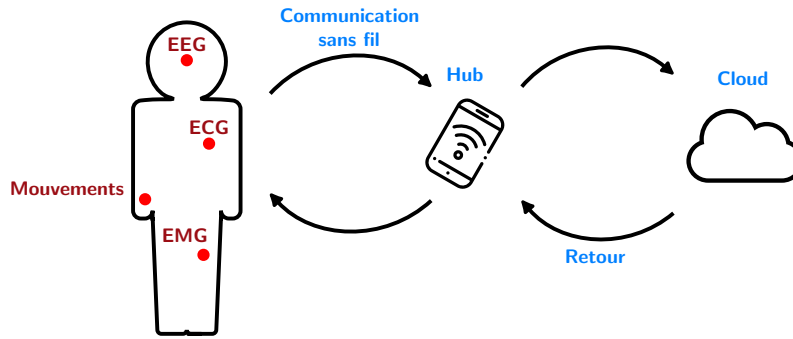


FIGURE 1.1 – Schéma d'un réseau de capteurs corporels

Le rôle des capteurs est donc de mesurer des grandeurs physiques pour obtenir une représentation de leur évolution temporelle. Pour détecter des anomalies, ou plus généralement des zones intérêt, dans le signal, un traitement est réalisé sur celui-ci pour extraire des informations caractéristiques de ces zones, également appelées "paramètres", qui peuvent ensuite être utilisées par des algorithmes de classification ou de régression. Dans une approche classique, ces deux étapes sont réalisées au niveau de l'agrégateur après réception des échantillons provenant des capteurs, comme illustré à la Figure 1.2a. Une fréquence d'échantillonnage élevée impose alors une utilisation importante du système de transmission. Ce dernier peut alors engendrer une part importante de la consommation d'énergie globale du capteur : dans [1], la transmission des échantillons représente 96% de la consommation globale.

Pour réduire la consommation due à la transmission, il est possible de réaliser l'extraction des paramètres directement au niveau du capteur, comme illustré à la Figure 1.2b, ou même, de réaliser l'étape de classification, ce qui permet de diminuer la quantité de données et ainsi d'utiliser le système de transmission de façon plus parcimonieuse. Dans [1], l'extraction des paramètres directement au niveau du capteur permet de diminuer la consommation liée au transmetteur de l'ordre de 97%, au prix de l'ajout d'un système d'extraction.

TABLE 1.1 – Exemples de capteurs [16]

Capteur	Grandeur mesurée	Fréquence d'échantillonnage
ECG	Activité électrique du cœur	100-1000 Hz
EEG	Activité électrique du cerveau	100-1000 Hz
EMG	Activité électrique des muscles	100-1000 Hz
Accéléromètre	Accélération sur 3 axes	10-100 Hz
Oxymètre	Taux d'oxygénation du sang	0,1-10 Hz
Capteur de pression artérielle	Pression artérielle systémique (tension artérielle)	0,1-10 Hz
Microphone	Activité sonore	10-100 kHz

## 1.2 Approche classique des capteurs sans fil

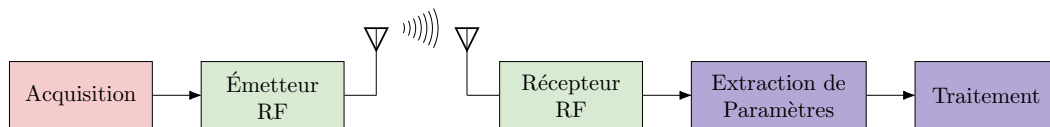
La méthode classique d'acquisition des signaux est celle basée sur le théorème de Shannon-Nyquist qui consiste en la mesure de la valeur d'un signal à intervalles de temps réguliers, afin d'en faire une représentation temporelle. Nous verrons ici les différents outils et techniques utilisés, pour les capteurs sans fils, afin d'acquérir, traiter et transmettre les données.

### 1.2.1 Capteur et conditionnement

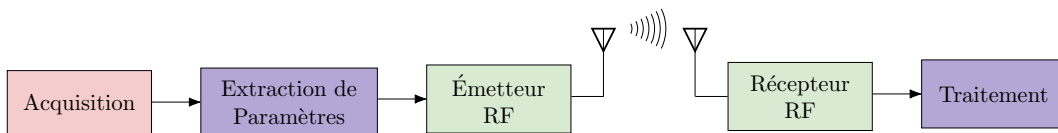
En règle générale, lorsque l'on parle de capteurs sans-fil, on parle de l'ensemble constitué d'un capteur, d'un système de conditionnement, comme un amplificateur ou un filtre, d'un convertisseur analogique-numérique (CAN), d'un calculateur numérique, comme un microcontrôleur, et d'un système de transmission sans-fil. Un capteur est un dispositif qui permet de transformer une grandeur physique, appelée le mesurande, comme la température, la pression ou une position, en un signal, le plus souvent électrique. On peut classer les capteurs en plusieurs catégories selon :

- Le mesurande (capteur de vitesse, de déformation, de flux lumineux, ...);
- le type de signal fourni en sortie, qui peut être analogique ou numérique;
- le principe physique permettant de traduire le mesurande (Effet photoélectrique, piézo-électrique, Hall, ...);
- leur principe de fonctionnement (Actif ou Passif).

Les capteurs actifs transforment directement le mesurande en une grandeur électrique alors que les capteurs passifs nécessitent une source d'énergie externe, car le signal délivré est fonction de la variation d'impédance du capteur. Cette variation d'impédance est dû à l'effet du mesurande sur la géométrie ou les propriétés électriques du matériau composant le



(a) Capteur transmetteur



(b) Capteur avec extraction de paramètres

FIGURE 1.2 – Approche classique de capteur sans-fil

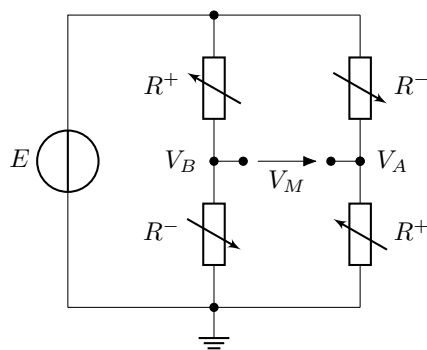


FIGURE 1.3 – Pont de Wheatstone en pont complet

capteur. Les principales caractéristiques d'un capteur sont : le mesurande, l'étendue de mesure, la sensibilité, la résolution, la précision, la linéarité, la plage de température de fonctionnement, l'hystérésis.

Dans le cas de capteur avec une sortie analogique, le signal issu directement du capteur est souvent d'une amplitude trop faible, trop bruité, et peut également contenir des composantes parasites et indésirables, ce qui rend le signal incompatible avec les caractéristiques du CAN. Par exemple, dans le cas d'un électrocardiogramme, les électrodes délivrent un signal de l'ordre du millivolt auquel s'ajoutent différents signaux parasites provenant des autres muscles, de la respiration du patient ou d'un mouvement des électrodes. On utilise alors un système de conditionnement pour améliorer la qualité du signal avant la numérisation. L'étape de conditionnement permet également de compenser l'influence de certaines grandeurs, comme la température, les champs magnétiques ou l'humidité, sur la valeur de la mesure. Le montage en pont de Wheatstone, illustré à la Figure 1.3, où  $R^+$  et  $R^-$  ont des sensibilités opposées, est un exemple de circuit de conditionnement. Utilisé en pont complet, il permet de compenser l'effet de certaines grandeurs d'influence du fait de sa structure différentielle, car les capteurs seraient affectés de manière égale.

Les signaux issus de capteurs sont généralement de faible amplitude. Il est donc nécessaire de les amplifier pour obtenir la meilleure précision possible. Le composant de base pour cette

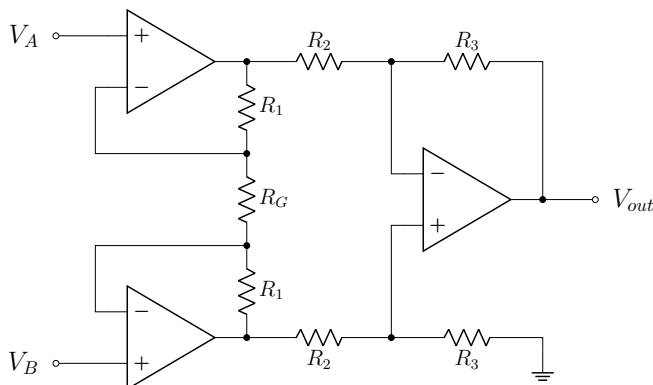


FIGURE 1.4 – Exemple d'amplificateur d'instrumentation



étape est l'amplificateur opérationnel (AOP) utilisé en mode différentiel. Les amplificateurs d'instrumentation sont des amplificateurs différentiels composés de plusieurs AOP, comme illustré à la Figure 1.4, qui permettent d'obtenir un meilleur taux de réjection de mode commun (TRMC) mais consomment plus que les montages simples à un seul AOP.

L'amplificateur permet ainsi d'obtenir un signal de plus grande amplitude et donc d'occuper au maximum la pleine échelle du CAN pour obtenir la meilleure précision lors de la conversion. Après cette étape d'amplification, un filtre peut également être utilisé afin de supprimer le bruit parasite du signal et ne conserver que les bandes de fréquence d'intérêt.

### 1.2.2 Échantillonnage de Nyquist

Les approches classiques de capteurs sans fils acquièrent les signaux en se basant sur le théorème de Shannon-Nyquist [17]. Cette phase d'acquisition consiste à représenter un signal en transformant à intervalle régulier une grandeur analogique en données numériques : pour cela on peut utiliser un convertisseur analogique numérique (CAN). Les données numérisées sont ensuite transmises à l'agrégateur où elles peuvent être traitées. La Figure 1.2 représente les différentes étapes de ce type d'approche.

Un CAN est un circuit permettant de traduire un signal analogique, c'est-à-dire un signal continu en temps et en valeurs, en une série d'informations binaire qui, par opposition, est un signal discret en temps et en valeurs. Le signal est donc, dans un premier temps, discrétisé en temps (Figure 1.5), cela veut dire que la valeur du signal est mesurée à intervalle de temps régulier. Cet intervalle est appelé la période d'échantillonnage  $T_s$ .

$$x_e[k.T_s] = x(t) \cdot \sum_{n=-\infty}^{n=\infty} \delta(t - n.T_s) \quad (1.1)$$

où  $x_e[k.T_s]$  est le  $k^{\text{ième}}$  échantillon du signal échantillonné,  $x(t)$  est le signal analogique et  $\delta(t)$  est la distribution de Dirac.

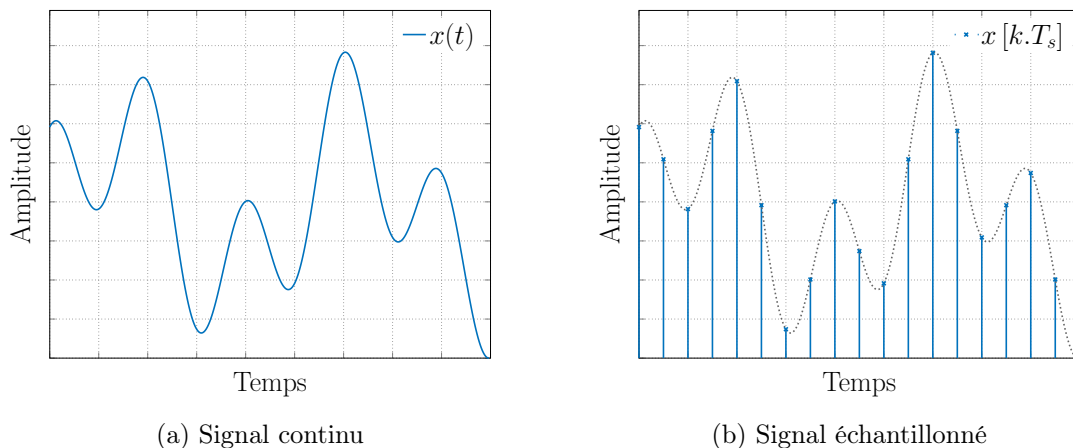


FIGURE 1.5 – Principe d'échantillonnage d'un signal

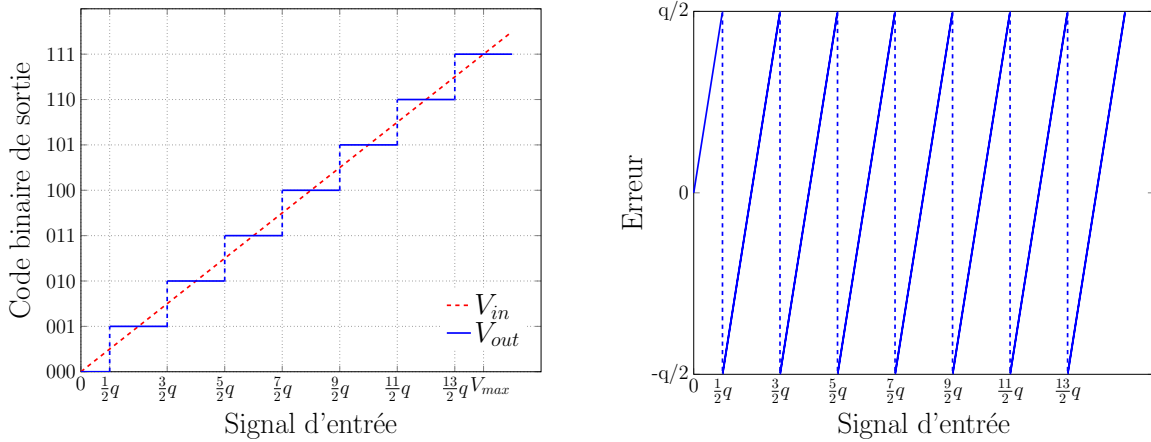


FIGURE 1.6 – Fonction de transfert d'un CAN 3 bits et erreur de quantification

Pour déterminer la fréquence d'échantillonnage  $F_s = 1/T_s$  d'un signal analogique, on applique le théorème de Shannon-Nyquist. Il permet d'obtenir une reconstruction fidèle du signal. Ce théorème dit que, pour permettre une reconstruction parfaite du signal, la fréquence d'échantillonnage doit être au moins deux fois supérieure à la fréquence maximale du signal. Ceci permet d'éviter l'apparition du phénomène de recouvrement de spectre.

$$X_e(f) = \frac{1}{T_s} X(f) * \sum_{n=-\infty}^{n=\infty} \delta(f - n.F_s) = \frac{1}{T_s} \sum_{n=-\infty}^{n=\infty} X(f - n.F_s) \quad (1.2)$$

où  $X(f)$  est la transformée de Fourier de  $x(t)$  et  $*$  est l'opérateur du produit de convolution.

L'opération de convolution après échantillonnage a pour effet de dupliquer le spectre du signal analogique centrés sur les multiples de  $F_s$ . Le recouvrement de spectre se caractérise par la superposition des différentes copies du spectre. Pour limiter ce phénomène et obtenir une reconstruction fidèle du signal, il suffit d'effectuer un filtrage passe-bas qui ne conserve que le signal contenu dans la bande  $[-B; B]$ . La fréquence  $B$ , appelée aussi bande passante du signal s'exprime comme  $B = F_s/2$ . Le spectre du signal de départ sera d'autant moins déformé que la fréquence d'échantillonnage sera grande devant la bande  $B$  du signal d'entrée.

Comme il a été dit précédemment, un CAN permet de transformer un signal continu en temps et en valeurs en un signal discret binaire. L'opération d'un CAN peut s'écrire sous la forme de l'équation suivante et illustrée à la Figure 1.6 :

$$A = b_1 \frac{V_{ref}}{2} + b_2 \frac{V_{ref}}{4} + \dots + b_n \frac{V_{ref}}{2^n} + e \quad (1.3)$$

où  $A$  est la valeur du signal analogique à numériser,  $V_{ref}$  est la tension de pleine échelle du convertisseur et qui correspond à la différence entre la tension maximale et minimale supportée par le CAN,  $e$  est l'erreur de quantification,  $n$  correspond au nombre de bit du CAN (aussi appelée résolution),  $b_1$  est le bit de poids le plus fort (MSB : Most Significant Bit) et  $b_n$  : le bit de poids le plus faible (LSB : Least Significant Bit).

On définit également le pas de quantification, que l'on nomme également "quantum" :

$$q = \frac{V_{ref}}{2^n} \quad (1.4)$$

L'approximation induite par la résolution du convertisseur lors de la quantification est appelée "erreur de quantification" et est comprise entre  $-q/2$  et  $q/2$  (voir Figure 1.6), donc plus la résolution est importante, plus on limite cette erreur.

Il existe différentes métriques pour mesurer les performances d'un CAN :

- Le rapport signal sur bruit (SNR) qui correspond au rapport de la puissance du signal d'entrée sur la puissance du bruit :

$$SNR_{dB} = 10 \log_{10} \left( \frac{P_{signal}}{P_{Bruit}} \right) \quad (1.5)$$

- Le rapport signal sur bruit et distorsion (SNDR) correspond au rapport de la puissance du signal d'entrée sur celle du bruit de quantification, du bruit aléatoire et des distorsions harmoniques :

$$SNDR_{dB} = 10 \log_{10} \left( \frac{P_{Signal}}{P_{Bruit} + P_{Distorsion}} \right) \quad (1.6)$$

- Le nombre de bits effectif (ENOB) correspond au nombre de bits nécessaire pour qu'un CAN idéal ait un SNR équivalent au SNDR du CAN caractérisé. L'ENOB est lié au SNDR par la relation :

$$ENOB = \frac{SNDR - 1.76}{6.02} \quad (1.7)$$

- Le facteur de mérite (FoM) est une métrique qui permet de combiner plusieurs mesures de performance pour obtenir une seule donnée permettant ainsi de comparer plusieurs CAN. Il existe plusieurs FoM comme le FoM de Walden, qui s'exprime en énergie par conversion ou le FoM de Schreier, qui s'exprime en dB :

$$FoM_W = \frac{Puissance}{F_s \cdot 2^{ENOB}} \quad (1.8)$$

$$FoM_S = SNDR + 10 \log_{10} \left( \frac{BW}{P} \right) \quad (1.9)$$

### 1.2.3 Les architectures des convertisseurs analogiques numériques

Un grand nombre d'architectures de CAN ont été développées pour répondre aux besoins d'une grande diversité d'applications. Le choix d'une architecture dépend alors grandement de l'application souhaitée, suivant si celle-ci nécessite de la rapidité, de la précision ou une faible consommation.

### 1.2.3.1 CAN Flash

L'architecture Flash [18, 19, 20] est la plus rapide, en effet, elle effectue la conversion des bits en parallèle et donc réalise la conversion en une période d'horloge. La Figure 1.7 présente le fonctionnement d'un CAN Flash sur 2 bits. Cette architecture est composée d'une chaîne de  $2^n$  résistances, de  $2^n - 1$  comparateurs en parallèles et d'un encodeur. Cette chaîne de résistance permet d'obtenir une série de tensions de seuil linéaire. Le signal analogique d'entrée est ensuite comparé à ces tensions de seuil et si le signal est supérieur à celles-ci alors les sorties des comparateurs passent à l'état logique "1". En se basant sur les valeurs délivrées par les comparateurs, l'encodeur génère le code binaire associé.

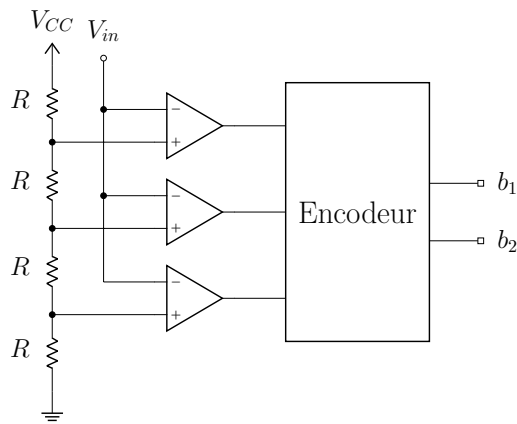


FIGURE 1.7 – Schéma de fonctionnement d'un CAN Flash 2 bits

Le principal avantage de cette architecture est sa large bande de conversion. Elle possède également une très faible latence (conversion en un seul cycle d'horloge) mais sa complexité croît de manière exponentielle avec le nombre de bits, en effet, pour  $n$  bits de résolution,  $2^n - 1$  comparateurs sont nécessaires, ce qui pose des problèmes de taille et de consommation d'énergie et limite sa résolution. Le Tableau 1.2 présente les caractéristiques de plusieurs convertisseurs de type Flash. Les exemples présentés ici sont les exemples issus de l'étude de B. Murmann [23] ayant le meilleur facteur de mérite (FoM).

TABLE 1.2 – Caractéristiques de CAN Flash

Réf.	Technologie (nm)	Surface (mm <sup>2</sup> )	ENOB (bits)	P (mW)	F <sub>s</sub> (GHz)	FoM (fJ/conv-step)
[18]	32 (SOI)	0.02	4.8	8.5	5	59.3
[19]	40	0.021	5.2	11	3	99.3
[20]	65	0.08	6.5	21	2	118.6
[21]	180	2	5	1.66e-3	4e-4	127.6
[22]	90	0.033	3.7	2.52	1.25	159.4

### 1.2.3.2 CAN pipeline

Dans les CAN de type pipeline [24, 25, 26], la conversion se fait à l'aide de plusieurs étages en cascade, ce qui permet d'obtenir une résolution plus élevée qu'avec un CAN Flash tout en nécessitant moins de comparateurs. La Figure 1.8 illustre le fonctionnement d'un CAN pipeline. Il est composé de  $N$  étages, chacun étant composé des éléments suivants :

- Un CAN flash de résolution  $M_i$  ( $i$  est l'index de l'étage)
- Un CNA de résolution  $M_i$
- Un soustracteur
- Un gain de valeur  $2^{M_i}$

Le principe de fonctionnement d'un CAN pipeline est le suivant : le signal à numériser est, dans un premier temps, discrétisé à une faible résolution par le CAN du premier étage, ce qui permet de générer les bits de poids forts de la sortie numérique. La précision du CAN des différents étages est généralement comprise entre 1 et 5 bits. Le signal numérique  $D_i$  est reconverti en un signal analogique afin de le soustraire au signal d'entrée. Le signal  $v_i$  qui en résulte correspond à l'erreur de quantification du CAN de l'étage. Cette erreur est amplifiée avec un gain de  $2^{M_i}$  et est transmise à l'étage suivant. Ce dernier réalise le même processus permettant ainsi d'augmenter la précision du convertisseur. Les sorties numériques des différents étages  $M_i$  sont ainsi recombinaées pour créer la sortie numérique globale du CAN.

La résolution globale du CAN est la somme des résolutions des étages. Ainsi pour construire un CAN 12 bits, on peut le réaliser par exemple à l'aide de 3 étages Flash de 4 bits chacun. Cette implémentation nécessitera 45 comparateurs ( $3 \times (2^4 - 1)$ ) alors que pour une architecture

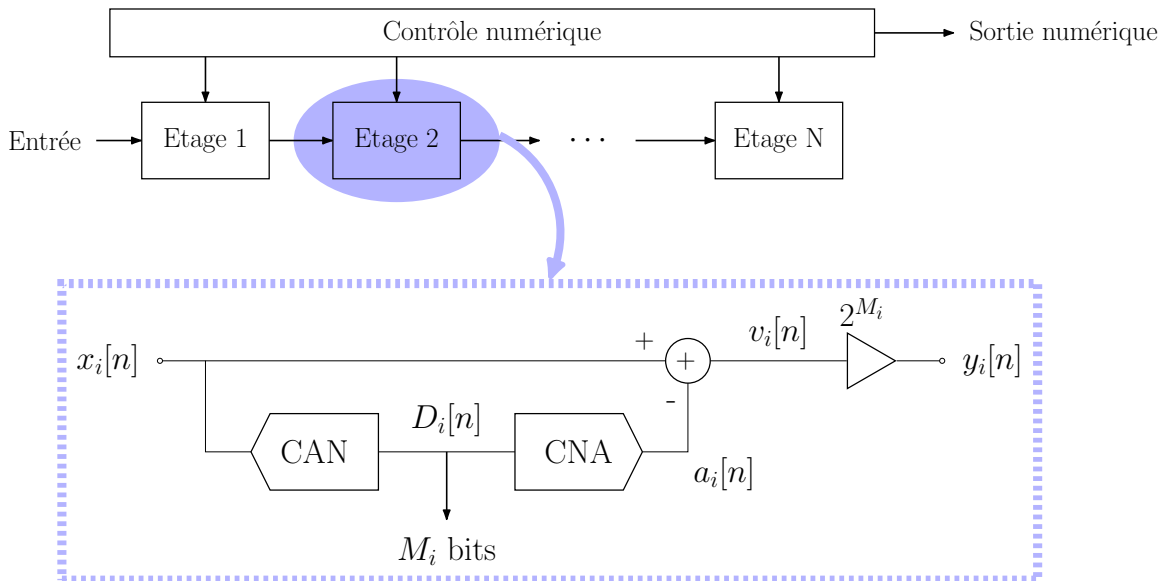


FIGURE 1.8 – Schéma de fonctionnement d'un CAN Pipeline

TABLE 1.3 – Caractéristiques de CAN Pipeline

Réf.	Technologie (nm)	Surface (mm <sup>2</sup> )	ENOB (bits)	P (mW)	F <sub>s</sub> (MHz)	FoM (fJ/conv-step)
[24]	28	0.11	8.8	2.14	410	11.4
[25]	16	0.037	9.7	6	600	12
[26]	40	0.066	9	1.7	250	13.2
[27]	28	0.015	9.1	6	500	21.7
[28]	180	1.98	12.3	2.96	20	29

Flash classique de 12 bits, il faut 4095 ( $2^{12} - 1$ ) comparateurs. L'inconvénient de l'architecture Pipeline est l'étage de soustraction/gain, qui nécessite l'utilisation d'un amplificateur opérationnel. La vitesse des CAN pipeline est alors limitée par rapport aux CAN Flash. Ils permettent donc d'acquérir des signaux à bande passante moyenne ou large (quelques dizaines à quelques centaines de MHz) avec une résolution moyenne ou haute. Les caractéristiques de plusieurs implémentations de CAN avec une architecture de type pipeline sont présentées dans le Tableau 1.3.

### 1.2.3.3 CAN à approximation successive

Les CAN à approximation successive (Successive Approximation Register, SAR) [29, 30, 31] permettent une acquisition plus lente que les convertisseurs flash, mais ont une complexité matérielle réduite ce qui permet d'augmenter la résolution. L'architecture d'un CAN SAR, présentée à la Figure 1.9, est composée d'un échantillonneur (S&H), d'un comparateur, d'un bloc logique de contrôle et d'un convertisseur numérique analogique (CNA). Les différents bits sont déterminés les uns après les autres, en commençant par le bit de plus fort poids. Contrairement aux CAN Flash qui ne nécessitent qu'un seul cycle d'horloge pour réaliser la conversion, les CAN SAR nécessitent  $n$  périodes d'horloge pour une précision de  $n$  bits, ce qui signifie que pour réaliser une acquisition à une fréquence  $F_s$ , un CAN SAR doit fonctionner à une fréquence  $n.F_s$ . Pour convertir le signal analogique en un code binaire, l'architecture SAR utilise le principe de dichotomie. À chaque étape le signal est comparé à une tension issue du

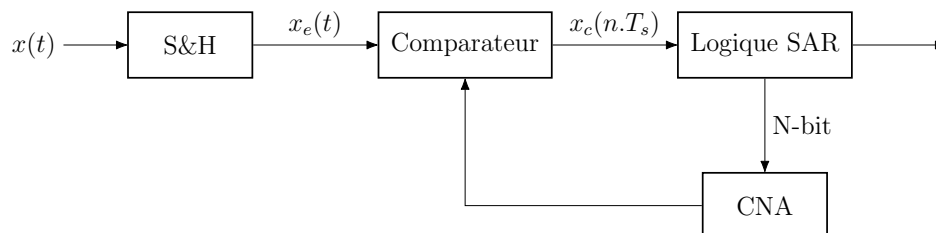


FIGURE 1.9 – Schéma de fonctionnement d'un CAN SAR

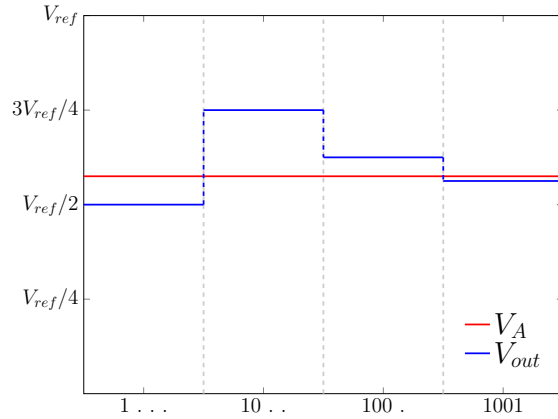


FIGURE 1.10 – Chronogramme d'une conversion sur 4 bits par une architecture SAR

CNA, piloté par le bloque logique de contrôle. Pour mieux comprendre son fonctionnement, on peut se référer à la relation suivante :

$$V_A = V_{ref} \left( \frac{b_1}{2} + \dots + \frac{b_n}{2^n} \right) = V_{ref} \sum_{i=1}^n \frac{b_i}{2^i} \quad (1.10)$$

Dans un premier temps, l'échantillonneur bloqueur copie la valeur de  $V_A$  et maintient cette dernière constante durant la conversion. On commence par déterminer le bit de poids fort (MSB), qui est dans un premier temps initialisé à 1. On compare  $V_A$  à  $V_1 = V_{ref}/2$  qui est la tension générée par le CNA à l'instant 1. Si  $V_A$  est supérieur à  $V_{ref}/2$ , on sait que  $b_1 = 1$  et donc qu'il faut ajouter  $V_{ref}/4$ , sinon on sait que  $b_1 = 0$  et on passe au bit suivant. Pour déterminer  $b_2$ , on se base sur le résultat précédent : on initialise dans un premier temps  $b_2$  à 1, on a donc  $V_2 = b_1(V_{ref}/2) + V_{ref}/4$ . Si  $V_A \geq V_2$  alors  $V_2 = 3V_{ref}/4$  et si  $V_A < V_2$  alors  $V_2 = V_{ref}/4$ . On compare  $V_A$  à  $V_2$ , si  $V_A \geq V_2$ , on sait que  $b_2 = 1$ , sinon on sait que  $b_1 = 0$  et on passe au bit suivant. Cette opération est ainsi répétée jusqu'au bit de poids faible (LSB). La Figure 1.10 montre un exemple d'un chronogramme de conversion sur 4 bits. Les caractéristiques de plusieurs CAN SAR sont présentées dans le Tableau 1.4.

TABLE 1.4 – Caractéristiques de CAN SAR

Réf.	Technologie (nm)	Surface (mm <sup>2</sup> )	ENOB (bits)	P (nW)	F <sub>s</sub> (kHz)	FoM (fJ/conv-step)
[29]	65	0.04	9.1	38	200	0.35
[30]	90	0.035	9.5	187	600	0.4
[31]	90	0.059	11.9	638	270	0.6
[32]	40	0.065	8.9	84	200	0.9
[33]	40	0.014	11.2	510	200	1.1

### 1.2.3.4 CAN Sigma-Delta ( $\Sigma\Delta$ )

Le principe de fonctionnement des architectures Sigma-Delta ( $\Sigma\Delta$ ) [34, 35, 36] repose sur le principe de sur-échantillonnage. On définit ainsi le facteur de sur-échantillonnage (OSR) :

$$OSR = \frac{F_s}{2 \cdot B} \quad (1.11)$$

où  $B$  est la bande passante du signal à acquérir. La Figure 1.11 montre l'architecture d'un CAN  $\Sigma\Delta$  de 1 bit, cette architecture est composée d'un système en boucle fermée qui contient un soustracteur, un intégrateur, un comparateur et un CNA de 1 bit et permet de réaliser le sur-échantillonnage. Un filtre numérique (décimateur) assure ensuite une restitution du code binaire à la fréquence de Nyquist et élimine le bruit en dehors de la bande de fréquence  $B$ .

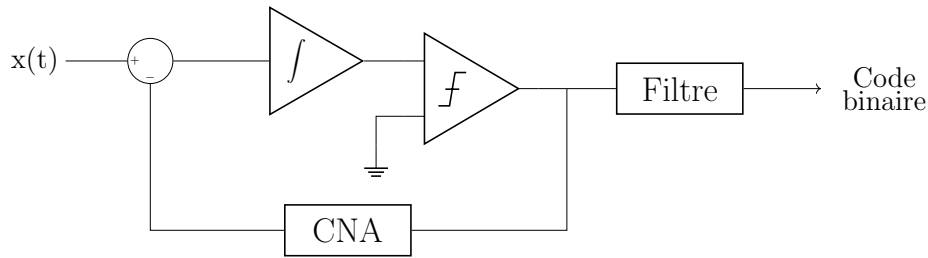


FIGURE 1.11 – Schéma de fonctionnement d'un CAN  $\Sigma\Delta$  1 bit

Le sur-échantillonnage du signal analogique, permet de minimiser la puissance du bruit de quantification dans la bande passante  $B$  du signal par rapport à une acquisition à la fréquence de Nyquist et ainsi d'augmenter le SNR. En effet, la densité spectrale du bruit de quantification  $DSP_e$  peut s'exprimer par la relation :

$$DSP_e = \frac{q^2}{12 \cdot F_s} \quad (1.12)$$

On sait également que le SNR maximal d'un CAN fonctionnant à la fréquence de Nyquist peut s'exprimer en fonction du nombre de bit de quantification  $n$  par la relation :

$$SNR_{dB} \approx 6.02n + 1.76 \quad (1.13)$$

En revanche, si le signal est sur-échantillonné, la DSP du bruit de quantification  $DSP_e$  devient :

$$DSP_e = \frac{q^2}{12 \cdot OSR \cdot F_s} \quad (1.14)$$

On peut constater l'amélioration du SNR lorsque le signal est sur-échantillonné, comme montré dans la Figure 1.12, grâce à la nouvelle expression du SNR :

$$SNR_{dB} \approx 6.02n + 1.76 + 10 \log_{10}(OSR) \quad (1.15)$$



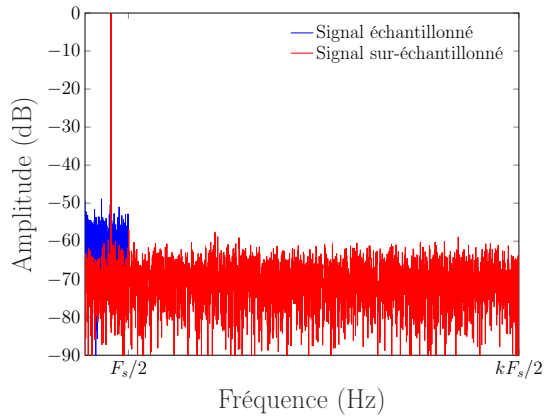


FIGURE 1.12 – Effet du sur-échantillonnage sur le niveau du bruit de quantification

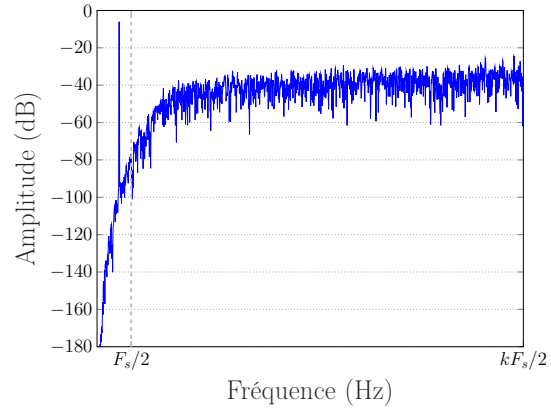


FIGURE 1.13 – Effet du Noise Shaping sur le spectre du signal

Pour réduire encore plus le bruit dans la bande de fréquence utile  $B$  et continuer à augmenter le SNR, on exploite le principe de modulation  $\Sigma\Delta$ . Il permet de rejeter le bruit à basses fréquences et de les rejeter dans les hautes fréquences et donc hors de la bande utile  $B$ . Cette technique est appelée Noise Shaping et est illustrée à la Figure 1.13. Après une opération de filtrage, le bruit en dehors de la bande de fréquence  $B$  est éliminé et le fonctionnement du système est ramené à la fréquence de Nyquist. Ce type de CAN permet d'obtenir une très bonne résolution, mais ne peut fonctionner que pour des signaux avec une bande passante moyenne. Le facteur de sur-échantillonnage, l'ordre de modulation et la résolution du quantificateur (dans le cas de la Figure 1.11 le quantificateur est un quantificateur 1 bit, soit un comparateur) sont les différents paramètres qui permettent de régler la résolution et le SNR d'un CAN  $\Sigma\Delta$ . Le Tableau 1.5 présente les caractéristiques de plusieurs convertisseurs de type  $\Sigma\Delta$ . De la même manière que pour les autres architectures de CAN, les exemples cités ici sont issus de l'étude de B. Murmann [23] et sont ceux ayant le meilleur facteur de mérite (FoM). La fréquence  $F_{Nyq}$  évoqué dans le tableau correspond à la fréquence de Nyquist, la fréquence de fonctionnement pouvant être obtenue par multiplication avec l'OSR.

 TABLE 1.5 – Caractéristiques de CAN  $\Sigma\Delta$ 

Réf.	Techno. (nm)	Surface (mm <sup>2</sup> )	ENOB (bits)	P (mW)	OSR	$F_{Nyq}$ (MHz)	FoM (fJ/conv)
[34]	28	0.025	12.6	3.2	30	80	6.5
[35]	40	0.053	15.2	4.5e-3	40	0.01	11.6
[36]	40	0.029	11.4	1.16	20	25	17.1
[37]	12	0.0585	11.6	3.2	13.87	60	17.6
[38]	7	0.056	12	3.8	8	50	18.6

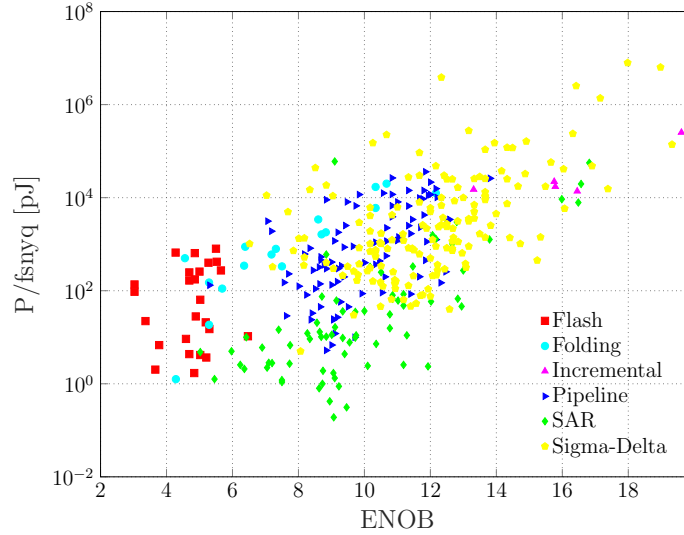


FIGURE 1.14 – Énergie de conversion (Puissance/Fréquence de Nyquist) en fonction de l'ENOB. Les données sont issues de l'étude de B. Murmann [23].

### 1.2.3.5 Conclusion sur les CAN

Le Tableau 1.6 compare les différentes architectures de CAN décrites précédemment. Le choix d'une architecture doit se faire en fonction des caractéristiques du signal à acquérir (fréquence d'acquisition, résolution) et des contraintes de l'application (consommation d'énergie). Le Tableau 1.6 peut être complété avec les données de l'étude de B. Murmann [23] regroupant les performances de plus de 550 publications parues entre 1997 et 2019. Grâce à la Figure 1.14, qui présente la quantité d'énergie nécessaire pour effectuer une conversion en fonction du nombre de bits effectif (ENOB) pour six familles de CAN, on constate que pour une application nécessitant une résolution moyenne et une faible consommation d'énergie, comme c'est le cas pour l'acquisition de signaux ECG, ce sont les CAN à approximations successives (SAR), présentés dans la Section 1.2.3.3, qui sont les plus adaptés.

TABLE 1.6 – Comparaison des différentes architectures de CAN

CAN	Résolution	Fréquence d'échantillonnage	Consommation
Flash	Faible	Très élevée	Très élevée
Pipeline	Élevée	Élevée	Élevée
SAR	Élevée	Faible	Faible
$\Sigma\Delta$	Très élevée	Moyenne	Élevée

### 1.2.4 Transmission de l'information

Il existe différents protocoles de communication pour l'IoT, chacun permettant des usages différents. Nous allons voir ici plusieurs d'entre eux : le Bluetooth Low Energy (BLE), les standards IEEE 802.15.4 et IEEE 802.15.6. Tous sont des standards de communication sans fils avec une transmission radio-fréquence.

Le BLE est une adaptation du Bluetooth standard, apparue en 2010, conçue pour fonctionner avec une très faible consommation d'énergie. Le standard Bluetooth en est actuellement à sa version 5.2 depuis décembre 2019 [39]. Le BLE utilise la bande de fréquence ISM (Industriel, Scientifique, Médical) à 2.4 GHz, qui est divisée en 40 canaux séparés de 2 MHz. Il prend en charge des débits de données allant de 125 kbit /s à 2 Mbit /s et plusieurs niveaux de puissance allant de 1 mW à 100 mW, permettant une communication sur des distances inférieures à 100 mètres dans un espace sans obstacles. Le BLE est capable de fonctionner dans une configuration de réseau en étoile, mais également dans une configuration de réseau maillé, aussi appelé mesh (Figure 1.15).

Dans un réseau en étoile, chaque nœud esclave ne peut communiquer qu'avec le nœud maître (également appelé concentrateur). Le concentrateur constitue le point névralgique d'un réseau en étoile, une défaillance à ce niveau rend le réseau inutilisable. Avec le BLE, le nombre d'esclaves maximum pour un maître n'est pas spécifié. Un réseau maillé est un réseau où tous les nœuds sont connectés les uns aux autres sans hiérarchie. Chaque nœud est donc capable de recevoir, envoyer et relayer des données. Cela permet de ne pas avoir de point névralgique dans le réseau comme pour un réseau en étoile.

Le standard 802.15.4 [40] est un standard de communication sans-fil défini par l'IEEE à faible débit, faible puissance, faible complexité, faible coût et à courte portée. Il est dédié aux réseaux de type *Low Rate Wireless Personal Area Network* (WPAN) et est notamment à la base du protocole ZigBee [41]. Ce standard peut être utilisé sur différentes bandes de fréquence : 16 canaux dans la bande de fréquence à 2.4 GHz, 10 canaux dans la bande de fréquence à 915 MHz et un canal dans la bande de fréquence à 868 MHz. Il prend en charge

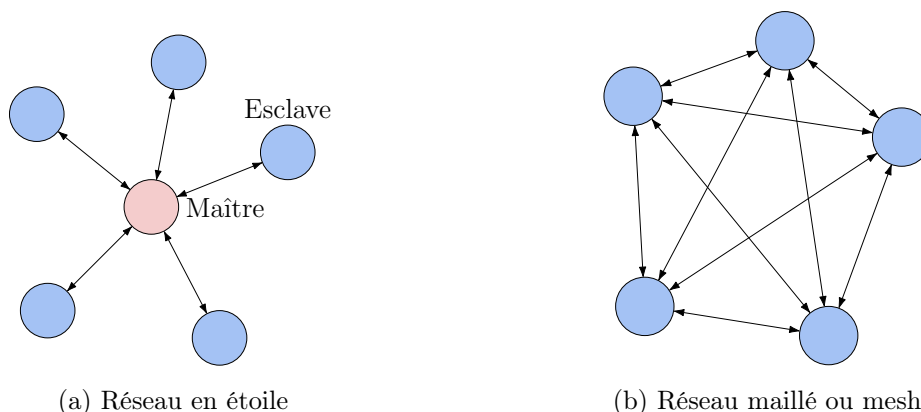


FIGURE 1.15 – Exemples de topologie de réseau

des débits de données allant de  $20 \text{ kbit s}^{-1}$  à  $250 \text{ kbit s}^{-1}$  et possède une portée d'une dizaine de mètres environ. Comme le BLE, ce standard est capable de fonctionner dans une configuration de réseau en étoile ou maillé.

Le standard 802.15.6 [52] est un autre standard de communication sans-fil défini par l'IEEE qui est destiné aux réseaux corporels (Wireless Body Area Network, WBAN). Par rapport au standard 802.15.4, sa consommation est réduite, mais également sa portée, qui ne dépasse pas deux mètres. Ce standard peut être configuré dans plusieurs modes : en mode bande étroite (Narrowband), en mode ultra large bande (Ultra Wideband, UWB) ou en mode Human body communication (HBC). Le mode bande étroite comprend sept bandes de fréquences différentes allant de la bande 402 MHz à 405 MHz, à la bande des 2.4 GHz et permet des débits allant de  $57.5 \text{ kbit s}^{-1}$  à  $971.4 \text{ kbit s}^{-1}$ . Le mode ultra large bande comprend deux bandes de fréquences 3.2 GHz à 4.7 GHz et 6.2 GHz à 10.3 GHz, et permet des débits allant de  $250 \text{ kbit s}^{-1}$  à  $15.6 \text{ Mbit s}^{-1}$ . Le mode Human body communication est basé sur le principe de communication par champ électrique où les propriétés diélectriques du corps humain sont utilisées pour créer un canal de communication. Les fréquences utilisées par ce mode se situent aux alentours des 21 MHz ce qui permet d'obtenir un débit compris entre 164 et  $1312 \text{ kbit s}^{-1}$ . Ce standard fonctionne avec une topologie de réseau en étoile.

### 1.2.5 Discussion

Le Tableau 1.7 résume les performances de différents systèmes de transmissions sans-fil utilisant les protocoles de communication présentés plus tôt. Il permet de constater le fort coût énergétique que représente la transmission d'informations. En effet, si l'on considère le CAN le

TABLE 1.7 – Caractéristiques de différents systèmes de transmissions sans-fil

Standard	Bande de fréquence	Technologie CMOS (nm)	Débit	Efficacité	Référence
BLE	2.4 GHz	28	1 Mbit/s	3.7 nJ/bit	[42]
BLE	2.4 GHz	55	1 Mbit/s	3.9 nJ/bit	[43]
BLE	2.4 GHz	65	1 Mbit/s	5.2 nJ/bit	[44]
802.15.4	2.4 GHz	180	250 kbit/s	93 nJ/bit	[45]
802.15.4	2.4 GHz	22	2 Mbit/s	200 pJ/bit	[46]
802.15.6	2.4 GHz	130	971 kbit/s	1.52 nJ/bit	[47]
802.15.6	400 MHz	180	187.5 kbit/s	12.1 nJ/bit	[48]
802.15.6 (HBC)	21 MHz	180	2625 kbit/s	660 pJ/bit	[49]
HBC	40 MHz à 120 MHz	180	10 Mbit/s	380 pJ/bit	[50]
HBC	13.56 MHz	65	100 kbit/s	210 pJ/s	[51]
	20 MHz à 60 MHz 140 MHz à 180 MHz		80 Mbit/s	32.5 pJ/bit	

plus énergivore [22] présenté dans la Section 1.2.3, dont le FoM est de 159.4 fJ/conversion, et le système de transmission [51] le plus efficace du Tableau 1.7, dont l'efficacité est de 32.5 pJ/bit, le coût de la transmission est plus de 200 fois supérieur à celui de la numérisation. Alors même que l'efficacité du transmetteur est considérée pour un bit et non pour un échantillon comme pour les CAN. Dans le cas inverse le plus extrême, c'est-à-dire avec le CAN de [29] et le transmetteur de [48], le coût de la transmission est  $3.4 \times 10^7$  fois supérieur à celui de la numérisation. On peut remarquer que le protocole HBC, bien qu'utilisant des technologies plus anciennes et donc potentiellement plus énergivores, à une meilleure efficacité énergétique. Comme évoqué plus tôt, la transmission des échantillons acquis au taux de Nyquist représente une part importante de la consommation globale du capteur, dans [1] celle-ci représente 96% de la consommation. Le système de transmission utilisé est un système de transmission radio ayant une efficacité énergétique de 40 nJ/bit et qui transmet à un débit de 43.2 kbit/s. Toutes ces données montrent bien que le budget énergétique de la transmission est le principal levier pour augmenter de façon significative l'autonomie des capteurs.

### 1.3 Échantillonnage compressif : conversion Analogique vers Information

Les convertisseurs Analogique vers Information (A2I) sont une technologie qui a vu le jour grâce à l'émergence d'une technique d'échantillonnage sous-Nyquist appelée échantillonnage compressif. L'objectif de ces convertisseurs est de tirer avantage du fait que certains signaux ont une bande passante physique qui est bien supérieure à leur taux d'information réel.

#### 1.3.1 Principe d'échantillonnage compressif

Dans le cas d'applications restreintes du point de vue du stockage ou de la consommation énergétique, on cherche alors à diminuer le nombre d'échantillons par compression : soit en effectuant un échantillonnage à la fréquence de Nyquist puis en réalisant une opération de compression, soit en faisant l'acquisition des données directement comprimées. Au lieu de générer  $N$  échantillons avec un échantillonnage uniforme, on effectue  $M$  mesures ( $M < N$ ) qui permettront par la suite de reconstruire le signal en se basant sur certaines propriétés du signal, c'est ce qu'on appelle l'échantillonnage compressif (CS) [2].

Soit  $\mathbf{y}$ , le vecteur, de taille  $M$ , des mesures réalisées :

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} + \mathbf{z} \tag{1.16}$$

où  $\mathbf{x}$  est le signal d'entrée de taille  $N$ ,  $\mathbf{\Phi}$  est la matrice de mesure de taille  $M \times N$  (modélisant un sous-échantillonnage) et  $\mathbf{z}$  représente le bruit (bruit thermique, bruit de quantification. . .) Le vecteur de mesures  $\mathbf{y}$  est alors transmis à un algorithme de reconstruction, or ce genre d'algorithme nécessite beaucoup de ressources. Dans le cas d'une application restreinte du

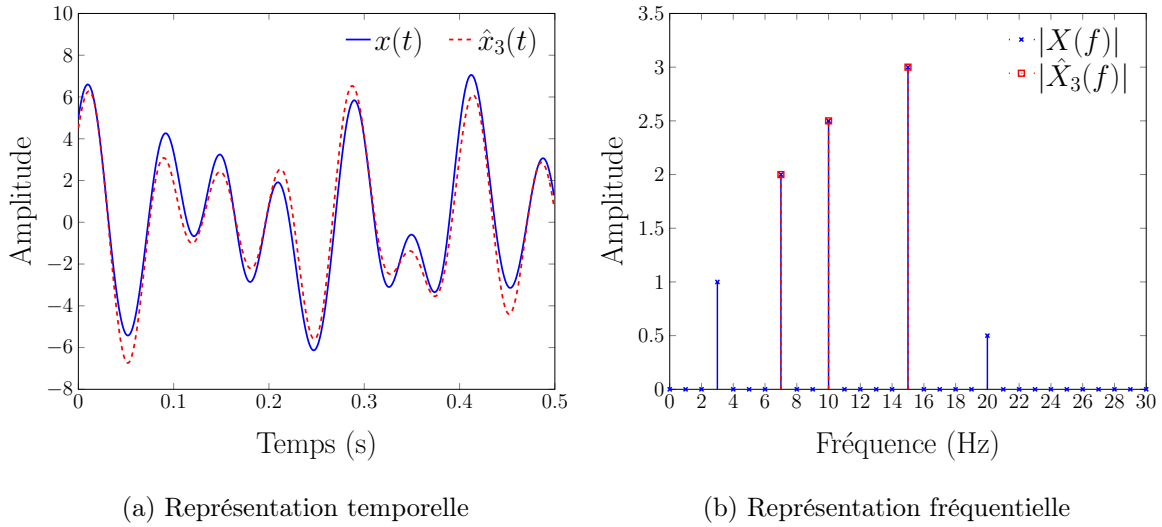


FIGURE 1.16 – Exemple de signal parcimonieux en fréquence

point de vue de la consommation d'énergie, le capteur va alors transmettre le vecteur  $\mathbf{y}$  à une unité de calcul qui va ensuite reconstruire le signal.

Les algorithmes de reconstruction de signaux compressés se basent sur la notion de parcimonie et sur la vérification de la propriété d'isométrie restreinte [53]. Un signal est dit  $K$ -parcimonieux dans une base (une famille de vecteurs est une base d'un espace vectoriel si et seulement si les vecteurs de cette famille sont linéairement indépendants et si tout vecteur de cet espace peut être décrit comme une combinaison linéaire des éléments de cette famille) ou un dictionnaire (où les vecteurs ne sont pas linéairement indépendants) s'il peut être décrit par  $K \ll M \ll N$  valeurs non-nulles dans cette base ou ce dictionnaire. La Figure 1.16 est un exemple de signal parcimonieux en fréquence. On peut alors écrire le vecteur  $\mathbf{x}$ , le signal d'entrée composé de  $N$  échantillons, de la manière suivante :

$$\mathbf{x} = \Psi \cdot \mathbf{s} \quad (1.17)$$

où  $\Psi$  est la matrice de changement de base,  $\Psi$  correspond à la matrice identité dans le domaine temporel, à la matrice de transformée de Fourier inverse dans le domaine de Fourier ou à la matrice de transformée en ondelettes inverse dans le domaine des ondelettes, et  $\mathbf{s}$  est la représentation du signal  $\mathbf{x}$  dans la base parcimonieuse, avec  $\|\mathbf{s}\|_0 \leq K$ , où  $\|\cdot\|_0$  correspond à la quasi-norme zéro, c'est-à-dire le nombre d'éléments non nuls dans le vecteur.

De (1.16) et (1.17), on peut alors dire que :

$$\mathbf{y} = \Phi \cdot \Psi \cdot \mathbf{s} + \mathbf{z} \quad (1.18)$$

En pratique, très peu de signaux sont parfaitement parcimonieux du fait de la présence de bruit dans les signaux. On réalise donc une étape de compression dans laquelle on cherche

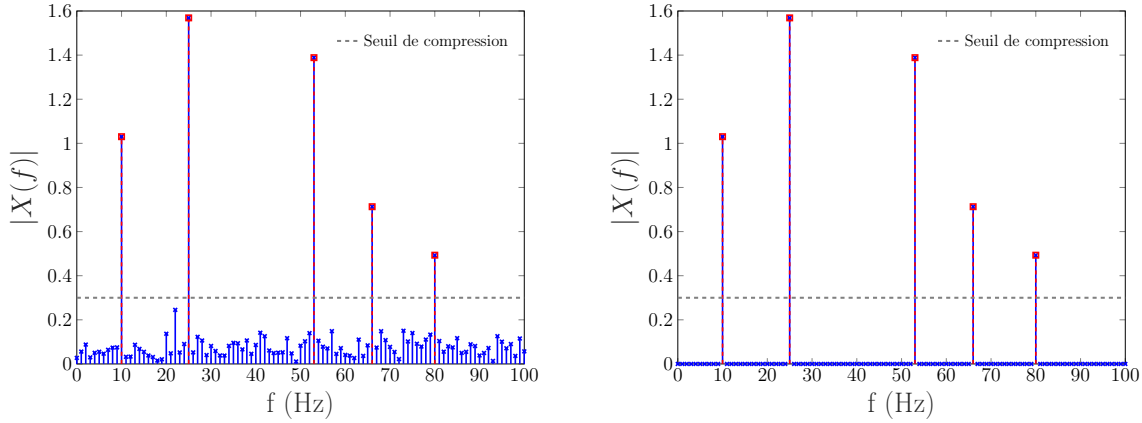


FIGURE 1.17 – Signal compressible

à sélectionner les coefficients significatifs dans la base parcimonieuse choisie et ignorer ceux proches de zéro, comme montré dans la Figure 1.17. Un signal est dit compressible si le module de ses coefficients, triés dans un ordre décroissant, dans la base  $\Psi$ , suivent une décroissance en loi de puissance [54] :

$$|s_i| \leq C \cdot i^{-q} \quad (1.19)$$

avec  $i \in [1; N]$  et  $C$  une constante. Plus  $q$  est grand, plus le signal est compressible. À partir du vecteur de mesures  $\mathbf{y}$ , on cherche une solution  $K$ -parcimonieuse parmi les solutions possibles de l'équation  $\mathbf{y} = \Phi \cdot \Psi \cdot \mathbf{s}$  qui satisfasse les critères d'unicité (solution unique) et de stabilité (de petites variations dans les observations doivent entraîner de petites variations lors de la récupération) tout en permettant une bonne reconstruction du signal d'entrée. Pour cela, il faut que la matrice  $\mathbf{A} = \Phi \cdot \Psi$  satisfasse la propriété d'isométrie restreinte (RIP).

$\mathbf{A}$  satisfait la RIP d'ordre  $K$ , si  $\exists \delta_k \in ]0; 1[ / \forall$  vecteur  $\mathbf{s}$ ,  $K$ -parcimonieux, on a :

$$(1 - \delta_k) \leq \frac{\|\mathbf{A} \cdot \mathbf{s}\|_2^2}{\|\mathbf{s}\|_2^2} \leq (1 + \delta_k) \quad (1.20)$$

où  $\|\cdot\|_2$  correspond à la norme deux. Cette constante  $\delta_k$  est appelée constante d'isométrie. Cette propriété permet de s'assurer que la matrice  $\mathbf{A}$  préserve le plus possible la norme d'un vecteur  $K$ -parcimonieux. Les matrices aléatoires ayant une forte probabilité de satisfaire cette propriété [55], elles sont souvent utilisées comme matrice de mesure  $\Phi$ . Pour assurer l'unicité et la stabilité de la solution de l'équation, il faut que  $\delta_k < \sqrt{2} - 1$  [53]. La propriété d'isométrie restreinte est difficile à démontrer en pratique, car c'est un problème NP difficile.

Pour contourner ce problème, on peut utiliser une autre approche basée sur la cohérence [56] entre la matrice de mesures  $\Phi$  et la matrice de changement de base  $\Psi$  qui s'exprime par la formule :

$$\mu(\Phi, \Psi) = \sqrt{N} \max_{1 \leq u, j \leq N} |\langle \phi_u, \psi_j \rangle| \quad \text{avec } 1 \leq \mu(\Phi, \Psi) \leq \sqrt{N} \quad (1.21)$$

avec  $\langle \cdot, \cdot \rangle$  l'opérateur de produit scalaire. La cohérence permet de mesurer la plus grande corrélation entre une ligne  $\phi_u$  de  $\Phi$  et une colonne  $\psi_j$  de  $\Psi$  et doit être le plus proche possible de 1. Plus la cohérence est faible, plus le nombre de mesures nécessaires à la reconstruction est faible et le facteur de compression est élevé. On peut citer comme exemple de bases incohérentes les bases temporelles et fréquentielles : une distribution de Dirac dans le domaine temporel est étalée dans le domaine fréquentiel et inversement un pic dans le domaine fréquentiel correspond à une onde sinusoïdale dans le domaine temporel. Le nombre minimum de mesures  $M$  nécessaires pour garantir une bonne reconstruction des signaux  $K$ -parcimonieux.

Pour garantir une bonne reconstruction du signal  $K$ -parcimonieux, il faut que :

$$M \geq C \cdot \mu^2(\Phi, \Psi) \cdot K \cdot \log(N) \quad (1.22)$$

où  $C$  est une constante connue dont la valeur dépend de la matrice de mesure [2]. Il faut donc minimiser  $\mu(\Phi, \Psi)$  pour diminuer le nombre de mesures nécessaires à la reconstruction et ainsi avoir un plus grand facteur de compression. Dans le cas où  $\Psi$  et  $\Phi$  sont totalement incohérentes ( $\mu(\Phi, \Psi) = 1$ ), le nombre minimal de mesure à avoir pour reconstruire le signal avec une bonne probabilité est :

$$M \geq C \cdot K \cdot \log(N) \quad (1.23)$$

On peut noter que les matrices de Bernoulli, composées aléatoirement des valeurs  $\pm 1$  selon une loi uniforme, ou Gaussienne, composées de valeurs choisies aléatoirement selon une loi normale en général de moyenne nulle, sont fortement incohérentes avec n'importe quelle base de parcimonie [53].

### 1.3.2 Architecture du convertisseur A2I

Un convertisseur A2I est un système matériel permettant la réalisation du CS. Il est composé d'un encodeur qui effectue l'opération compression, mathématiquement, il réalise l'opération de multiplication du signal  $\mathbf{x}$  par la matrice  $\Phi$ . L'encodeur est suivi d'un CAN qui transforme la grandeur analogique compressée en donnée numérique. La Figure 1.18 représente le schéma d'un capteur utilisant un convertisseur A2I. Nous verrons dans cette section, différentes architectures de convertisseurs A2I pour la mise en place du CS, puis nous aborderons différents algorithmes de reconstruction utilisés et parlerons des limitations de cette technique.

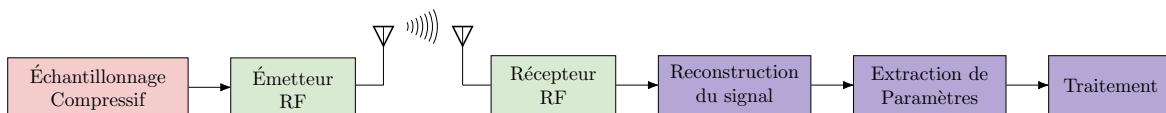


FIGURE 1.18 – Capteur sans-fil utilisant le CS



### 1.3.2.1 Échantillonnage non uniforme (NUS)

L'échantillonnage non-uniforme, présenté dans [57, 58, 59], consiste à échantillonner le signal à des intervalles de temps non-uniformes, en sélectionnant un sous ensemble des échantillons normalement acquis au taux de Nyquist, par le biais d'une horloge (pseudo) aléatoire. L'architecture de l'échantillonnage non-uniforme est présentée dans la Figure 1.19, elle est constituée d'un échantillonneur bloqueur (S&H) et d'un convertisseur analogique-numérique (CAN) pilotés par une horloge pseudo-aléatoire. On peut souligner le fait qu'aujourd'hui, la plupart des CAN possèdent déjà un bloc S&H.

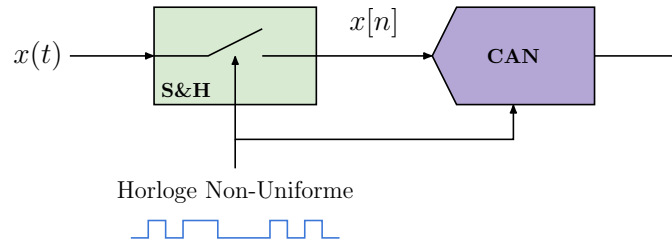


FIGURE 1.19 – Architecture d'un échantillonnage non uniforme

Pour cette architecture, on a donc la matrice de mesure  $\Phi$ , de taille  $M \times N$ , qui est un sous ensemble  $\Omega$  de lignes de la matrice identité  $\mathbf{I}_N$ , avec  $|\Omega| = M$ .

$$\Phi_{NUS} = R_{\Omega} \mathbf{I}_N \quad (1.24)$$

où  $|\Omega| = M$  est la cardinalité de  $\Omega$ ,  $R_{\Omega}$  est un opérateur de restriction qui permet de sélectionner un sous-ensemble  $\Omega$  de lignes de la matrice identité. Cette architecture est la plus simple pour réaliser un échantillonnage compressif, mais elle nécessite de pouvoir fonctionner au taux de Nyquist, ce qui est coûteux du point de vue énergétique. De plus, elle n'est adaptée qu'aux signaux parcimonieux dans le domaine fréquentiel.

### 1.3.2.2 Démodulateur aléatoire

Le principe du démodulateur aléatoire (Random Demodulator, RD), présenté dans la Figure 1.20, est de multiplier le signal entrant  $x(t)$  par une séquence pseudo-aléatoire pouvant prendre comme valeurs -1 ou 1 et qui oscille à une fréquence supérieure ou égale au taux de Nyquist. Le but de cette étape de démodulation est d'étaler le contenu fréquentiel du signal de sorte qu'il ne soit pas détruit par l'intégrateur qui se situe en aval. Le signal est ensuite échantillonné à une fréquence  $F_s$  bien inférieure à la fréquence de Nyquist [60, 61, 62]. L'intégrateur est remis à zéro à chaque période  $T_s$ .

On peut exprimer la matrice de mesure  $\Phi$  par le produit de deux matrices :

- Une matrice diagonale  $\mathbf{D}$  de taille  $N \times N$ , modélisant la séquence pseudo-aléatoire de valeur  $\pm 1$

- Une matrice  $\mathbf{L}$  diagonale par blocs de taille  $M \times N$ , chaque bloc étant un vecteur, de  $N/M$  valeurs, composé uniquement de 1, modélisant l'intégrateur.

$$\mathbf{D} = \begin{pmatrix} \pm 1 & 0 & \cdots & 0 \\ 0 & \pm 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \pm 1 \end{pmatrix} \quad \mathbf{L} = \begin{pmatrix} 1 \ 1 \ \cdots \ 1 & 0_{[1, N/M]} & \cdots & 0_{[1, N/M]} \\ 0_{[1, N/M]} & 1 \ 1 \ \cdots \ 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0_{[1, N/M]} \\ 0_{[1, N/M]} & \cdots & 0_{[1, N/M]} & 1 \ 1 \ \cdots \ 1 \end{pmatrix}$$

On a alors :

$$\mathbf{y} = \Phi_{RD} \cdot \mathbf{x} = \mathbf{L} \cdot \mathbf{D} \cdot \mathbf{x} \quad (1.25)$$

La matrice 1.26 est un exemple de ce que peut être la matrice de mesure  $\Phi$  d'un convertisseur A2I utilisant cette architecture avec une configuration de  $M = 4$  et  $N = 16$  :

$$\Phi_{RD} = \begin{pmatrix} 1 - 1 \ 1 \ 1 & 0_{[1,4]} & 0_{[1,4]} & 0_{[1,4]} \\ 0_{[1,4]} & -1 - 1 \ 1 - 1 & 0_{[1,4]} & 0_{[1,4]} \\ 0_{[1,4]} & 0_{[1,4]} & 1 \ 1 \ 1 - 1 & 0_{[1,4]} \\ 0_{[1,4]} & 0_{[1,4]} & 0_{[1,4]} & 1 - 1 \ 1 - 1 \end{pmatrix} \quad (1.26)$$

Le RD a l'avantage de permettre de réduire les contraintes du CAN, en termes de fréquence d'échantillonnage, par rapport à la méthode de l'échantillonnage non-uniforme. En revanche, il est nécessaire d'avoir un générateur de séquence aléatoire qui doit pouvoir fonctionner au taux de Nyquist.

Le Pré-intégrateur à modulation aléatoire (Random Modulator Pre-Integrator, RMPI) est une extension de l'architecture précédente. Alors que le démodulateur aléatoire n'est constitué que d'une seule branche, cette nouvelle architecture est composée d'une association de  $N_c \leq M < N$  démodulateurs aléatoires en parallèle. Cette architecture permet donc de calculer plusieurs coefficients  $y_m(t)$  en même temps et à partir de coefficients d'entrée communs, ce qui n'était pas possible avec le démodulateur aléatoire [4, 63].

La matrice de mesure  $\Phi$  du RMPI est une matrice diagonale par blocs, où chaque bloc est une matrice  $B$  de taille  $N_c \times \frac{N_c \cdot N}{M}$  composée des valeurs -1 et 1.

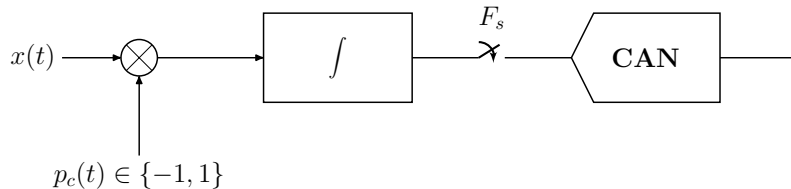


FIGURE 1.20 – Architecture du Démodulateur aléatoire

$$\Phi_{RMPI} = \begin{pmatrix} B_{[N_c \times \frac{N_c \cdot N}{M}]} & 0_{[N_c \times \frac{N_c \cdot N}{M}]} & \cdots & 0_{[N_c \times \frac{N_c \cdot N}{M}]} \\ 0_{[N_c \times \frac{N_c \cdot N}{M}]} & B_{[N_c \times \frac{N_c \cdot N}{M}]} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0_{[N_c \times \frac{N_c \cdot N}{M}]} \\ 0_{[N_c \times \frac{N_c \cdot N}{M}]} & \cdots & 0_{[N_c \times \frac{N_c \cdot N}{M}]} & B_{[N_c \times \frac{N_c \cdot N}{M}]} \end{pmatrix} \quad (1.27)$$

Le RMPI, grâce au calcul de coefficients en parallèle, permet d'augmenter le temps d'intégration, par rapport au RD, sans forcément augmenter le temps nécessaire pour obtenir l'ensemble des coefficients du vecteur de mesure  $\mathbf{y}$ .

Dans la Figure 1.21, l'architecture a été présentée avec un CAN par branche, on peut alors se demander si l'on souhaite avoir  $N_c$  convertisseurs fonctionnant à une fréquence inférieure à la fréquence de Nyquist ou alors avoir un unique CAN, couplé à un multiplexeur, fonctionnant à une fréquence beaucoup plus élevée. Cela dépend des contraintes en matière de consommation, de surface ou de bande passante.

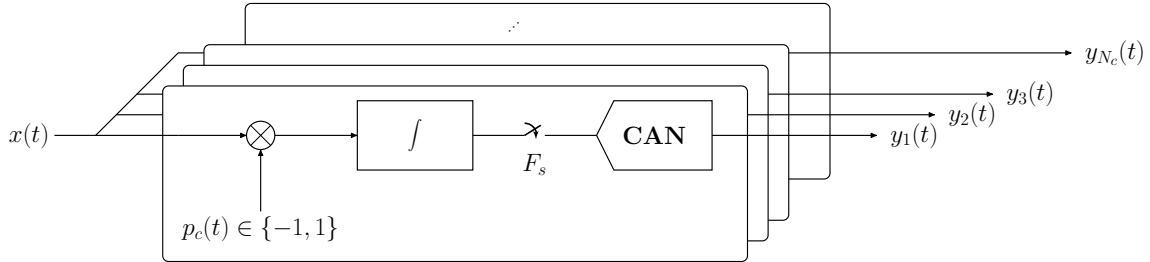


FIGURE 1.21 – Architecture du RMPI

### 1.3.2.3 Convertisseur large bande modulé

La modulation d'un signal avec une séquence pseudo-aléatoire, pour la compression, ne convient que pour des classes de signaux très spécifiques [64]. Le principal avantage du convertisseur large bande modulé (Modulated Wideband Converter, MWC) est de réduire les contraintes sur la bande passante du S&H, pour fonctionner à des taux inférieurs à ceux du taux de Nyquist. Un MWC est constitué de  $M$  branches. Dans chaque branche, le signal d'entrée  $x(t)$ , de bande passante  $B$ , est d'abord multiplié par un signal  $p_i(t)$ , de période  $T_p = 1/f_p$  et décomposable en une série de Fourier. Les signaux  $p_i(t)$  sont des répétitions périodiques de séquences pseudo-aléatoires. Puis le signal est filtré grâce à un filtre passe-bas de fréquence de coupure  $f_c = 1/(2.T_s)$ . Le signal filtré est alors échantillonné à la fréquence  $F_s$ , avec  $2B \geq F_s \geq f_p \geq B$ .

Cette architecture est semblable au démodulateur aléatoire et au pré-intégrateur à modulation aléatoire. La principale différence entre le RD et le MWC se situe au niveau du filtre passe-bas [65, 66]. Dans le RD, on échantillonne le signal  $x(t)$ , à une fréquence  $F_s = 1/T_s$ ,

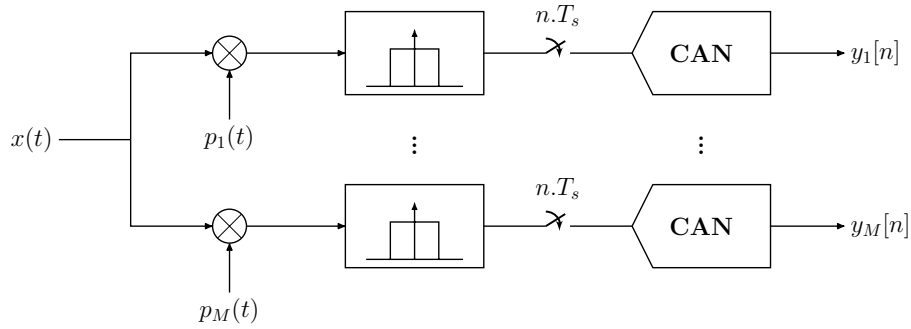


FIGURE 1.22 – Architecture du Convertisseur large bande modulé

sur une période d'observation  $T$ . On considère que  $x(t)$  est un signal multiton, qui peut être décomposé en une série de Fourier sur l'intervalle  $[0, T]$ . Le filtre utilisé dans l'architecture RD est un intégrateur considéré comme idéal avec une réponse impulsionnelle de la forme :

$$h(t) = \Pi\left(\frac{2M}{T}t - 1\right) \quad \text{avec } \Pi(x) = \begin{cases} 1 & \text{si } x \in [-1; 1] \\ 0 & \text{sinon} \end{cases} \quad (1.28)$$

où  $T_s = T/M$  est la période d'échantillonnage et  $T$  est la période d'observation (la période d'échantillonnage est choisie pour être  $M$  fois inférieure à la fenêtre d'observation). Dans le cas de l'architecture MWC, on considère que  $x(t)$  est un signal multibande d'énergie finie, c'est-à-dire que sa transformée de Fourier  $X(\omega)$  est à support borné :

$$X(\omega) = 0, \quad \text{si } |\omega| \geq \pi \cdot W \quad (1.29)$$

avec  $\pi \cdot W$  la bande passante du signal  $x(t)$  et  $2\pi \cdot W$  est la fréquence de Nyquist en radian par seconde. Le filtre utilisé est un filtre passe-bas idéal avec une réponse impulsionnelle de la forme :

$$h'(t) = \frac{\pi \cdot W}{L} \operatorname{sinc}\left(\frac{\pi \cdot W}{L}t\right) \quad (1.30)$$

Le convertisseur large bande modulé est une architecture adaptée pour des signaux multibandes et permet de réduire la bande passante nécessaire lors de la phase d'échantillonnage (CAN et S&H). Le problème est qu'il est nécessaire d'allouer des ressources pour stocker et générer la séquence aléatoire du signal  $p_i(t)$ .

#### 1.3.2.4 Échantillonnage non-uniforme en ondelettes

Le principe de l'échantillonnage non-uniforme en ondelettes (Non-Uniform Wavelet Sampling, NUWS) est d'effectuer un échantillonnage non-uniforme de coefficients de transformées en ondelettes [57, 67]. Une ondelette  $\xi$  est une fonction appartenant à l'ensemble des fonctions de carré intégrable, c'est-à-dire que l'intégrale  $\int^{\mathbb{R}} |\xi(t)|^2 dt$  est convergente. La fonction  $\xi$  est

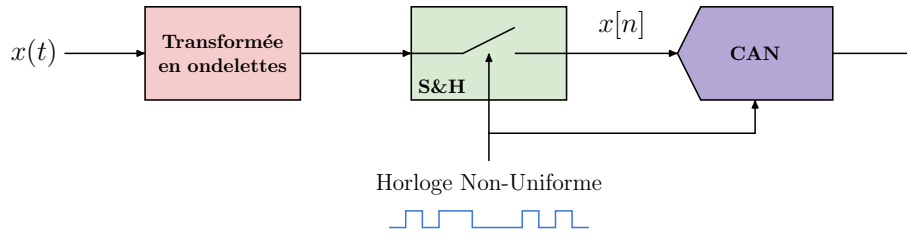


FIGURE 1.23 – Schéma de l'architecture d'un échantillonnage non uniforme en ondelettes

définie sur un support borné, oscillante et de moyenne nulle. On définit une famille d'ondelette par la formule :

$$\forall t \in \mathbb{R}, \quad \xi_{a,b}(t) = \frac{1}{\sqrt{a}} \cdot \xi \left( \frac{t-b}{a} \right) \quad (1.31)$$

avec  $\psi$  l'ondelette mère (ondelette de Haar, de Morlet, dérivées de Gaussienne, de Daubechies...), un facteur d'échelle  $a > 0$  (donnant la taille du support) et  $b \in \mathbb{R}$  (donnant la position). On définit alors la transformée en ondelette par la formule :

$$\mathcal{W}f(a, b) = \int_{\mathbb{R}} f(t) \cdot \xi_{a,b}^*(t) dt \quad (1.32)$$

où  $\xi_{a,b}^*(t)$  est le conjugué de  $\xi_{a,b}(t)$ . En pratique, la transformée en ondelette continue peut être vue, pour  $a$  fixé, comme un produit de convolution :

$$\mathcal{W}f(a, b) = \frac{1}{\sqrt{a}} \int_{\mathbb{R}} f(t) \cdot \xi^* \left( \frac{t-b}{a} \right) dt \quad (1.33)$$

$$\mathcal{W}f(a, b) = (f * \check{\xi}_a)(b) \quad \text{avec } \check{\xi}_a(t) = \frac{1}{\sqrt{a}} \cdot \xi \left( -\frac{t}{a} \right) \quad (1.34)$$

En comparaison avec un échantillonnage non-uniforme classique, l'échantillonnage non uniforme en ondelettes consiste à effectuer une transformée en ondelette continue du signal d'entrée  $x(t)$  avant d'effectuer un échantillonnage non uniforme (Figure 1.23). Pour réaliser cette transformée en ondelette continue, on multiplie le signal d'entrée par une ondelette puis on intègre le résultat sur la durée du support de cette ondelette (Equation 1.32).

La transformée en ondelette permet d'obtenir des informations à la fois temporelle et fréquentielle sur le signal. Plus le support de l'ondelette sera large, plus l'information en fréquence sera précise. À l'inverse avec un support très court, l'information sera précise en temps au détriment de la précision en fréquence. Contrairement à l'échantillonnage non uniforme qui n'a qu'un degré de liberté (les instants d'échantillonnage), on a, avec cette architecture, trois degrés de liberté : les instants d'échantillonnage, la taille du support temporel de l'ondelette et la fréquence d'oscillation. En reprenant les équations de l'échantillonnage compressif en temps discret, on peut dire que la matrice de mesure  $\Phi$  est composée d'un ensemble  $\Omega$ , avec  $|\Omega| = M$ ,

de ligne de la matrice  $\mathcal{W}$ , qui est qui est une base d'ondelette où chaque ligne correspond à une ondelette différente obtenue grâce à l'équation 1.33. On a donc :

$$\mathbf{y} = R_\Omega \mathcal{W} \mathbf{x} + \mathbf{z}, \quad (1.35)$$

où  $R_\Omega$  est un opérateur de restriction qui permet de sélectionner un sous-ensemble  $\Omega$  de lignes de la matrice  $\mathcal{W}$  contenant l'ensemble des ondelettes. Dans l'échantillonnage non uniforme en ondelettes (NUWS), on sous-échantillonne la matrice  $\mathcal{W}$ . Chaque coefficient du vecteur de mesures  $\mathbf{y}$  est donc le résultat du produit scalaire entre une ondelette et le vecteur  $\mathbf{x}$ , le signal entrant.

$$y_i = \langle \mathbf{w}_i, \mathbf{x} \rangle + z_i, \quad i \in \Omega \quad (1.36)$$

Le NUWS offrant plusieurs degrés de liberté, le nombre d'ondelettes, et donc de mesures, potentiel est extrêmement grand. On va donc chercher à constituer un ensemble d'ondelettes de taille  $M \ll N \ll L$ , où  $L$  est le nombre d'ondelettes dans la matrice  $\mathcal{W}$ , permettant une bonne reconstruction du signal ou d'obtenir de bonnes performances de classification. Pour choisir ces  $M$  ondelettes, il faut trouver des critères permettant de réduire l'ensemble de départ et choisir une méthode de sélection permettant de maximiser les résultats. On peut par exemple utiliser comme critères : la famille d'ondelettes (Gabor, Morlet, Haar...), de sorte que celle-ci soit facile à générer au niveau matériel, ou encore, réduire l'ensemble des valeurs de facteur d'échelle ( $a \in \mathbb{R}^{*+}$ ) et de décalage temporel ( $b \in \mathbb{R}$ ).

L'objectif du CS est d'offrir une solution permettant de combiner les étapes de compression et d'acquisition d'un flux de données. On peut alors définir comme critère d'évaluation, le taux de compression qui correspond au rapport entre le nombre d'échantillons que l'on aurait acquis au taux de Nyquist et le nombre d'échantillons acquis grâce au CS. Pour réaliser cette acquisition compressée, on utilise certaines propriétés du signal comme le degré de parcimonie. En se basant sur cette propriété, il est ensuite possible de retrouver, ou au moins approximer, le signal de départ grâce à des algorithmes de reconstruction.

### 1.3.3 Algorithmes de reconstruction

Nous avons vu précédemment que le vecteur de mesures  $\mathbf{y}$  pouvait être exprimé en fonction du signal dans la base parcimonieuse via la relation :  $\mathbf{y} = \Phi \cdot \Psi \cdot \mathbf{s} + \mathbf{z}$ . Le but d'un algorithme de reconstruction est d'aboutir à une représentation fidèle du signal de départ. Pour cela, il doit trouver une solution à l'équation précédente, c'est-à-dire retrouver une solution pour  $\mathbf{s}$  à partir de  $\mathbf{y}$  et de  $\mathbf{A} = \Phi \cdot \Psi$ . Or ce système est sous-déterminé, car c'est un système de  $M$  équations à  $N \gg M$  inconnues. Le problème se résume alors à trouver une solution parcimonieuse avec un nombre d'éléments non-nuls  $K$ , le plus petit possible et qui soit solution de l'équation de l'échantillonnage compressif.

$$\tilde{\mathbf{s}} = \arg \min_{\hat{\mathbf{s}} \in \mathbb{R}^N} \|\hat{\mathbf{s}}\|_0 \quad \text{avec } \mathbf{A} \cdot \hat{\mathbf{s}} = \mathbf{y} \quad (1.37)$$

où  $\|\cdot\|_0$  représente la norme  $\ell_0$ , c'est-à-dire au nombre d'éléments non-nul du vecteur,  $\hat{\mathbf{s}}$  est une des solutions  $K$ -parcimonieuses de l'équation  $\mathbf{A} \cdot \hat{\mathbf{s}} = \mathbf{y}$  et  $\tilde{\mathbf{s}}$  est la solution la plus parcimonieuse de  $\hat{\mathbf{s}}$ . Pour résoudre ce problème, nous allons à présent voir deux types d'algorithmes : les algorithmes d'optimisation convexe et les algorithmes gloutons.

### 1.3.3.1 Algorithmes d'optimisation convexe

La résolution du problème (1.37) étant difficile à résoudre en pratique (problème NP difficile), il a été démontré qu'une solution à ce problème pouvait être trouvée en remplaçant la norme  $\ell_0$  par la norme  $\ell_1$  avec un nombre de mesures  $M$  suffisant [53]. Cela est possible si la matrice  $\mathbf{A}$  vérifie la propriété d'isométrie restreinte ou si la matrice  $\mathbf{A}$  est incohérente ( $\mu(\Phi, \Psi) = 1$ ). Le principe de cette solution appelée Basis Pursuit (BP) [68] est donc de trouver une solution au problème suivant :

$$\tilde{\mathbf{s}} = \arg \min_{\hat{\mathbf{s}} \in \mathbb{R}^N} \|\hat{\mathbf{s}}\|_1 \quad \text{avec } \mathbf{A} \cdot \hat{\mathbf{s}} = \mathbf{y} \quad (1.38)$$

Dans le BP, si la solution n'est pas parfaitement parcimonieuse, on ne peut pas satisfaire la condition  $\mathbf{A} \cdot \hat{\mathbf{s}} = \mathbf{y}$ . Il faut donc prendre en compte le bruit de mesure ( $\mathbf{y} - \mathbf{A} \cdot \hat{\mathbf{s}}$ ).

Lorsque les données sont bruitées et donc imparfaites, on utilise le Basis pursuit de-noising (BPDN) [3, 68, 69, 70], on cherche la solution la plus parcimonieuse, en autorisant une certaine quantité de bruit  $\epsilon$ .

$$\tilde{\mathbf{s}} = \arg \min_{\hat{\mathbf{s}} \in \mathbb{R}^N} \|\hat{\mathbf{s}}\|_1 \quad \text{avec } \|\mathbf{y} - \mathbf{A} \cdot \hat{\mathbf{s}}\|_2 < \epsilon \quad (1.39)$$

Le problème précédent peut également être énoncé sous la forme suivante (One-norm regularization) :

$$\tilde{\mathbf{s}} = \arg \min_{\hat{\mathbf{s}} \in \mathbb{R}^N} (\|\mathbf{y} - \mathbf{A} \cdot \hat{\mathbf{s}}\|_2^2 + \lambda \cdot \|\hat{\mathbf{s}}\|_1) \quad (1.40)$$

Le paramètre  $\lambda$  permet de faire un compromis entre parcimonie et fidélité de la reconstruction, en pénalisant, plus ou moins, les solutions les moins parcimonieuses. Ces algorithmes sont des algorithmes d'optimisation convexe qui cherchent une solution optimale au prix de la complexité et de la lenteur.

### 1.3.3.2 Algorithmes Gloutons

Les algorithmes gloutons sont des algorithmes dits de décomposition parcimonieuse, dont l'objectif est de reconstruire le signal de manière itérative. À chaque itération, l'algorithme construit une approximation  $\mathbf{y}_k$  du signal  $\mathbf{y}$  et évalue l'erreur entre le signal construit et

le signal approximé, qu'on appelle résidu. L'algorithme de cette famille le plus connu est le Matching Pursuit (MP) [71, 72].

Dans le MP, on cherche à exprimer le vecteur de mesures  $\mathbf{y}$  comme une combinaison linéaire d'éléments de la matrice  $\mathbf{A} = \Phi \cdot \Psi$ . Au départ,  $\mathbf{r}_0 = \mathbf{y}$  et  $\hat{\mathbf{y}}_0 = 0$ . À chaque itération, on cherche la colonne de  $\mathbf{A}$  qui est la plus corrélée au résidu  $\mathbf{r}_{k-1}$ .

$$\lambda_k = \arg \max_{i \in [1; M]} |\langle \mathbf{r}_{k-1}, \mathbf{a}_i \rangle| \quad (1.41)$$

où  $\lambda_k$  indique l'indice de la colonne la plus corrélée à  $\mathbf{y}$ ,  $\mathbf{r}_{k-1}$  représente l'erreur à l'étape  $k-1$  et  $\mathbf{a}_i$  représente la colonne  $i$  de la matrice  $\mathbf{A}$ . On met alors à jour l'estimation  $\hat{\mathbf{y}}_k$  du signal  $\mathbf{y}$  et du résidu  $\mathbf{r}_k$ .

$$\hat{\mathbf{y}}_k = \hat{\mathbf{y}}_{k-1} + \langle \mathbf{r}_{k-1}, \mathbf{a}_{\lambda_k} \rangle \cdot \mathbf{a}_{\lambda_k} \quad (1.42)$$

$$\mathbf{r}_k = \mathbf{y} - \hat{\mathbf{y}}_k \quad (1.43)$$

L'estimation de  $\mathbf{y}$  à l'étape  $k$  ( $\hat{\mathbf{y}}_k$ ) peut également s'exprimer de la forme :

$$\hat{\mathbf{y}}_k = \sum_{l=1}^k \langle \mathbf{r}_{l-1}, \mathbf{a}_{\lambda_l} \rangle \cdot \mathbf{a}_{\lambda_l} = \sum_{l=1}^k \mathbf{s}_{\lambda_l} \cdot \mathbf{a}_{\lambda_l} \quad (1.44)$$

On réitère ces opérations jusqu'à ce que le critère d'arrêt soit atteint (erreur résiduelle inférieure à un seuil donné, nombre maximum d'itérations, temps de calcul...). L'un des inconvénients de cet algorithme est que l'on peut sélectionner une colonne  $\mathbf{a}_i$  de  $\mathbf{A}$  (également appelé atome) lors de plusieurs itérations de l'algorithme. Dans l'Orthogonal Matching Pursuit (OMP), le vecteur résiduel  $\mathbf{r}_k$  est toujours orthogonale aux atomes déjà sélectionnés. Par conséquent, aucun atome ne sera sélectionné deux fois.

### 1.3.3.3 Comparaison des algorithmes de reconstruction

Nous avons vu ici deux algorithmes de reconstruction utilisés dans le contexte de convertisseurs A2I, mais cela ne constitue pas une liste exhaustive de ce genre d'algorithmes. Le Tableau 1.8 présente la complexité de différents algorithmes de reconstruction : le Basis Pursuit De-Noising (BPDN), l'Orthogonal Matching Pursuit (OMP), le Regularized Orthogonal Matching Pursuit (ROMP), le Compressive Sampling Matching Pursuit (CoSaMP), le Chaining Pursuit (CP) ou encore le Subspace Pursuit (SP).

Des algorithmes comme le ROMP, le CoSaMP ou le SP sont des alternatives de l'OMP permettant de corriger son principal défaut. En effet, dans l'OMP, un nouvel atome est sélectionné à chaque itération, l'OMP réalise au maximum  $M$  itérations pour un vecteur de mesure  $\mathbf{y}$  ayant  $M$  éléments. Il est donc sensible au choix de l'atome qui est fait à chaque étape. Par exemple, la différence entre les algorithmes OMP et SP réside dans la manière dont ils génèrent l'ensemble des atomes sélectionnés. Dans le cas de l'OMP, après qu'un atome soit sélectionné,



TABLE 1.8 – Tableau montrant la complexité et le nombre de mesures minimal pour différents algorithmes de reconstruction [3, 73]

Algorithme	Complexité	Nombre minimal de mesures	Références
BPDN	$O(N^3)$	$O(K \log N)$	[3, 68]
OMP	$O(KMN)$	$O(K \log N)$	[3, 74]
ROMP	$O(KMN)$	$O(K \log^2 N)$	[3, 74]
CoSaMP	$O(MN)$	$O(K \log N)$	[3, 75]
CP	$O(K \log^2 N \log^2 K)$	$O(K \log^2 N)$	[76]
SP	$O(KMN)$	$O(K \log(N/K))$	[3, 77]

il y reste pendant tout le processus, alors que dans le SP chaque atome peut être ajouté ou supprimé de l'ensemble à n'importe quelle itération de l'algorithme. Le CoSaMP va lui pouvoir sélectionner plus d'atomes à chaque étape par rapport au SP. Le ROMP peut ajouter un ou plusieurs atomes à chaque itération, mais ne peut pas en supprimer. Les algorithmes gloutons ont l'avantage d'être plus rapides et plus faciles à implémenter que les algorithmes d'optimisation convexe. En revanche, ils ne garantissent pas d'obtenir la meilleure solution au problème.

Le Tableau 1.8 permet de constater l'importante complexité de ce genre d'algorithme, ce qui représente un coût important, en temps et donc en énergie. De plus le choix d'algorithmes plus rapides, comme les algorithmes gloutons, peut se faire au détriment de la précision de reconstruction. Ils sont un des principaux inconvénients des convertisseurs A2I. Pour certaines applications, la reconstruction peut être une étape superflue, car après reconstruction du signal, on cherche à extraire des informations du signal dans le but de détecter certains événements avec des algorithmes d'apprentissage automatique.

### 1.3.4 Conclusion sur les convertisseurs A2I

L'objectif des convertisseurs A2I est de minimiser le nombre de mesures tout en permettant une reconstruction fidèle d'un grand nombre de types de signaux. Cela permet de réduire les contraintes de bande passante des capteurs. Ils doivent également être ajustable pour pouvoir être utilisés dans différentes applications.

Le Tableau 1.9 résume les avantages et les inconvénients des différentes architectures de convertisseurs A2I présentées. La plupart des convertisseurs A2I présentés reposent sur un mélange ou un échantillonnage aléatoire. De telles architectures nécessitent soit d'importantes capacités de mémoires pour stocker les séquences aléatoires, soit de moyens efficaces pour les générer. D'un point de vue matériel, des architectures comme le RD, le RMPI ou le MWC comportent ces circuits qui doivent pouvoir fonctionner à des fréquences allant jusqu'au taux de Nyquist, même si le taux d'échantillonnage moyen est largement inférieur. Dans le cas du NUS, les circuits échantillonneurs bloqueurs et les CAN doivent pouvoir fonctionner au taux de Nyquist, pour les architectures RD et RMPI, la génération de la séquence aléatoire doit également être faite au taux de Nyquist.

TABLE 1.9 – Avantages et inconvénients des différentes architectures de convertisseur A2I

Architecture	Avantage(s)	Inconvénient(s)
NUS	<ul style="list-style-type: none"> <li>• Architecture simple</li> </ul>	<ul style="list-style-type: none"> <li>• Fonctionne au taux de Nyquist</li> <li>• Seulement adaptée aux signaux parcimonieux en fréquence</li> </ul>
RD - RMPI	<ul style="list-style-type: none"> <li>• Fréquence d'échantillonnage réduite</li> </ul>	<ul style="list-style-type: none"> <li>• Nécessite un générateur de séquence aléatoire fonctionnant au taux de Nyquist</li> </ul>
MWC	<ul style="list-style-type: none"> <li>• Signaux multibandes</li> <li>• Fréquence d'échantillonnage réduite</li> </ul>	<ul style="list-style-type: none"> <li>• Générer et stocker les séquences pseudo-aléatoires</li> </ul>
NUWS	<ul style="list-style-type: none"> <li>• Échantillonnage largement inférieur au taux de Nyquist</li> <li>• Flexibilité</li> </ul>	<ul style="list-style-type: none"> <li>• Sélection des ondelettes</li> <li>• Génération des ondelettes</li> </ul>

Le concept d'échantillonnage compressif a été défini dans les années 2000 par E. CANDES et depuis de nombreuses architectures ont été proposées pour des applications allant de l'acquisition de signaux radio-fréquence (RF) [57, 59, 60, 61, 62], à l'acquisition de signaux astrophysiques [78], en passant par l'acquisition de signaux biologiques comme les signaux ECG [4] ou les signaux EEG [79]. La plupart des convertisseurs A2I manquent de polyvalence ou d'adaptabilité, la plupart des paramètres du système étant fixés au moment de la conception. L'échantillonnage non uniforme en ondelettes représente une bonne solution d'extraction de paramètres, car il permet d'obtenir des informations à la fois dans les domaines temporel et fréquentiel sur le signal. De plus, les ondelettes offrent de nombreux degrés de liberté : la famille d'ondelette, les instants d'échantillonnage, la fréquence et le décalage temporel. Cela permet alors d'avoir plus de flexibilité et ainsi pouvoir s'adapter à des types de signaux différents.

### 1.3.5 Discussion sur les convertisseurs A2I

L'architecture proposée dans [4] et illustrée en Figure 1.24 est une architecture de convertisseur A2I de type RMPI, destinée à l'acquisition de signaux biologiques et en particulier les signaux ECG. Elle est composée de plusieurs branches ( $M$ ) en parallèle, chacune faisant l'acquisition d'un échantillon. Une branche est composée d'un mélangeur dont l'une des entrées est codée sur 6 bit (Multiplying Digital-to-Analog Converter (MDAC)); d'un générateur de signaux aléatoire, permettant de créer des matrices de mesures de Bernoulli codées sur 1 bit et des matrices aléatoires uniformes codées sur 6 bit ; d'un intégrateur et d'un CAN SAR ayant une précision de 10 bits. Pour générer les matrices de mesures on utilise un système appelé : Hybrid Linear Feedback Shift Register (HLFSR). Ce circuit est réalisé dans une technologie CMOS 130 nm. Pour une fréquence de fonctionnement  $F_s$  de 2 kHz, la consommation d'une branche est évaluée à 28 nW. La taille de la fenêtre temporelle d'analyse est configurable entre  $N = 128$  et 1024 cycles d'échantillonnage de Nyquist et le nombre de branches en parallèle

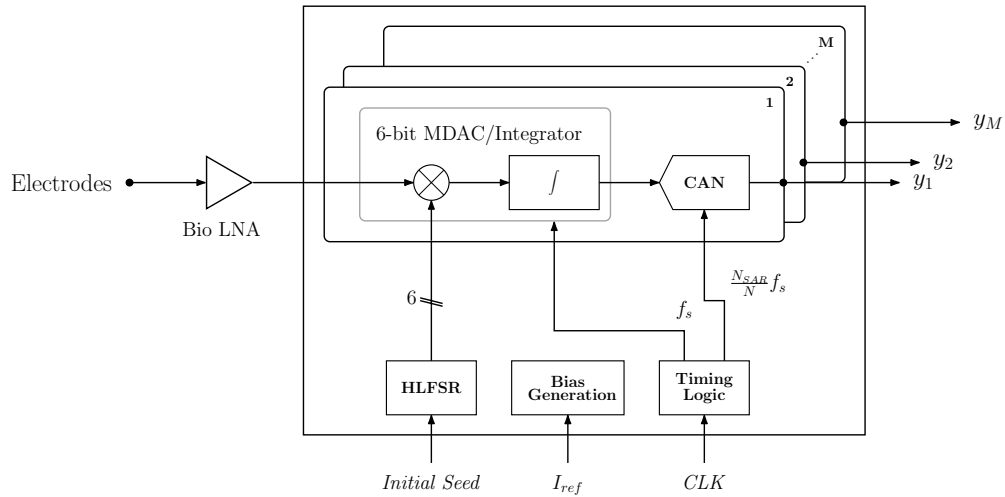


FIGURE 1.24 – Architecture A2I implémentée dans [4]

peut varier entre 1 et 64. Dans les tests réalisés dans [4], le taux de compression maximum obtenu est de 20 pour un signal sinusoïdal 2-parcimoneux dans la base de Fourier et entre 6 et 16 pour un signal ECG, en fonction de la base utilisée.

Nous pouvons comparer la consommation de cette architecture A2I avec une approche classique de capteur sans fil. La chaîne d'acquisition considérée pour l'approche classique est celle de [80]. Cette chaîne est composée d'un amplificateur faible-bruit (LNA), d'un amplificateur à gain variable (PGA), d'un filtre passe-bas de type  $G_m$ -C et d'un CAN SAR ayant une précision de 10 bits. Cette chaîne d'acquisition consomme 3.02 nJ par échantillon, pour une fréquence d'échantillonnage  $F_s$  de 2 kHz. La seconde approche étudiée est basée sur le conver-

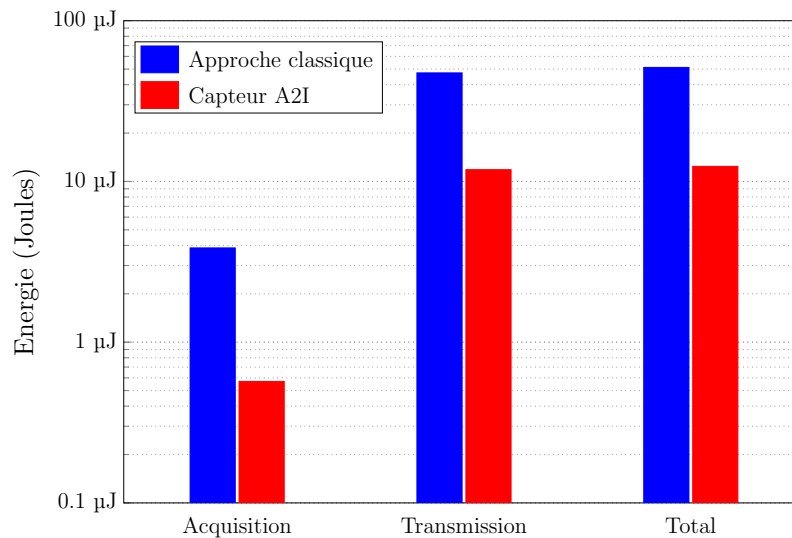


FIGURE 1.25 – Comparaison de la consommation d'énergie entre un capteur utilisant une approche classique et un capteur utilisant le CS

tisseur A2I proposée dans [4]. Celui-ci est configuré pour fonctionner avec  $M = 32$  branches et une fenêtre temporelle d'analyse équivalente à  $N = 128$  cycles d'échantillonnage de Nyquist. Ce qui correspond à un taux de compression de 4. La comparaison de consommation entre ces deux applications est réalisée sur 10 périodes d'analyse du convertisseur A2I (soit 1280 échantillons au taux de Nyquist). Le système de transmission sans-fil considéré est tiré de [42]. Il utilise le standard BLE et consomme 3.7 nJ par bit transmis (soit 37 nJ par échantillon).

Dans le cas présent, l'utilisation d'un convertisseur A2I permettrait de diviser par quatre la consommation engendrée par la transmission des données, comme le montre la Figure 1.25. Cette dernière permet également d'illustrer les propos de la Section 1.2.5, en effet, on constate que la quantité d'énergie dépensée pour la transmission est largement supérieure à celle dépensée pour l'acquisition. Ainsi, le gain total d'énergie est, à presque 97%, dû à la réduction de la quantité d'énergie dépensée pour la transmission. Les 3% restants étant dû à la réduction de la consommation du circuit d'acquisition, alors que celle-ci est divisée par 6.7 dans le circuit A2I. Cette comparaison permet ainsi de donner une idée du gain d'énergie apporté par l'utilisation d'un convertisseur A2I dans le cas du traitement de signaux ECG. Cela dépend en effet du niveau de parcimonie, dans une certaine base, du signal étudié et donc du taux de compression, mais également des performances du circuit réalisé.

## 1.4 Algorithmes de classification

Le rôle des capteurs, notamment dans le domaine de la santé, est de relever l'évolution de différents signaux afin d'en extraire des informations permettant de détecter des anomalies ou des zones d'intérêt dans le signal. L'apprentissage automatique, également appelé apprentissage machine ou apprentissage artificiel (ou en anglais Machine Learning, ML), est une technique qui permet à un système d'apprendre à partir d'un ensemble de données et non à l'aide d'une programmation explicite. Ces algorithmes d'apprentissage automatique permettent d'obtenir des modèles permettant de résoudre certaines tâches comme reconnaître la présence d'un chat dans une photographie. Ce domaine de l'intelligence artificielle est de plus en plus utilisé comme outils d'aide au diagnostic, car elle permet, dans certains cas, de détecter des anomalies difficiles à voir à l'œil nu. Dans [81], une étude est menée pour recenser les progrès récents en matière de machine learning appliqué à la détection de différents types de cancer. Dans [82], les performances d'un classifieur de type Support Vector Machine (SVM) sont évaluées pour détecter des tumeurs cérébrales. La précision du classifieur obtenu est de 96%. Dans le contexte des capteurs connectés et intelligents, l'utilisation de l'apprentissage automatique est également de plus en plus fréquente et les exemples d'application se multiplient comme la détection d'activité physique grâce à des accéléromètres, des gyroscopes ou des signaux EMG [83], la détection de chute [84] ou la détection d'épisodes épileptiques [85].

### 1.4.1 Principe de la classification

La classification consiste à regrouper dans un même ensemble, que l'on appelle classe, label ou étiquette, des observations ayant des caractéristiques communes. Dans le domaine de l'apprentissage automatique, il existe deux types de méthodes pour réaliser une opération de classification : la classification supervisée et la classification non supervisée. L'apprentissage supervisé, dont le fonctionnement est présenté dans la Figure 1.26 consiste à apprendre un modèle de prédiction à partir d'exemples dont la classe est connue, au contraire de l'apprentissage non supervisé qui tente d'extraire des classes présentant des caractéristiques communes à partir de données non étiquetées. Puisque les données ne sont pas étiquetées, il est alors impossible d'évaluer de façon certaine les performances de ce type de classifieur. Nous nous concentrons ici sur les méthodes dites supervisées. Comme le montre la Figure 1.26, la construction supervisée d'un modèle se fait en deux étapes : l'algorithme responsable de l'apprentissage utilise un ensemble de données, appelé ensemble d'entraînement, pour configurer le modèle de classification. L'algorithme tente alors d'ajuster les paramètres du modèle, afin que la décision prise corresponde à la classe donnée en entrée.

Afin d'éviter les biais, les données sont généralement séparées en trois sous-ensembles : un premier ensemble, appelé ensemble d'apprentissage, est utilisé pour ajuster les paramètres du modèle. Les algorithmes d'apprentissage étant souvent itératifs, on utilise un ensemble de données appelé ensemble de validation qui permet de vérifier la pertinence du modèle avec des échantillons qu'il ne connaît pas, cette évaluation sert de critère d'arrêt quand les performances sont jugées suffisamment bonnes. Les ensembles d'apprentissage et de validation sont généralement regroupés pour former l'ensemble d'entraînement. Enfin, l'ensemble de test permet de vérifier si le classifieur généralise suffisamment les données d'entraînement pour cela il faut que les ensembles d'apprentissage et de test soient indépendants.

Les performances de classification sont évaluées avec différentes métriques utilisant les notions de vrai positif (VP), faux positif (FP), faux négatif (FN) et vrai négatif (VN). La matrice de confusion de la Figure 1.27 illustre, pour le cas d'une classification binaire, ces quatre notions.

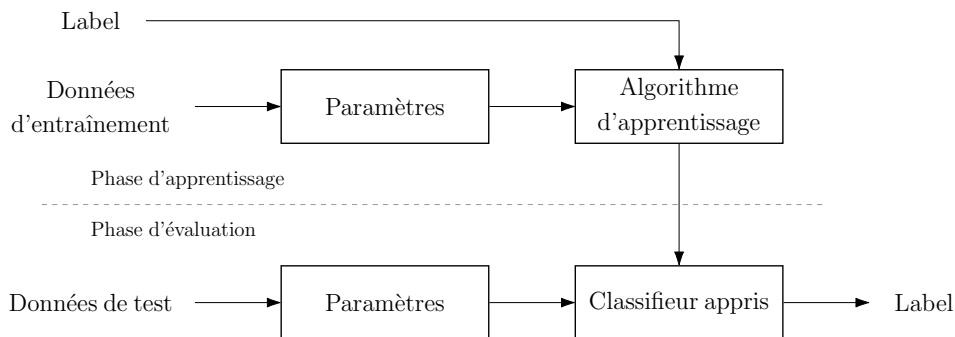


FIGURE 1.26 – Schéma de fonctionnement de la classification supervisée

Voici différentes métriques utilisées pour évaluer les performances de classification d'un modèle appris :

- Précision : elle correspond à la proportion d'observations qui sont correctement classées, par rapport au nombre total d'observations.

$$Précision = \frac{VP + VN}{VP + VN + FP + FN} \quad (1.45)$$

- Sensibilité (Taux de vrai positif) : elle correspond à la proportion d'observations positives qui sont correctement considérés comme positives, par rapport au nombre total d'observations positives.

$$Sensibilité = Se = \frac{VP}{FN + VP} \quad (1.46)$$

- Spécificité (Taux de vrai négatif) : elle correspond à la proportion d'observations négatives qui sont correctement considérés comme négatives, par rapport au nombre total d'observations négatives.

$$Spécificité = \frac{VN}{VN + FP} \quad (1.47)$$

- Taux de faux positifs : il correspond à la proportion d'observations négatives qui sont considérés à tort comme positives, par rapport au nombre total d'observations négatives.

$$Taux\ de\ faux\ positifs = \frac{FP}{VN + FP} \quad (1.48)$$

- Valeur prédictive positive : il correspond à la proportion d'observations positives qui sont correctement classées, par rapport au nombre total d'observations considérées comme positives.

$$Valeur\ prédictive\ positive = +P = \frac{VP}{VP + FP} \quad (1.49)$$

La sensibilité permet d'évaluer la capacité du modèle à correctement prédire la présence d'une anomalie ou d'un évènement lorsque qu'ils surviennent. De manière réciproque, la spécificité permet d'évaluer la capacité à prédire correctement l'absence d'anomalie ou d'évènement lorsque tout est normal. Ces deux métriques ne peuvent pas être utilisées seules et doivent

		Classe estimée	
		Positif	Négatif
Classe réelle	Positif	Vrai Positif (VP)	Faux Négatif (FN)
	Négatif	Faux positif (FP)	Vrai Négatif (VN)

FIGURE 1.27 – Matrice de confusion binaire

être mises en perspectives l'une avec l'autre. En effet, la connaissance de la sensibilité, ou de la spécificité, seule ne permet pas de juger de la qualité d'un modèle de classification : un modèle prédisant toujours la présence d'un événement aura une sensibilité de 100%, mais une spécificité nulle et inversement si le modèle prédit toujours l'absence de cet événement alors la sensibilité sera nulle, alors que la spécificité sera de 100%.

### 1.4.2 Mise en œuvre d'algorithmes de classification

Il existe plusieurs types de modèles utilisant l'apprentissage supervisé, comme les arbres de décisions, les réseaux de neurones ou machines à vecteurs de support. Certains de ces modèles seront plus ou moins adaptés en fonction des situations : la taille de l'ensemble d'apprentissage, le nombre de paramètres, la complexité du modèle ou contraintes matérielles.

#### 1.4.2.1 Arbre de décision et forêt aléatoire

L'arbre de décision [86, 87] est l'une des méthodes de classification les plus utilisés, mais aussi l'une des plus intuitives. Cette méthode utilise des paramètres à valeurs discrètes et peuvent être représenté sous la forme d'une suite de choix en forme d'arbre. Un arbre de décision classe donc les exemples selon les choix effectués depuis la racine jusqu'à atteindre une feuille. Dans un arbre de décision, un nœud est un test basé sur la valeur d'un paramètre et chaque branche partant de ce nœud correspond à une valeur de ce paramètre. Une branche peut soit mener à un autre nœud soit à une feuille qui correspond à une décision sur la classe de l'exemple testé. Le premier nœud d'un arbre de décision est appelé la racine.

La Figure 1.28 est un exemple simple d'arbre de décision. Il permet, en fonction des conditions météorologiques (prévision, humidité, vent et température), de savoir s'il est possible de jouer au tennis. Par exemple, l'instance  $\langle \text{Ensoleillement}=\text{Ensoleillé}, \text{Température}=\text{Chaud}, \text{Humidité}=\text{Élevée}, \text{Vent}=\text{Fort} \rangle$  serait attribué à la classe  $\langle \text{Jouer au Tennis}=\text{Non} \rangle$ . Pour construire un arbre de décision, il existe plusieurs algorithmes dont ID3 et C4.5 [87]. Ces algorithmes sont des algorithmes supervisés, dont l'objectif est de déterminer, à chaque nœud, quel est

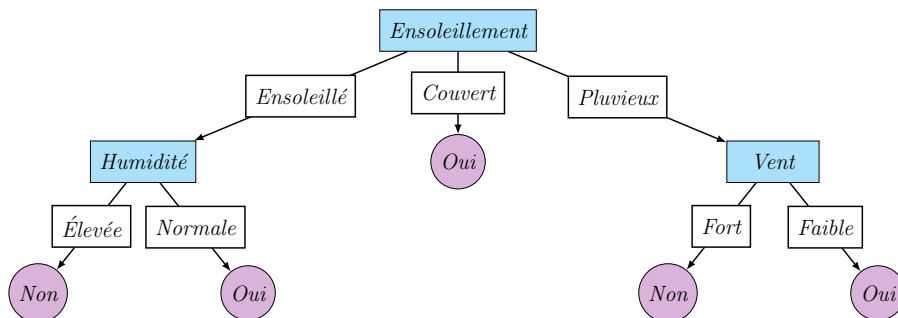


FIGURE 1.28 – Exemple d'arbre de décision permettant de savoir s'il est possible de jouer au tennis en fonction des conditions météorologiques

le meilleur paramètre à tester. Pour cela, on peut utiliser des critères comme le Gain d'information (voir 2.2.1.1) ou le ratio de gain d'information. Des paramètres à valeur continue peuvent être utilisés dans un arbre de décision en les partitionnant en intervalles : le paramètre température (en  $^{\circ}C$ ) est une variable continue que l'on peut diviser en deux intervalles  $T_1$  ( $T < 20^{\circ}C$ ) et  $T_2$  ( $T \geq 20^{\circ}C$ ).

Le phénomène de sur-apprentissage est un problème qui peut apparaître lorsque l'on utilise des méthodes de classification supervisées. Le classifieur appris s'appuie avec excès sur les données d'entraînement et on obtient alors un classifieur qui généralise mal les données. Il en résulte alors une chute des performances de classification. Pour traiter ce problème, on peut utiliser des algorithmes appelés Forêts d'arbres décisionnels ou forêts aléatoires (Random Forest, RF) [88, 89]. Les forêts aléatoires sont un ensemble d'arbres de décision entraînés avec des sous-ensembles des données d'entraînement. Pour constituer ces sous-ensembles d'apprentissage, on utilise un double processus de sélection : sur les données et sur les paramètres. On considère un ensemble de données d'apprentissage de  $N$  observations, décrites par  $P$  paramètres. La constitution des sous-ensembles se fait d'abord via un tirage aléatoire avec remise de  $m < N$  observations, puis par une sélection d'un nombre  $q$  de paramètres pour décrire le sous-ensemble. En général, on considère que  $q < \sqrt{P}$ . Chaque arbre est alors entraîné avec un sous-ensemble des données et la décision finale lors de la classification est déterminée en effectuant un vote majoritaire des décisions des arbres de la forêt.

### 1.4.2.2 Réseaux de neurones artificiels

Un réseau de neurones (Neural Network, NN) artificiel [87] est une technique de classification inspirée du fonctionnement des neurones du cerveau. Ils sont composés de neurones connectés entre eux selon différentes architectures. Ils sont particulièrement bien adaptés aux problèmes utilisant des données bruitées issues de capteurs comme des images ou du son. Les réseaux de neurones sont appropriés dans le cas de problème avec beaucoup de paramètres, ils permettent d'obtenir une sortie discrète (classification), réelle (régression) ou bien un vecteur de valeurs réelles ou discrètes.

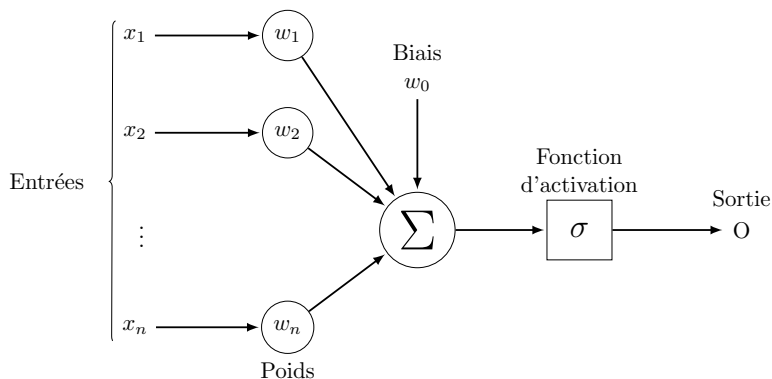


FIGURE 1.29 – Schéma d'un neurone artificiel



La Figure 1.29 représente un neurone artificiel. Il possède  $n$  entrée(s)  $x_i$  ( $i \in [1, n]$ ) à valeur réelle, une pour chaque paramètre de l'ensemble de données. Chacune des entrées est associée à un poids  $w_i$  déterminant leur contribution sur la sortie. On ajoute également un biais  $w_0$ , correspondant au poids d'une entrée  $x_0$  constante ( $x_0 = 1$ ). On calcule alors la somme des valeurs d'entrée pondérées. La fonction d'activation permet de déterminer la valeur de sortie du neurone à partir de la valeur de cette somme. La fonction d'activation la plus simple est la fonction *Sgn*, tel que si la somme est positive alors la sortie est 1, sinon la sortie est  $-1$ .

$$\sigma(s) = \begin{cases} 1 & \text{si } s > 0 \\ -1 & \text{sinon} \end{cases} \quad \text{avec } s = \sum_{i=0}^n w_i \cdot x_i \quad (1.50)$$

Le biais  $w_0$  peut alors être vu comme un seuil tel que :

$$\sigma(s) = \begin{cases} 1 & \text{si } s = \sum_{i=1}^n w_i \cdot x_i > -w_0 \\ -1 & \text{sinon} \end{cases} \quad (1.51)$$

D'autres fonctions peuvent être utilisées comme fonction d'activation des neurones, à la place de la fonction *Sgn*, comme les fonctions sigmoïde ou tangente hyperbolique :

$$\text{Sigmoïde} = \sigma(s) = \frac{1}{1 + e^{-s}} \quad \text{avec } s = \sum_{i=0}^n w_i \cdot x_i \quad (1.52)$$

$$\text{Tangente hyperbolique} = \tanh(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}} \quad (1.53)$$

Le perceptron est le type de réseau de neurones le plus simple, qui a été proposé par Frank Rosenblatt en 1957 [90]. Il s'agit d'un unique neurone auquel on ajoute une règle d'apprentissage permettant de calculer et adapter les poids du neurone. Les poids  $w_i$ , associés aux entrées

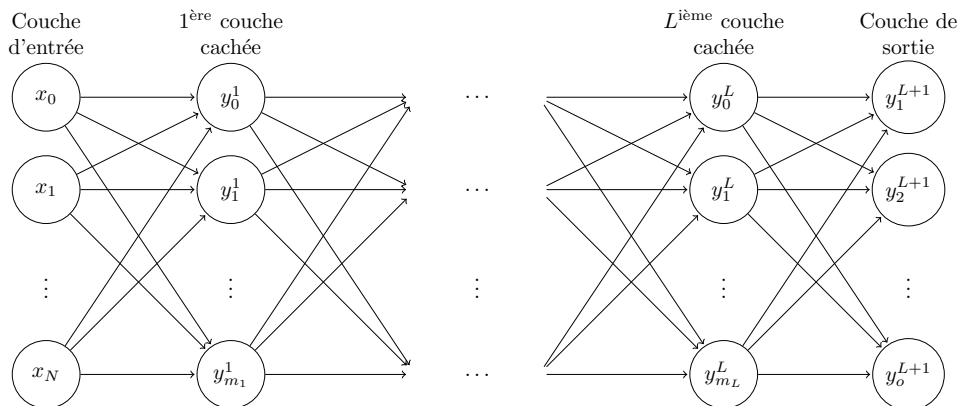


FIGURE 1.30 – Réseau de neurones multicouches

$x_i$ , sont mis à jour à chaque étape selon la règle suivante :

$$w'_i = w_i + \eta(t - o)x_i, \quad (1.54)$$

où  $w'_i$  est la version mise à jour de  $w_i$ ,  $t$  est la valeur cible (target),  $o$  est la sortie calculée par le neurone et  $\eta$  est une constante positive appelée le taux d'apprentissage (Learning rate). Le rôle du taux d'apprentissage  $\eta$  est de modérer le degré de modification des poids à chaque étape.

Contrairement à un perceptron qui permet seulement d'avoir des classifieurs linéaires, les réseaux multicouches permettent également d'obtenir des classifieurs non-linéaires. Un réseau de neurones multicouche, représenté à la Figure 1.30, possède au moins trois couches : une couche d'entrée, une ou plusieurs couches cachées et d'une couche de sortie. La couche d'entrée contient autant d'entrées ( $N$ ) qu'il y a de paramètres dans l'ensemble de données.

### 1.4.2.3 Classifieur Bayésien

Les classifieurs Bayésiens [87] sont des classifieurs basés sur les probabilités, le but est donc de déterminer quelle est l'hypothèse la plus probable, parmi l'ensemble des hypothèses candidates  $H$ , à la vue des exemples d'entraînement. Pour cela, on utilise le théorème de Bayes :

$$P(h|D) = \frac{P(D|h).P(h)}{P(D)} \quad (1.55)$$

avec  $P(h|D)$  la probabilité que l'hypothèse  $h$  soit la bonne connaissant les données  $D$ ,  $P(h)$  la probabilité de l'hypothèse  $h$  avant d'avoir observé les données d'entraînement,  $P(D)$  la probabilité d'observer les données  $D$  et  $P(D|h)$  la probabilité d'observer les données  $D$  si  $h$  est la solution. La meilleure hypothèse est donc celle qui maximise  $P(h|D)$  :

$$\begin{aligned} h_m &= \operatorname{argmax}_{h \in H} P(h|D) \\ h_m &= \operatorname{argmax}_{h \in H} \frac{P(D|h).P(h)}{P(D)} \\ h_m &= \operatorname{argmax}_{h \in H} P(D|h).P(h) \end{aligned} \quad (1.56)$$

Notez que le terme  $P(D)$  a été supprimé, car il s'agit d'une constante indépendante de l'hypothèse  $h$ . Un classifieur Bayésien assigne donc à un nouvel exemple, la classe la plus probable  $c_m$  en fonction de la valeur de ses paramètres.

$$\begin{aligned} c_m &= \operatorname{argmax}_{c_j \in C} P(c_j|a_1, a_2, \dots, a_n) \\ c_m &= \operatorname{argmax}_{c_j \in C} \frac{P(a_1, a_2, \dots, a_n|c_j).P(c_j)}{P(a_1, a_2, \dots, a_n)} \\ c_m &= \operatorname{argmax}_{c_j \in C} P(a_1, a_2, \dots, a_n|c_j).P(c_j) \end{aligned} \quad (1.57)$$

La probabilité  $P(c_j|a_1, a_2, \dots, a_n)$  est la probabilité que l'exemple appartienne à la classe  $c_j$  connaissant la valeur de ses paramètres  $(a_1, a_2, \dots, a_n)$  et  $P(c_j)$  est la probabilité que l'exemple appartienne à la classe  $c_j$ . Il est facile d'estimer les valeurs de  $P(c_j)$  simplement en comptant le nombre d'exemples dans l'ensemble d'entraînement qui appartiennent à la classe  $c_j$ . Il est en revanche extrêmement compliqué de déterminer les valeurs de  $P(a_1, a_2, \dots, a_n|c_j)$ , car cela nécessite beaucoup de calculs et d'avoir un ensemble de données extrêmement grand. Les classifieurs Bayésiens Naïfs sont construits sur le principe que tous les paramètres sont indépendants. Cela signifie que la probabilité  $P(a_1, a_2, \dots, a_n|c_j)$  peut être calculée de la façon suivante :

$$P(a_1, a_2, \dots, a_n|c_j) = \prod_i P(a_i|c_j) \quad (1.58)$$

La probabilité  $P(a_i|c_j)$  est la probabilité que le paramètre  $a_i$  ait une certaine valeur sachant que l'exemple appartient à la classe  $c_j$ . À partir de cette hypothèse, on peut alors réécrire l'équation 1.57 :

$$c_m = \operatorname{argmax}_{c_j \in C} P(c_j) \cdot \prod_i P(a_i|c_j) \quad (1.59)$$

#### 1.4.2.4 K plus proches voisins (KNN)

Le but d'un algorithme des K plus proches voisins [87, 91] est de classer un exemple dans la même classe que les exemples qui lui sont similaires. L'algorithme KNN recherche les  $K$  exemples les plus proches de l'exemple à classer parmi les exemples de l'ensemble d'entraînement. Chaque exemple peut être considéré comme un point dans un espace de dimension  $n$  ( $n$  étant le nombre de paramètres). Pour mesurer la similarité entre deux exemples, on utilise une mesure de distance comme la distance Euclidienne ou la distance de Manhattan. Idéalement, cette fonction minimise la distance entre deux exemples de la même classe et maximise la distance entre des exemples de classes différentes. La classe attribuée à l'exemple testé est la classe majoritaire parmi les  $k$  exemples les plus proches.

#### 1.4.2.5 Support Vector Machine (SVM)

Les Séparateurs à Vastes Marges, également appelés Machines à Vecteurs de Support, (SVM) [92] sont des algorithmes d'apprentissage automatique initialement conçus pour des tâches de classification binaire et qui ont ensuite été adaptés à des problèmes de régression. Dans le cas d'une classification binaire, le but de ces algorithmes est de trouver l'hyperplan qui permet le mieux de séparer les données de classe différente tout en étant le plus éloigné possible de toutes les observations. La marge est alors la distance entre l'hyperplan et les échantillons les plus proches qu'on appelle « vecteurs supports ».

La recherche de cet hyperplan, consiste à résoudre un problème d'optimisation. Dans le cas où les données ne sont pas linéairement séparables, on utilise une fonction noyau (kernel)

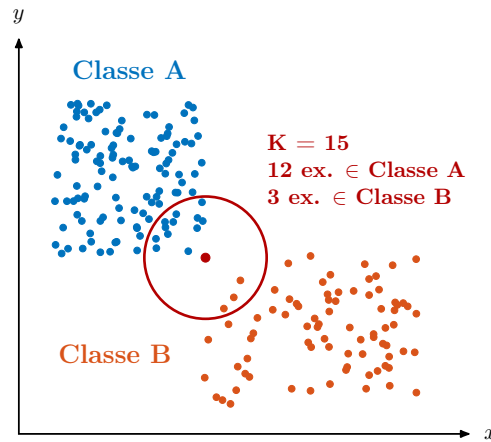


FIGURE 1.31 – Principe de fonctionnement d'un classifieur K plus proches voisins

qui permet de transformer l'espace de représentation de départ en un espace de plus grande dimension dans lequel il existe une séparation linéaire. On considère un ensemble d'exemple décrit par des paramètres  $x = \{x_1, x_2, \dots, x_n\}$  et une classe  $y = \{-1; 1\}$ . Pour déterminer la classe d'un exemple, il est nécessaire de construire une fonction  $f$  qui à un vecteur d'entrée  $x$ , situé dans un espace  $F$  muni d'un produit scalaire, fait correspondre une valeur  $y$ . Dans le cas d'un classifieur linéaire :

$$f(x) = \langle w, x \rangle + b \quad (1.60)$$

où  $x$  est donc le vecteur d'entrée et  $w$  est un vecteur de pondération orthogonal au plan. Le signe de la fonction  $f$  indique alors de quel côté du plan se trouve l'exemple  $x$ , ce qui permet d'en déduire sa classe. Si  $f(x) > 0$  alors  $y = 1$  et si  $f(x) < 0$  alors  $y = -1$ . Si le problème est linéairement séparable, on a alors :

$$y \cdot f(x) > 0, \quad \forall(x, y) \quad (1.61)$$

Afin d'être plus tolérant face aux petites variations dans les données, on cherche un hyperplan avec la plus grande marge possible. On choisit de définir un coefficient multiplicatif afin de normaliser le couple  $(w, b)$ , de façon à ce que les exemples à la marge, ceux les plus proches de l'hyperplan, satisfassent la relation  $f(x) = 1$ . Donc pour tout échantillon, on a :

$$y \cdot f(x) \geq 1, \quad \forall(x, y) \quad (1.62)$$

La distance entre un échantillon  $x$  et l'hyperplan est défini par la relation :

$$d(x) = \frac{|\langle w, x \rangle + b|}{\|w\|} = \frac{|f(x)|}{\|w\|} \quad (1.63)$$

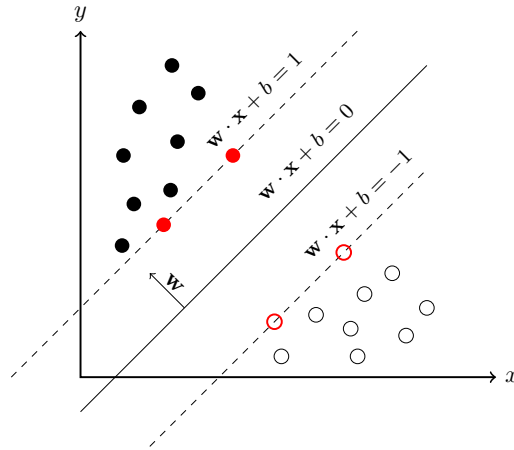


FIGURE 1.32 – Principe de fonctionnement d'un SVM

Du fait de la normalisation de  $(w, b)$ , la marge vaut donc  $1/\|w\|$ . Pour trouver le plan séparateur de marge maximale, on doit donc résoudre le problème d'optimisation suivant :

$$\min \frac{1}{2} \|w\|^2 \quad \text{avec } \forall i, y_i (\langle w, x_i \rangle + b) \geq 1 \quad (1.64)$$

Dans le cas où les données ne sont pas linéairement séparables, on applique au vecteur d'entrée une transformation non-linéaire  $\phi$ . On cherche alors l'hyperplan :

$$f'(x) = \langle w', \phi(x) \rangle + b / \forall i, y_i f'(x_i) > 0 \quad (1.65)$$

Exemple :  $x = (x_1, x_2)$  et  $\phi(x) = (x_1^2, \sqrt{2} \cdot x_1 \cdot x_2, x_2^2)$

Le problème est que l'on souhaite effectuer un produit scalaire dans un espace de dimension élevée, ce qui est coûteux en calculs. Pour cela, on utilise une fonction noyau qui vérifie la relation :

$$K(x_i, x_j) = \phi(x_i)^T \cdot \phi(x_j) \quad (1.66)$$

Il existe plusieurs noyaux dont les plus communs sont le noyau polynomial et le noyau gaussien :

$$\begin{aligned} K_{\text{polynomial}}(x_i, x_j) &= (x_i^T \cdot x_j + 1)^d \\ K_{\text{gaussien}}(x_i, x_j) &= e^{-\frac{\|x_i - x_j\|^2}{2 \cdot \sigma^2}} \end{aligned} \quad (1.67)$$

À noter que, sur le plan algorithmique, ces algorithmes sont plus pénalisés par le nombre d'observations, c'est-à-dire le nombre de vecteurs supports potentiels, que par le nombre de paramètres décrivant les données d'entraînement.

### 1.4.2.6 Régression logistique

La régression logistique est un modèle de classification, basé sur une approche statistique, permettant d'évaluer les liens entre un ensemble de variables qualitatives  $x = \{x_1, x_2, \dots, x_i\}$  et une classe binaire  $y = \{0, 1\}$ . Elle est particulièrement utilisée en médecine [93] ou dans les sciences sociales. On considère un ensemble de  $N$  exemples  $(x, y)$  décrits par un ensemble de paramètres  $x = \{x_1, x_2, \dots, x_i\}$  et une classe binaire  $y = \{0, 1\}$ . On considère également que les variables  $x_j$  sont des variables continues. On note :

- $P(y = 1)$  la probabilité que la classe d'un exemple donné soit  $y = 1$
- $P(x|y = 1)$  la probabilité d'obtenir un vecteur de paramètres  $x$  sachant qu'il appartient à la classe  $y = 1$ .
- $P(y = 1|x)$  la probabilité qu'un exemple appartienne à la classe  $y = 1$  sachant les valeurs des différents paramètres  $x$
- $E(y|x)$  correspond à la valeur attendue de  $y$  sachant la valeur de  $x$ , on l'appelle Espérance conditionnelle

Dans une régression linéaire  $E(y|x) = \beta_0 + \beta_1 \cdot x$  et  $E(y|x) \in \mathbb{R}$ . Dans une régression logistique, où la variable à prédire  $y$  est binaire,  $E(y|x) \in [0, 1]$  et la formule utilisée pour estimer  $E(y|x)$  est :

$$E(y|x) = \frac{e^{\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_i \cdot x_i}}{1 + e^{\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_i \cdot x_i}} \quad (1.68)$$

Cette formule vient du fait que l'on utilise une fonction logistique pour modéliser la loi de probabilité :

$$\ln \left( \frac{E(y|x)}{1 - E(y|x)} \right) = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_i \cdot x_i \quad (1.69)$$

L'objectif est ensuite de déterminer les valeurs de  $\beta_j$  à partir des différents exemples grâce au principe du maximum de vraisemblance. Pour utiliser cette méthode, on doit construire une fonction de vraisemblance. Du fait que  $y$  est binaire (0 ou 1), on peut dire que  $P(y = 1|x) = E(y|x)$  et  $P(y = 0|x) = 1 - E(y|x)$ . La « contribution » d'un exemple  $k = (x, y)$  à la vraisemblance peut être décrit par la grandeur :

$$P(y = 1|x)^y \cdot [1 - P(y = 1|x)]^{1-y} \quad (1.70)$$

La vraisemblance d'un ensemble  $\Omega$  d'exemples s'écrit alors :

$$l(\beta) = \prod_{i=1}^N P(y_i = 1|x_i)^{y_i} \cdot [1 - P(y_i = 1|x_i)]^{1-y_i} \quad (1.71)$$

Les valeurs de  $\beta$  sont alors choisies de façon à maximiser la fonction de vraisemblance  $L(\beta)$ .

### 1.4.3 Conclusion sur les algorithmes de classification

Dans cette section, différents modèles de classification ont été abordés : les arbres de décision, les réseaux de neurones, les classifieurs Bayésien ou encore les machines à vecteurs de supports. Le Tableau 1.10 résume les avantages et les inconvénients des différents modèles. Il permet de conclure qu'il n'existe globalement pas de modèle idéal : le choix d'un modèle de classification s'effectue en fonction de l'application souhaitée, des données à disposition, mais également en fonction des contraintes matérielles pour le déploiement du modèle. En effet, il est difficilement imaginable de déployer un modèle de KNN dans un système avec des ressources limitées en termes de mémoire.

## 1.5 Conclusion

Ce chapitre présente les différentes parties d'un capteur sans-fil : le conditionnement du signal, la numérisation, le traitement et la transmission. Cela permet ainsi de se rendre compte du coût énergétique que représente la transmission de données pour un capteur sans-fil et désigne donc cette étape comme principal vecteur d'économie d'énergie. Une réduction de la quantité de données à transmettre, et donc une utilisation réduite du système RF, permettrait

TABLE 1.10 – Avantages et inconvénients des différents modèles de classification

Modèle	Avantage(s)	Inconvénient(s)
Forêts Aléatoires	<ul style="list-style-type: none"> <li>• Traitement de tâches non-linéaires</li> <li>• Robuste au sur-apprentissage</li> </ul>	<ul style="list-style-type: none"> <li>• Taille excessive suivant le nombre et la profondeur des arbres</li> </ul>
Réseaux de neurones	<ul style="list-style-type: none"> <li>• Traitement de tâches non-linéaires</li> </ul>	<ul style="list-style-type: none"> <li>• Choix de la topologie (nombre de couches et de neurones)</li> <li>• Influence des valeurs initiales des poids sur la vitesse de convergence</li> </ul>
K plus proches voisins	<ul style="list-style-type: none"> <li>• Traitement de tâches non-linéaires</li> </ul>	<ul style="list-style-type: none"> <li>• Non-prise en compte des interactions entre les paramètres</li> <li>• Stockage nécessaire des données "d'apprentissage"</li> </ul>
Classifieur Bayésien Naïf	<ul style="list-style-type: none"> <li>• Étape de classification simple</li> <li>• Calculs rapides sur des ensembles de données volumineux</li> </ul>	<ul style="list-style-type: none"> <li>• Considère les paramètres comme indépendants</li> <li>• Ne traite pas les tâches non-linéaires</li> </ul>
Machines à Vecteurs de Support	<ul style="list-style-type: none"> <li>• Recommandé pour les tâches non-linéaires</li> </ul>	<ul style="list-style-type: none"> <li>• Calculs important sur des ensembles de données volumineux</li> </ul>
Régression logistique	<ul style="list-style-type: none"> <li>• Décrit l'influence relative des paramètres sur la classe</li> </ul>	<ul style="list-style-type: none"> <li>• Non-adapté pour les ensembles avec beaucoup de paramètres</li> <li>• Ne traite pas les tâches non-linéaires</li> </ul>

de réduire le coût énergétique de ce dernier. Le CS [2] est une technique bien connue qui permet de réduire le nombre d'échantillons acquis, par rapport aux approches classiques fonctionnant au taux de Nyquist, et donc de réduire la quantité de données à transmettre. Le but du CS est de faire l'acquisition de données compressées et ensuite de reconstruire, à partir de ces données, le signal d'origine grâce à un algorithme de reconstruction. Le problème est que ces algorithmes sont complexes [3] et que le taux de compression est limité [4]. Pour certaines applications, reconstruire le signal d'origine n'est pas nécessaire. Dans le cas d'une application réalisant une classification pour détecter certains types d'événements, seules certaines informations sont nécessaires. Les convertisseurs analogique-paramètre (Analog-to-Feature, A2F) sont des systèmes d'acquisition pensée pour l'IoT, dans le but d'augmenter l'autonomie de ces objets. Comme pour le CS, le but des convertisseurs A2F est de diminuer le nombre d'échantillons acquis en extrayant seulement les informations utiles du signal. Mais contrairement au CS, le but ici n'est pas de reconstruire le signal, mais d'utiliser les informations extraites comme données d'entrée d'algorithmes de Machine Learning (ML), par exemple, pour réaliser de la détection d'activité vocale ou bien pour détecter des anomalies dans un électrocardiogramme (ECG) ou un électroencéphalogramme (EEG). L'objectif principal de la conversion analogique-paramètre est donc de définir, pour une application donnée, ce qu'est une information utile et comment l'extraire. Dans le Chapitre suivant, nous proposons d'utiliser l'échantillonnage non-uniforme en ondelettes (NUWS) pour réaliser l'extraction de paramètres. Cette technique offre différents degrés de liberté et permet d'obtenir des informations à la fois temporelles et fréquentielles sur le signal étudié. Elle impose donc d'utiliser des algorithmes de sélection de paramètres afin de déterminer, pour une application donnée, quelles doivent être les informations à extraire en vue de la classification.



## Chapitre 2

# Extraction de paramètres par échantillonnage non-uniforme en ondelettes pour la détection d'anomalies d'un signal ECG

### Introduction

Le Chapitre 1 a mis en évidence le coût énergétique du système RF dans la consommation des capteurs sans-fil, ainsi que le gain que peut apporter une réduction de son utilisation grâce à une diminution de la quantité de données transmises. Les convertisseurs A2I utilisant le CS sont une première solution pour diminuer le nombre d'échantillons, mais le taux de compression est limité. Les convertisseurs A2F sont une nouvelle solution qui, comme pour les convertisseurs A2I, permet de diminuer la quantité de données en extrayant, de manière analogique, des informations utiles du signal pour la réalisation d'une classification ou d'une régression. Les solutions actuelles de convertisseurs A2F sont conçues pour des applications spécifiques telles que la détection d'activité vocale [7]. Le but de cette thèse est de proposer une architecture et une méthode afin de concevoir un convertisseur A2F générique, utilisable pour différents types de signaux basse fréquence (ECG, EEG, voix, ...). Pour cela, nous proposons d'utiliser l'échantillonnage non-uniforme en ondelettes (NUWS), qui est une méthode issue du CS, pour réaliser l'extraction de paramètres. Ensuite, nous proposons d'utiliser des algorithmes de sélection de paramètres comme le score de Fisher (FS), le gain d'information (IG) ou la recherche séquentielle (SFS), afin de déterminer, pour une application donnée, quelles doivent être les informations à extraire en vue de la classification. Lorsque nous aurons déterminé les caractéristiques des ondelettes que l'on souhaite utiliser, nous pourrons proposer une architecture matérielle et ainsi proposer un modèle de consommation énergétique qui permettra d'évaluer le gain de consommation par rapport à une approche classique de capteur sans fil.

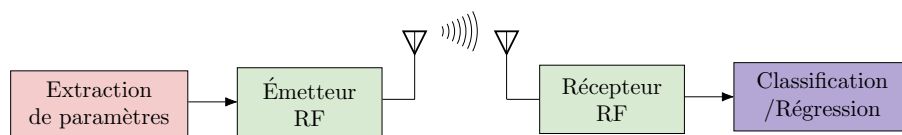
## 2.1 Convertisseur Analogique-Paramètre (A2F) basé sur le NUWS

### 2.1.1 Principe du convertisseur analogique-paramètre

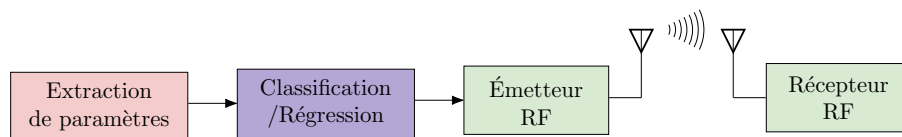
L'objectif du CS est de minimiser le nombre de mesures tout en permettant une reconstruction fidèle d'un grand nombre de types de signaux et de minimiser les besoins en bande passante. Dans le cas de certaines applications, comme la détection d'activité vocale, la reconstruction du signal d'origine n'est pas nécessaire. Les convertisseurs analogique-paramètre (Analog-to-Feature, A2F) ont pour objectif d'extraire les informations utiles du signal de manière analogique, puis de les convertir en données numériques et enfin de les utiliser comme données d'entrée d'algorithmes de Machine Learning (ML) permettant de réaliser le travail de classification ou de détection souhaité. De manière simple, les paramètres pourraient être extraits de manière numérique, suite à une acquisition au taux de Nyquist, mais cela nécessite d'acquérir, de façon superflue, la totalité de l'information du signal pour ne filtrer que les informations nécessaires. Ce qui n'est pas efficace d'un point de vue énergétique. C'est pour cette raison que le but de la conversion analogique-paramètre est de combiner les étapes d'extraction et d'acquisition.

La Figure 2.1 présente deux exemples de convertisseurs A2F où l'étape de classification peut être réalisée au niveau du capteur ou de l'agrégateur. En réalisant cette étape au niveau du capteur, seul le résultat de classification est transmis, ce qui permet de diminuer encore plus la consommation due à la transmission, mais cela nécessite d'implémenter un classifieur dans le capteur ce qui induira une augmentation de consommation au niveau du capteur.

L'architecture d'un convertisseur A2F, tel que proposée dans [94], est illustrée à la Figure 2.2. Cette architecture est composée de plusieurs extracteurs de paramètres reconfigurables et utilisés en parallèle. La sortie de chaque extracteur est connectée à un multiplexeur qui redirige les différents signaux vers un CAN qui numérise les paramètres extraits. Ces paramètres peuvent ensuite être utilisés par un classifieur et un détecteur de contexte. Ce dernier peut être utilisé pour activer, désactiver ou reconfigurer les extracteurs de paramètres



(a) Convertisseur A2F transmettant les paramètres



(b) Convertisseur A2F avec traitement local

FIGURE 2.1 – Exemples de convertisseurs A2F

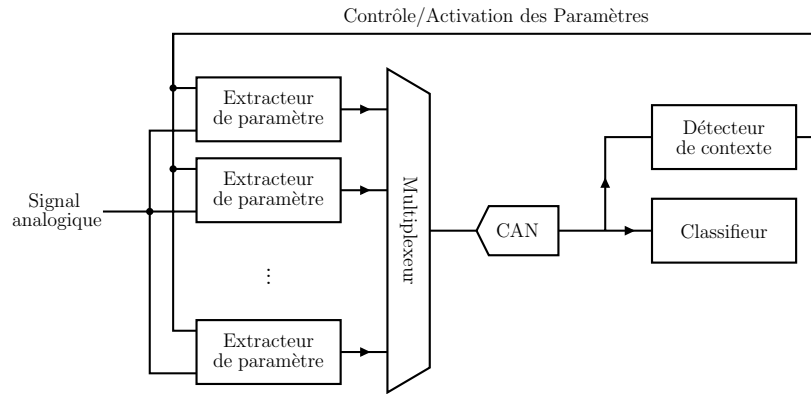


FIGURE 2.2 – Architecture de convertisseur A2F proposée dans [94]

en fonction du type de signal traité ou de la précision souhaitée. Il existe une variante à cette architecture dans laquelle chaque extracteur possède son propre CAN. Cela permet de s'affranchir du multiplexeur, mais aussi et surtout de diminuer les contraintes en termes de fréquence d'échantillonnage, sur les CAN, au prix d'une surface plus importante. Un des avantages qu'apporte l'utilisation d'algorithmes de classification dans les convertisseurs A2F, est que les contraintes sur la conception des circuits analogiques est réduite. En effet, les non-linéarités pouvant être induites par les circuits analogiques, comme les amplificateurs ou les filtres, n'ont pas forcément d'influence sur le processus de classification. Celles-ci peuvent en effet être apprises en utilisant un modèle approprié, alors qu'un traitement classique suppose que tout est linéaire.

Une implémentation matérielle de convertisseur A2F est proposée dans [6] et [7], et réalisée dans une technologie CMOS 90 nm, pour une consommation maximale de 6  $\mu$ W. Ce convertisseur A2F permet de réaliser une détection d'activité vocale avec une précision maximum de 89% tout en réduisant par 10 la quantité d'énergie par rapport à des systèmes provenant de l'état de l'art. Ce convertisseur A2F permet d'extraire le contenu énergétique du signal audio entrant, sur 16 bandes de fréquences comprises entre 75 Hz et 5 kHz. Il est composé de seize extracteurs en parallèles, chacun permettant de connaître le contenu d'une bande de fréquence. Un arbre de décision, implémenté avec une architecture mixte sur circuit, permet de réaliser la détection d'activité vocale. La détection du contexte permet d'adapter l'extraction des paramètres en fonction du bruit de fond : en effet, l'information nécessaire est la même quel que soit le niveau de bruit, mais ce dernier peut créer des interférences dans certaines bandes de fréquence. Il est ainsi plus difficile de distinguer la parole de la non-parole. L'extraction et l'utilisation par le classifieur d'un ensemble de paramètres plus grand, permet de faciliter la détection dans un environnement bruyant, justifiant ainsi l'augmentation de la consommation énergétique.

L'un des inconvénients des solutions de convertisseurs A2F actuelles est qu'elles sont conçues pour des applications spécifiques [95, 7]. L'objectif de cette thèse est de proposer une méthode et une architecture afin de concevoir un convertisseur A2F générique, qui soit

utilisable pour une large gamme de signaux basse fréquence (voix, ECG, EEG, ...). L'échantillonnage non-uniforme en ondelettes (NUWS) a été choisi comme méthode d'extraction de paramètres, en effet, cette méthode permet d'obtenir des informations à la fois temporelles et fréquentielles sur le signal. De plus, les ondelettes offrent de nombreux degrés de liberté : la famille d'ondelette, les instants d'échantillonnage, la fréquence et le décalage temporel. Cela permet donc d'avoir plus de flexibilité et ainsi pouvoir s'adapter à des types de signaux différents. La possibilité de reconfigurer les extracteurs et les différents degrés de liberté du NUWS permettent de développer une méthode d'extraction adaptative : on peut identifier la présence (ou non) d'anomalies avec une échelle grossière puis réduire cette échelle pour effectuer une analyse plus détaillée et ainsi détecter le type d'anomalie.

Le schéma de la Figure 2.2 représente l'architecture globale d'un convertisseur A2F qui utilise plusieurs extracteurs en parallèle. Pour réaliser l'extraction des paramètres, la technique du NUWS a été proposée, dont l'objectif est l'acquisition de coefficients d'une transformée en ondelette, qui est définie à l'Equation 1.32

Afin d'obtenir les coefficients d'ondelettes de la transformée en ondelettes, l'architecture de la Figure 2.3 est proposée comme extracteur de paramètres. Cette architecture s'appuie ainsi sur la définition de la transformée en ondelette. Le signal entrant est mixé avec une ondelette puis le résultat est intégré sur le support de celle-ci. Une unité de contrôle permet de configurer, pour une ondelette mère donnée, la largeur de l'ondelette ( $l$ ) et donc le début et la fin de la période d'intégration ( $t_1, t_2$ ), la fréquence de l'ondelette ( $f_c$ ), le décalage temporel ( $t_c$ ) et les instants d'échantillonnage. Le résultat de l'intégration est finalement numérisé par le CAN. L'échantillon généré est un paramètre qui sera ensuite utilisé par le classifieur.

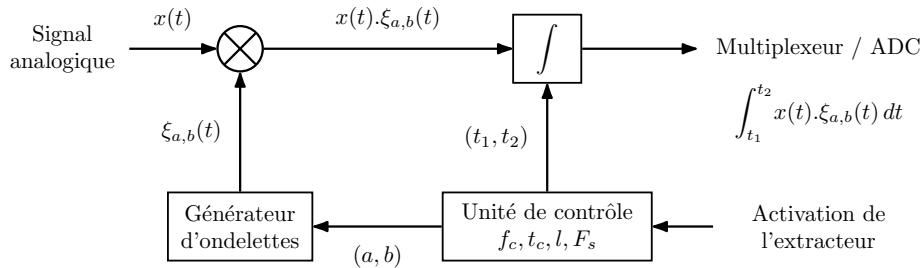


FIGURE 2.3 – Extracteur de paramètres proposé

L'unité de contrôle, présente dans chaque extracteur, permet de rendre ce dernier flexible et ainsi d'être reconfiguré pour extraire plusieurs paramètres. Un extracteur est capable d'acquérir deux paramètres si les supports temporels des ondelettes associées ne se superposent pas. Afin d'obtenir un modèle de consommation de notre convertisseur A2F, une implémentation au niveau matériel de cette architecture doit être proposée. Pour réaliser l'extraction de paramètres notre architecture a donc besoin d'un mixeur analogique, d'un intégrateur, d'un générateur d'ondelette reconfigurable avec une unité de contrôle et d'un CAN.

Pour montrer l'intérêt des convertisseurs A2F dans des applications de détection d'événements, nous allons utiliser l'exemple de la détection d'anomalie dans des signaux ECG. Nous allons donc maintenant nous pencher sur les signaux ECG.

### 2.1.2 Introduction sur les signaux ECG

Un électrocardiogramme (ECG) est une représentation graphique de l'activité électrique du cœur. La Figure 2.4 est un schéma représentant le cœur humain. Celui-ci est composé de 4 cavités : les ventricules gauche et droit, numérotés (6) et (7) sur le schéma, et les oreillettes gauche et droite, numérotés (8) et (9). Chacune étant recouverte de terminaisons nerveuses. Le rôle du cœur est de distribuer le sang chargé en oxygène à l'ensemble des organes du corps humain. Le sang provenant des muscles et des organes et donc pauvre en oxygène arrive au niveau de l'oreillette droite. Celle-ci, lorsqu'elle est remplie, se contracte pour propulser le sang dans le ventricule droit, qui se contracte à son tour pour envoyer le sang vers les poumons pour qu'il se charge en oxygène. Le sang oxygéné retourne au cœur au niveau de l'oreillette gauche qui l'envoie dans le ventricule gauche, qui, par sa contraction, envoie le sang vers les organes via l'aorte.

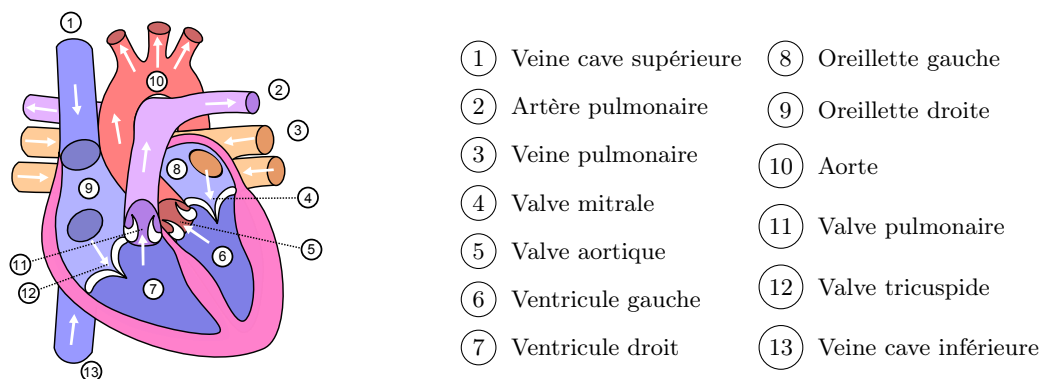


FIGURE 2.4 – Schéma du cœur humain. Source : [96]

C'est donc l'activité électrique, transmise via le système nerveux, responsable de la contraction du cœur, qui est mesurée lors d'un électrocardiogramme. Pour cela, on place plusieurs électrodes à la surface de la peau. Le signal représenté à la Figure 2.5 est un exemple de signal obtenu lors d'un ECG. Un cycle cardiaque est composé de plusieurs ondes regroupées dans ce que l'on appelle le complexe PQRST, chacune de ces lettres désignant une onde.

- L'onde P correspond à la contraction des oreillettes, sa durée est d'environ 110 ms ;
- le complexe QRS correspond à la contraction des ventricules et sa durée est habituellement d'environ à 100 ms ;
- l'onde T correspond à la relaxation des ventricules.

Les caractéristiques classiques du complexe PQRST d'un patient sain et avec un rythme cardiaque de 60 battements par minute sont décrites dans [97] et résumées dans le Tableau 2.1.

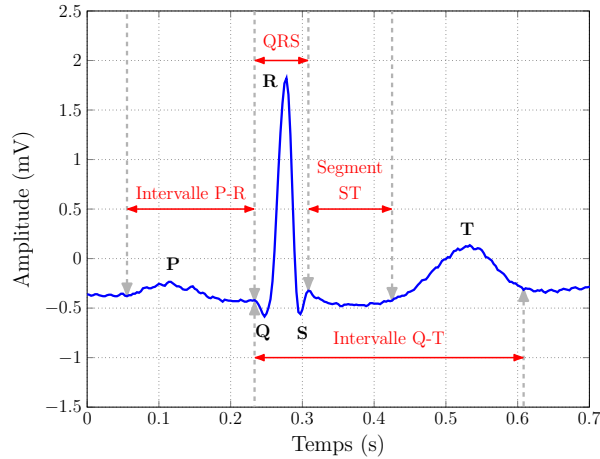


FIGURE 2.5 – Exemple de période de signal ECG

Il existe différentes informations spécifiques aux signaux ECG qui peuvent être utilisées pour la détection d'anomalies. Le paramètre le plus couramment utilisé dans la littérature, pour la classification des battements cardiaques, est le rythme cardiaque, également appelé intervalle R-R [98]. À l'exception des patients porteurs d'un stimulateur cardiaque, les variations de la durée de cet intervalle sont corrélées aux variations de la morphologie des ondes [99]. L'utilisation d'un intervalle R-R normalisé permet d'améliorer significativement les performances de classification [100, 101]. D'autres distances entre les différents points de repère d'un battement cardiaque, illustrés à la Figure 2.5, sont également utilisées dans la littérature, comme la durée du complexe QRS ou les intervalles Q-T et P-R [102].

Dans cette étude, nous ne chercherons pas à déterminer les valeurs de ces paramètres qui sont spécifiques aux signaux ECG. Comme évoqué précédemment, nous extrairons des informations temporelles et fréquentielles du signal grâce à la transformée en ondelettes, ce

TABLE 2.1 – Caractéristiques classiques du complexe PQRST pour un rythme cardiaque de 60 battements par minutes [97]

Caractéristique	Valeur normale	Variation normale
Largeur de l'onde P	110 ms	$\pm 20$ ms
Intervalle PR	160 ms	$\pm 40$ ms
Largeur du complexe QRS	100 ms	$\pm 20$ ms
Intervalle QT	400 ms	$\pm 40$ ms
Amplitude de l'onde P	0.115 mV	$\pm 0.05$ mV
Amplitude de l'onde QRS	1.5 mV	$\pm 0.5$ mV
Amplitude de l'onde T	0.3 mV	$\pm 0.2$ mV

qui permet d'obtenir une méthode générique applicable à différents types de signaux. Ces mesures pourront être directement utilisées par un classifieur sans passer par une étape de reconstruction ou de détermination des paramètres physiques évoqués précédemment. Par la suite, nous appellerons paramètres les mesures effectuées par transformée en ondelettes sur le signal ECG.

Dans le but de démontrer l'efficacité des convertisseurs A2F dans le contexte des capteurs sans-fil, ce concept a été utilisé pour la détection d'anomalies ECG. Pour cela, les signaux de la base de données MIT-BIH Arrhythmia [103] ont été utilisés. Cette base de données est composée de 48 enregistrements ECG de 30 minutes chacun, échantillonnés à une fréquence de 360 Hz, auprès de 47 patients différents. Parmi ces 48 enregistrements, 34 ont été utilisés comme ensemble de données d'apprentissage et 14 ont été utilisés comme ensemble de données de test. Cette répartition 70/30%, entre les ensembles d'entraînement et de test, constitue une répartition classique dans les problèmes de ML. L'ensemble de données résultant contient 107 044 battements annotés. La fenêtre d'analyse des signaux ECG et donc la fenêtre temporelle maximale des ondelettes est fixée à une taille  $N = 256$  échantillons (soit environ 710 ms, pour une fréquence d'échantillonnage de 360 Hz), qui correspondent approximativement à la durée d'un battement, comme le montrent la Figure 2.5 et le Tableau 2.1. L'objectif étant de détecter des anomalies dans les battements, chaque bloc de données contient un complexe PQRST annoté, où le complexe QRS est centré au niveau du centième échantillon.

### 2.1.3 Échantillonnage non-uniforme en ondelettes

L'échantillonnage non-uniforme en ondelettes a été proposé dans [57] comme technique d'échantillonnage compressif pour l'acquisition de signaux radiofréquence. Cette technique consiste en l'acquisition non-uniforme de coefficients de transformées en ondelettes du signal observé.

#### 2.1.3.1 Rappels sur la transformée en ondelettes

Une ondelette  $\xi$  est une forme d'onde continue qui est limitée dans le temps, qui a une valeur moyenne égale à zéro.

$$\int_{-\infty}^{+\infty} \xi(t) dt = 0 \quad (2.1)$$

C'est une fonction appartenant à l'ensemble des fonctions de carré intégrable, c'est-à-dire que sa norme  $\mathcal{L}^2$  est bornée :

$$\int_{\mathbb{R}} |\xi(t)|^2 dt < +\infty \quad (2.2)$$

Les ondelettes ont été introduites pour le traitement du signal par Morlet [104] en 1982, qui a montré que les fonctions continues dans  $\mathcal{L}^2$  peuvent être représentées par une ondelette  $\xi_{a,b}(t)$  qui est obtenue grâce à une opération de mise à l'échelle ( $a \in \mathbb{R}^+$ ) et par translation

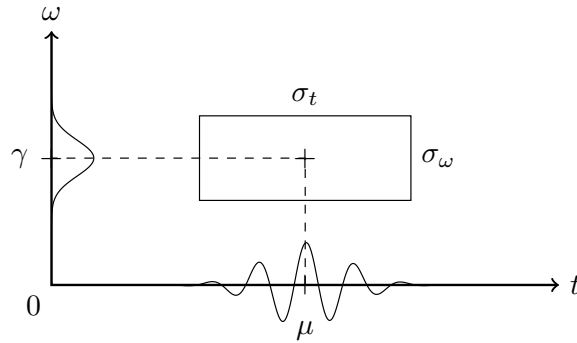


FIGURE 2.6 – Boîte de Heisenberg d'une ondelette

( $b \in \mathbb{R}$ ) d'une ondelette mère  $\xi(t)$ . Les opérations de mise à l'échelle et de décalage temporel peuvent être formalisées comme suit :

$$\xi_{a,b}(t) = \frac{1}{\sqrt{a}} \xi\left(\frac{t-b}{a}\right), \quad a \in \mathbb{R}^+, b \in \mathbb{R} \quad (2.3)$$

La transformée en ondelette d'une fonction  $f(t)$  s'écrit de la façon suivante :

$$\mathcal{W}f_{a,b} = \langle f, \xi_{a,b} \rangle = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t) \cdot \xi^*\left(\frac{t-b}{a}\right) dt = f \star \bar{\xi}_a(b) \quad (2.4)$$

avec  $\bar{\xi}_a(t) = \frac{1}{\sqrt{a}} \xi^*\left(\frac{-t}{a}\right)$  et où  $\star$  est l'opérateur du produit de convolution.

La transformée en ondelette permet d'obtenir des informations à la fois temporelles et fréquentielles sur le signal  $f(t)$ . Plus le support de l'ondelette sera large, plus l'information en fréquence sera précise. À l'inverse avec un support très court, l'information sera précise en temps au détriment de la précision en fréquence. Cette affirmation se vérifie en utilisant les boîtes de Heisenberg [105]. On appelle *atomes temps-fréquence* ( $\phi_k$ ) une famille de fonctions concentrées en temps et en fréquence. On suppose que  $\phi_k \in \mathcal{L}^2$  et que :

$$\|\phi_k\|^2 = \int_{-\infty}^{+\infty} |\phi_k(t)|^2 dt = 1 \quad (2.5)$$

La fonction  $|\phi(t)|^2$  peut ainsi être considérée comme une densité de probabilité de moyenne  $\mu$  et de variance  $\sigma_t$  :

$$\mu = \int_{-\infty}^{+\infty} t |\phi_k(t)|^2 dt \quad (2.6)$$

$$\sigma_t^2 = \int_{-\infty}^{+\infty} (t - \mu)^2 |\phi_k(t)|^2 dt \quad (2.7)$$



L'égalité de Parseval nous rappelle que si  $\phi$  est dans  $\mathcal{L}^1(\mathbb{R}) \cap \mathcal{L}^2(\mathbb{R})$ , alors :

$$\int_{-\infty}^{+\infty} |\phi_k(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |\hat{\phi}_k(\omega)|^2 d\omega \quad (2.8)$$

où  $\hat{\phi}_k(\omega)$  est la transformée de Fourier de  $\phi_k(t)$ . On peut ainsi en déduire que :

$$\int_{-\infty}^{+\infty} |\hat{\phi}_k(\omega)|^2 d\omega = 2\pi \|\phi_k\|^2 \quad (2.9)$$

Ceci permet de définir la moyenne  $\gamma$  et la variance  $\sigma_\omega$  de  $\hat{\phi}_k$  :

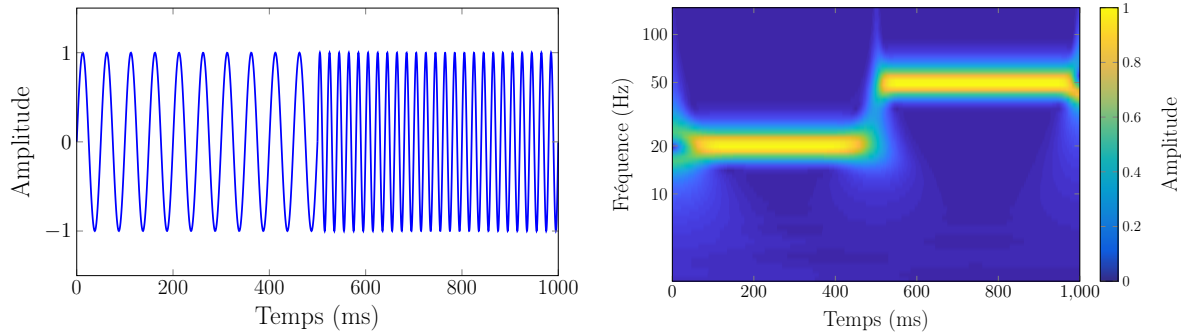
$$\gamma = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \omega |\hat{\phi}_k(\omega)|^2 d\omega \quad (2.10)$$

$$\sigma_\omega^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (\omega - \gamma)^2 |\hat{\phi}_k(\omega)|^2 d\omega \quad (2.11)$$

La résolution en temps et en fréquence de  $\phi$  peut être représenté par une boîte de Heisenberg, centrée en  $(\mu, \gamma)$  et de largeurs  $(\sigma_t, \sigma_\omega)$ , dans le plan  $(t, \omega)$ , comme représenté à la Figure 2.6.

Le théorème d'incertitude de Heisenberg [105, 106] montre que la surface de ce rectangle, de largeur  $(\sigma_t, \sigma_\omega)$  est supérieure ou égale à  $1/2$ . Dans l'idéal, nous voulons tendre vers des valeurs de  $\sigma_t$  et  $\sigma_\omega$  les plus petites possibles pour avoir les informations les plus précises possibles en temps et en fréquence. Ce qui est impossible, car ce théorème impose que le produit de ces deux valeurs soit supérieur à  $1/2$ , ce qui limite la résolution simultanée en temps et en fréquence de la fonction  $\phi$ .

Dans la Figure 2.7, le signal analysé est une sinusoïde dont la fréquence change au cours du temps, la transformée en ondelette permet de déterminer la fréquence du signal, mais aussi l'instant auquel ce changement de fréquence intervient.



(a) Représentation temporelle

(b) Représentation temps/fréquence

FIGURE 2.7 – Transformée en ondelettes continue

### 2.1.3.2 Dictionnaires d'ondelettes

Pour mettre en œuvre le NUWS, nous avons testé deux types d'ondelettes : les ondelettes de Haar et les ondelettes de Gabor. Ce sont les principales familles d'ondelettes pouvant être réalisées de manière analogique. Les ondelettes de Haar sont une famille d'ondelettes proposée par Alfred Haar en 1909 [107]. Elles sont considérées comme les premières ondelettes connues et sont les ondelettes les plus simples. C'est une famille de fonctions carrées qui prennent les valeurs -1, 0 et 1. L'avantage des ondelettes de Haar réside dans leur simplicité ce qui permet de les générer facilement. L'ondelette mère de Haar peut être décrite comme suit :

$$\xi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \\ 0 & \text{sinon.} \end{cases} \quad (2.12)$$

La base des ondelettes de Haar peut alors s'écrire :

$$\left\{ \xi_{i,j}(t) = \frac{1}{\sqrt{2^i}} \xi\left(\frac{t - 2^i \cdot j}{2^i}\right) \right\}_{(i,j) \in \mathbb{Z}^2} \quad (2.13)$$

Les ondelettes de Gabor sont une famille de fonctions à valeur complexe. Ces ondelettes sont le résultat du produit d'une exponentielle complexe avec une fenêtre gaussienne. Elles peuvent être décrites par la fonction :

$$\xi_{f,\delta,\tau}(t) = \frac{2^{\frac{1}{4}}}{\sqrt{\tau} \pi^{\frac{1}{4}}} e^{j2\pi f(t-\delta)} \cdot e^{-\left(\frac{t-\delta}{\tau}\right)^2}, \quad (2.14)$$

où  $\tau$  est une variable permettant de régler la largeur de la gaussienne et  $\delta$  est une variable correspondant à un décalage temporel. L'avantage des ondelettes de Gabor est que la fenêtre utilisée est une gaussienne aussi bien dans le domaine temporel que dans le domaine fréquentiel, en revanche, la génération d'une fonction gaussienne paramétrable peut se révéler complexe. La Figure 2.8 représente un exemple d'ondelette pour chacune de ces deux familles.

La dernière famille d'ondelettes dont on peut parler sont les ondelettes de Daubechies : c'est une famille d'ondelettes orthogonales définissant une transformée en ondelettes discrète. Les ondelettes de Daubechies ne peuvent pas être définies sous forme d'équation, elles sont définies par les différents coefficients d'un filtre associé. Elles sont caractérisées par un nombre maximal  $A$  de moments nuls pour un support donné, qui correspond à la moitié du nombre de coefficients. La décomposition en ondelettes est effectuée par un processus de filtrage récursif fournissant une information, dans le domaine fréquentiel, de plus en plus précise à chaque étape. Cette décomposition récursive divise le spectre fréquentiel en deux sous-bandes, grâce à l'utilisation de deux filtres (un passe-haut et un passe-bas), les signaux résultants sont alors

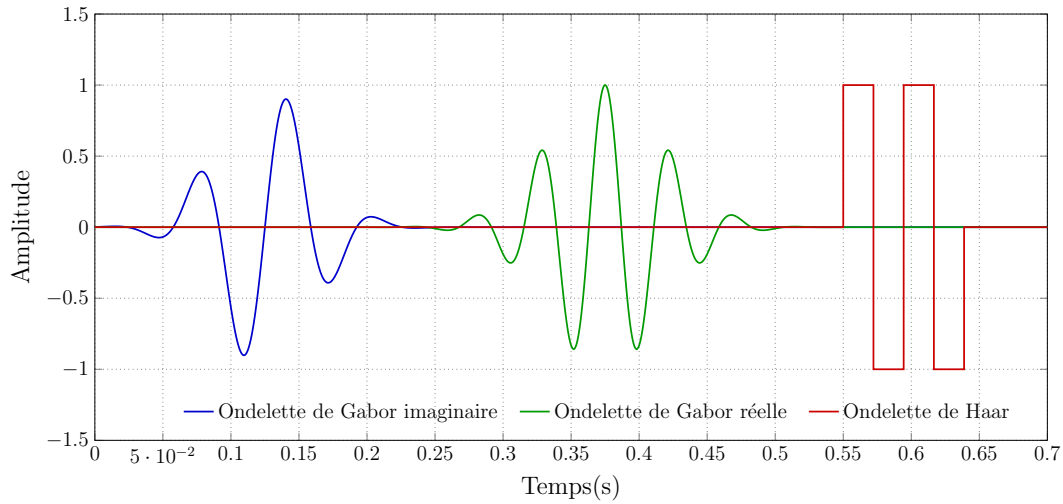


FIGURE 2.8 – Exemple d'ondelettes utilisées

sous-échantillonnés d'un facteur deux. Dans cette étude, nous avons considéré la fonction Daubechies *db4* pour le NUWS, qui semble être le meilleur candidat en raison de sa forme similaire à celle du signal ECG, comme le montre la Figure 2.9.

Un dictionnaire d'ondelettes contient des ondelettes avec différentes fréquences, décalages temporels ou fenêtre temporelle. Chacune de ces trois grandeurs pouvant prendre une infinité de valeurs, la taille du dictionnaire ainsi obtenu, et donc le nombre de paramètres extractibles, est potentiellement infini. L'objectif étant de diminuer le nombre d'échantillons acquis, il n'est donc pas concevable de considérer un ensemble de paramètre comme potentiellement infini. Certaines contraintes ont donc été imposées sur les valeurs de ces trois grandeurs.

Pour construire le dictionnaire d'ondelettes de Haar, nous choisissons de balayer la plage de fréquence allant de  $F_m \simeq 1.4$  Hz, qui correspond à une période sur le vecteur de dimension

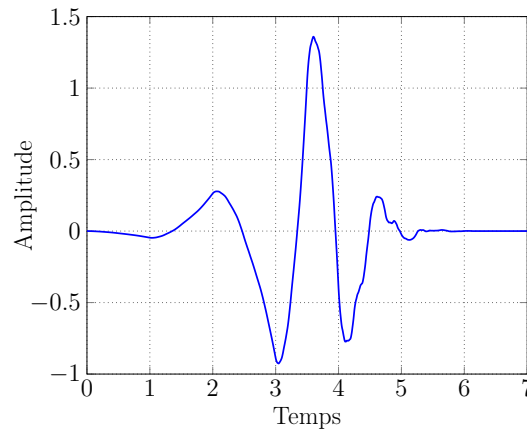


FIGURE 2.9 – Ondelette de Daubechies db4

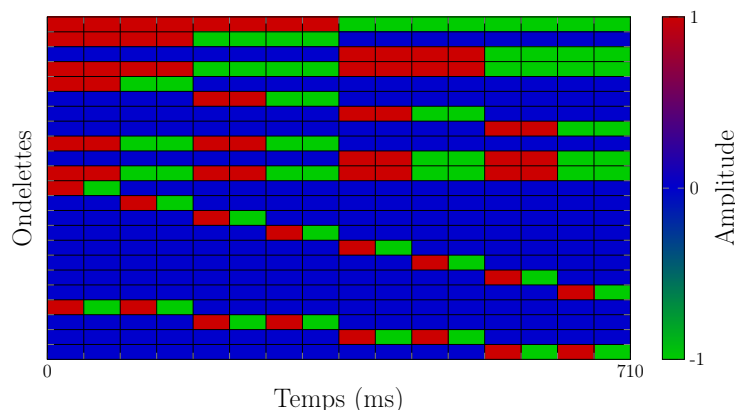


FIGURE 2.10 – Exemples d'ondelettes de Haar constituant le dictionnaire utilisé, où chaque ligne représente une ondelette

$N = 256$ , comme énoncé dans la Section 2.1.2, à une fréquence  $F_M = 180$  Hz, en multipliant à chaque fois la fréquence par deux jusqu'à la fréquence maximale. Ensuite, nous considérons différentes tailles de fenêtre temporelle, à partir d'une fenêtre complète de  $N$  échantillons, puis en la divisant par deux jusqu'à une taille de  $N/128 = 2$  échantillons, ce qui permet, pour une fréquence donnée, d'occuper différentes tailles de support temporel. Nous imposons également que les ondelettes aient un nombre fini de périodes sur ce support temporel. Enfin, pour une fréquence et une taille de fenêtre temporelle données, nous avons décidé que les différentes ondelettes ne pourraient pas se chevaucher dans le temps, afin de limiter le nombre total d'ondelettes et limiter les redondances. Ces choix conduisent à un dictionnaire de 502 ondelettes pour les ondelettes Haar. La Figure 2.10 représente les premières ondelettes de Haar du dictionnaire ainsi constitué.

Pour le dictionnaire d'ondelettes de Gabor, nous avons utilisé un processus similaire. Nous avons d'abord défini un nombre minimal de périodes visibles dans l'ondelette. Cette valeur définit les fréquences minimale et maximale d'une ondelette dans notre dictionnaire, mais également la taille minimale de l'enveloppe gaussienne. Comme l'enveloppe gaussienne n'est jamais totalement nulle, des bornes ont été définies pour définir ses limites de taille et de positions. Afin de limiter le nombre d'ondelettes, un décalage temporel minimal est défini, car la notion de "chevauchement" est plus floue pour les ondelettes de Gabor que pour les ondelettes de Haar. Comme pour les ondelettes de Haar, la fréquence de l'ondelette est multipliée par deux à chaque itération jusqu'à une fréquence maximale, cette dernière ne dépassant pas la limite de 180 Hz. De plus, pour une fréquence donnée un nombre maximal de périodes est fixé de manière à ce qu'un nombre entier de périodes apparaissent sur le support temporel, en plus du nombre minimal de périodes. Le dictionnaire créé contient, en tenant compte des parties réelles et imaginaires, 2534 ondelettes. La Figure 2.11 représente quelques ondelettes constituant le dictionnaire d'ondelettes de Gabor.

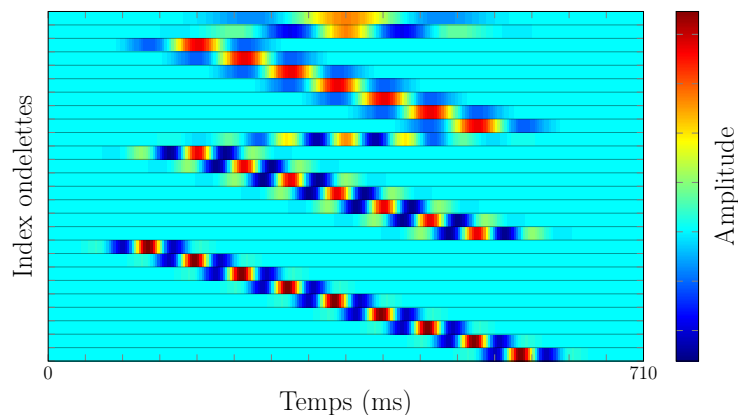


FIGURE 2.11 – Exemples d'ondelettes de Gabor constituant le dictionnaire utilisé, où chaque ligne représente une ondelette

Comme mentionné précédemment, la fonction Daubechies *db4* a été considérée pour effectuer l'extraction de caractéristiques en raison de sa ressemblance avec le signal ECG. Nous avons effectué une décomposition en paquets d'ondelettes sur 8 niveaux ( $\log_2(256)$ ), fournissant les informations de fréquence les plus précises sur tout le spectre. Dans une transformée en ondelettes discrète classique, chaque niveau de décomposition est calculé en faisant passer uniquement le signal issu du filtre passe-bas de l'étape précédente. Dans une décomposition en paquets d'ondelettes, chaque niveau de décomposition est calculé en faisant passer les signaux issus des filtres passe-bas et passe-haut de l'étape précédente.

La taille des dictionnaires ainsi formés est telle que le nombre de paramètres extraits, et donc d'échantillons, est plus grand que le nombre d'échantillons obtenus avec une acquisition classique à la fréquence de Nyquist. Ce qui est en contradiction avec l'objectif initial qui est de diminuer la quantité de données. La section suivante présente donc des méthodes permettant de sélectionner, parmi ces dictionnaires, les ondelettes utiles pour la classification et ainsi réduire au minimum le nombre d'échantillons.

## 2.2 Algorithmes de sélection de paramètres

L'espace des ondelettes possibles est extrêmement important, il faut donc trouver des méthodes pour réduire le nombre d'ondelettes, en supprimant celles qui ne fournissent pas d'informations ou celles qui fournissent des informations déjà disponibles tout en conservant de bonnes performances de classification. Dans le domaine de l'apprentissage automatique, les algorithmes de sélection de paramètres sont des processus qui permettent la sélection d'un sous-ensemble de paramètres pour la construction de modèles. Il existe plusieurs types d'algorithmes de sélection de paramètres, tels que les méthodes par filtrage (Filter methods) ou les méthodes par encapsulation (Wrapper methods).

Les méthodes par filtrage sont généralement plus rapides que les méthodes par encapsulation qui sont extrêmement coûteuses à exécuter pour des ensembles de données avec un grand nombre de paramètres. Cependant, elles offrent de meilleures solutions avec des performances de classification plus élevées. La solution que nous proposons consiste à combiner ces méthodes, tout d'abord en utilisant une méthode par filtrage afin de trouver un ensemble réduit de paramètres, puis en exécutant une méthode par encapsulation. Nous allons ici présenter différentes méthodes de sélection de paramètres.

### 2.2.1 Méthodes par filtrage

Le principe de fonctionnement des méthodes par filtrage consiste à classer les paramètres selon certains critères, puis à utiliser ceux ayant les scores les plus élevés pour construire un modèle de classification.

#### 2.2.1.1 Gain d'information

Le gain d'information (Information Gain, IG) [86, 87, 108] est l'un des critères de sélection de paramètres les plus connus, il est notamment utilisé dans certains algorithmes de construction d'arbres de décisions comme ID3. C'est une méthode dite supervisée, c'est-à-dire qu'elle nécessite de connaître la classe des exemples traités. Un paramètre est plus pertinent qu'un autre s'il possède un gain d'information plus élevé. Pour un ensemble de données  $S$ , on peut exprimer le gain d'information d'un paramètre  $A$  par la formule :

$$IG(S, A) = H(S) - \sum_{i=1}^K \frac{|S_i|}{|S|} \cdot H(S_i) , \quad (2.15)$$

où  $K$  est le nombre de valeurs possible du paramètre  $A$ ,  $S_i$  est un sous-ensemble de  $S$  dans lequel le paramètre  $A$  prend la valeur  $A_i$ ,  $|S_i| / |S|$  correspond donc à la proportion d'exemples de l'ensemble  $S$  pour lesquels le paramètre  $A$  prend la valeur  $A_i$  et  $H(S)$  est l'entropie de Shannon. L'entropie  $H(S)$  est définie par la relation :

$$H(S) = \sum_{j=1}^C -p_j \cdot \log_2(p_j) \quad (2.16)$$

où  $C$  correspond au nombre de classes possible et  $p_j$  correspond à la proportion d'exemples appartenant à la classe  $j$ . Dans le contexte d'une détection binaire d'anomalies dans les signaux ECG, le nombre de classes  $C$  est égal à deux : les deux classes possibles étant *Vrai* ou *Faux*. Les ensembles  $S_i$  sont donc les sous-ensembles de données contenant respectivement les exemples de ces deux classes. Parmi l'ensemble d'apprentissage, 48753 exemples sont classés comme normaux (soit environ 64%) et 27379 comme anormaux (soit 36%). Cette formule du gain d'information étant adaptée aux variables qualitatives, la grandeur  $K$  correspond au nombre d'intervalles permettant de catégoriser une variable quantitative, par la suite, nous avons choisi

une valeur de  $K = 12$ . Dans le cas où les paramètres n'ont pas tous le même nombre de valeurs, le gain d'information favorise la sélection des paramètres ayant un grand nombre de valeurs. Il existe ainsi une extension du gain d'information, appelée le ratio de gain d'information, qui permet de corriger ce biais. Il introduit un facteur qui va permettre de pénaliser les paramètres qui partitionnent fortement et qui correspond à l'entropie du paramètre dans l'ensemble  $S$ .

### 2.2.1.2 Score de Fisher

Le score de Fisher (Fisher Score, FS) [108] est un algorithme de sélection de paramètres largement utilisé. Il permet d'évaluer des paramètres et de trouver un sous-ensemble tel que, pour un paramètre donné, la distance entre les échantillons d'une même classe soit la plus petite possible tandis que la distance entre les échantillons de classes différentes est la plus grande possible. En d'autres termes, les échantillons d'une même classe doivent avoir la même valeur et les échantillons de différentes classes doivent avoir des valeurs différentes. Ce score est calculé comme suit :

$$FS_i = \frac{\sum_{j=1}^C n_j \cdot (\mu_{ij} - \mu_i)^2}{\sum_{j=1}^C n_j \cdot \rho_{ij}^2}, \quad (2.17)$$

avec  $C$  le nombre de classes,  $n_j$  le nombre d'exemples appartenant à la classe  $j$ ,  $\mu_i$  la moyenne du paramètre  $i$ ,  $\mu_{ij}$  la moyenne du paramètre  $i$  pour les exemples de la classe  $j$  et  $\rho_{ij}$  la variance du paramètre  $i$  pour la classe  $j$ .

TABLE 2.2 – Exemples de scores de Fisher pour deux paramètres extraits avec des ondelettes de Haar

$\mu$	$\mu_{Faux}$	$\mu_{Vrai}$	$\rho_{Faux}$	$\rho_{Vrai}$	$SF$
$4.7 \times 10^{-3}$	$-2 \times 10^{-2}$	$4.9 \times 10^{-2}$	$2.4 \times 10^{-3}$	$5 \times 10^{-3}$	79
$5.3 \times 10^{-2}$	$5.3 \times 10^{-2}$	$5.38 \times 10^{-2}$	0.166	0.205	$4.16 \times 10^{-6}$

Le Tableau 2.2 présente deux exemples de calculs du score de Fisher pour deux paramètres extraits avec deux ondelettes de Haar différentes. On constate que le premier paramètre, dont les moyennes respectives pour chaque classe sont plus éloignées et dont les variances sont plus faibles, obtient bien un score de Fisher plus grand, car la limite entre les données des deux classes est plus marquée.

### 2.2.1.3 Relief - Relief-F

La famille d'algorithme Relief [109] permet de déterminer si les valeurs prises par un paramètre permettent de distinguer les exemples d'une même classe et les exemples de classes différentes. L'algorithme original Relief ne permet de traiter que les problèmes de classification avec deux classes. L'algorithme Relief-F est une extension de Relief qui permet de traiter des données ayant plus de deux classes. R-Relief-F est une adaptation de Relief-F pour les problèmes de régression.

L'algorithme Relief sélectionne un exemple  $R_i$  dans l'ensemble de données  $S$  d'entraînement et recherche les deux exemples les plus proches, l'un appartenant à la même classe, appelé  $H$  (Nearest Hit) et l'autre appartenant à la seconde classe, appelé  $M$  (Nearest Miss). À partir des valeurs des paramètres de ces trois exemples, on peut actualiser l'estimation de la qualité du paramètre. Si, pour un paramètre  $A$ , les instances  $R_i$  et  $H$  ont des valeurs différentes alors l'estimation de qualité diminue. De la même façon, si les instances  $R_i$  et  $M$  ont des valeurs différentes pour le paramètre  $A$ , alors l'estimation de qualité augmente. Ce processus d'évaluation est effectué  $m$  fois ( $m \leq$  Nombre d'exemple), en sélectionnant à chaque fois l'instance  $R_i$  de manière aléatoire.

Pour mesurer la différence entre deux instances (exemples)  $I_1$  et  $I_2$  de l'ensemble de données  $S$ , pour un paramètre  $A$ , on définit la fonction  $\text{diff}(A, I_1, I_2)$  et la fonction  $\text{val}(A, I_k)$  qui correspond à la valeur du paramètre  $A$  de l'exemple  $I_k$ .

Pour des variables qualitatives :

$$\text{diff}(A, I_1, I_2) = \begin{cases} 0 & \text{si } \text{val}(A, I_1) = \text{val}(A, I_2) \\ 1 & \text{sinon} \end{cases} \quad (2.18)$$

Pour des variables quantitatives :

$$\text{diff}(A, I_1, I_2) = \frac{|\text{val}(A, I_1) - \text{val}(A, I_2)|}{\max(A) - \min(A)} \quad (2.19)$$

Cette fonction est utilisée pour trouver les instances les plus proches  $H$  et  $M$  : la distance entre deux instances étant égale à la somme des distances pour chaque paramètre. Elle sert également lors de la mise à jour de l'estimation de la qualité d'un paramètre  $W[A]$ .

$$W_i[A] = W_{i-1}[A] - \frac{\text{diff}(A, R_i, H)}{m} + \frac{\text{diff}(A, R_i, M)}{m} \quad (2.20)$$

Le principe de l'algorithme Relief-F est le même que pour l'algorithme Relief, mais des améliorations ont été apportées afin de pouvoir traiter les données ayant plus de deux classes, traiter les données incomplètes, c'est-à-dire les exemples où il manque la valeur d'un ou plusieurs paramètres, et augmenter sa résistance aux données bruitées. Après avoir choisi aléatoirement une instance  $R_i$ , on recherche ses  $k$  plus proches voisins appartenant à la même classe ( $H_j$ ) et ses  $k$  plus proches voisins pour chacune des autres classes ( $M_j(C)$ ).

La méthode pour actualiser l'estimation de la qualité de chaque paramètre reste similaire, mais en faisant une moyenne des distances pour les éléments de  $H_j$  et pour les éléments de  $M_j$ . La contribution de chaque classe de  $M_j$  est pondérée par sa probabilité d'apparition parmi toutes les classes de  $M_j$ , dans l'ensemble d'apprentissage.

Les méthodes par filtrage ont pour avantage de ne pas nécessiter de calculs complexes, de ne pas nécessiter l'apprentissage de classifieurs et d'être rapide. En revanche, les paramètres sont



évalués indépendamment les uns des autres et elles ne prennent pas en compte leur influence sur les performances de classification.

## 2.2.2 Méthodes par encapsulation

Les méthodes par encapsulation utilisent les performances de classification d'un classifieur spécifique (arbre de décision, réseau de neurones, ...) pour évaluer un ensemble de paramètres. Les différentes étapes d'une sélection par encapsulation consistent à sélectionner un sous-ensemble de paramètres, à l'évaluer en entraînant un classificateur et à mesurer ses performances. Ces étapes sont répétées jusqu'à ce que les performances souhaitées ou le nombre de paramètres sélectionnés soit atteint ou jusqu'à ce qu'il n'y ait plus de solution.

### 2.2.2.1 Algorithmes de recherche séquentielle

La recherche exhaustive, qui consiste à tester toutes les solutions possibles, devient rapidement trop complexe lorsque le nombre total de paramètres augmente. Le Sequential Forward Search (SFS) [110], présenté dans l'Algorithme 1, est un exemple de modèle par encapsulation. Le principe de cet algorithme est le suivant : on commence avec un ensemble  $S$  de paramètres sélectionnés vide et chaque paramètre est évalué individuellement selon un critère  $J_i$ . Ce critère correspond à la précision de classification d'un modèle choisi. Le paramètre ayant la meilleure précision est alors sélectionné. Ensuite, à chaque étape, l'algorithme évalue les différentes combinaisons réalisées à partir de l'ensemble des paramètres sélectionnés  $S$  et des paramètres restants  $X_i$ . Ici, la condition d'arrêt de l'algorithme est un nombre maximal de paramètres sélectionnés ( $D$ ), mais il peut également être arrêté lorsque aucun des paramètres restants ne permet d'améliorer les performances.

Le Sequential Backward Search (SBS) reprend le même principe, mais au lieu de partir d'un ensemble de vide, l'ensemble de départ comprend tous les paramètres puis on supprime successivement les moins bons paramètres. Le principal inconvénient de ces deux algorithmes est l'effet « d'imbrication » car une fois qu'un paramètre a été choisi, il n'y a pas de retour en

---

**Algorithm 1** Sequential Forward Search

---

```
1:  $S \leftarrow \emptyset$ 
2: repeat
3:   for all  $X_i \notin S$  do
4:      $J_i \leftarrow J(S \cup \{X_i\})$ 
5:    $i' \leftarrow \arg \max (J_i)$ 
6:    $S' \leftarrow S \cup \{X_{i'}\}$ 
7:   if  $J(S') > J(S)$  then
8:      $S \leftarrow S'$ 
9:      $J(S) \leftarrow J(S')$ 
10: until  $|S| == D$ 
```

---

arrière possible. L'algorithme Sequential Floating Forward Search (SFFS) a été proposé dans le but de corriger cet effet. La méthode SFFS [110] consiste en un mixte entre les algorithmes SFS et SBS. Comme dans le SFS l'ensemble de départ est vide puis on choisit les meilleurs paramètres. Mais après chaque paramètre ajouté, on regarde si la suppression de certains paramètres déjà sélectionnés permet d'améliorer les performances de classification. On peut alors supprimer des paramètres tant que cela permet d'améliorer les performances.

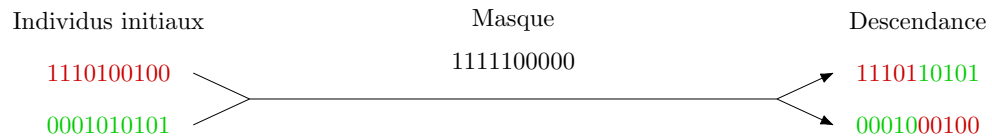
### 2.2.2.2 Algorithme génétique

Les algorithmes génétiques [111] sont des algorithmes appartenant à la famille des algorithmes évolutionnistes dont le but est de faire évoluer un ensemble de solutions à un problème afin d'aboutir aux meilleures solutions. Le processus d'évolution de l'ensemble de solutions est basé sur les mécanismes de la sélection naturelle et de la génétique. Dans un algorithme génétique, un individu représente une solution au problème posé et l'ensemble des individus constitue la population. Dans le cadre d'un problème de sélection de paramètres, un individu représente donc un sous-ensemble de paramètres candidats et peut être représenté par un vecteur de  $n$ -bits (où  $n$  représente le nombre total de paramètres disponibles), quand un bit est à 1, cela signifie que le paramètre correspondant est sélectionné. La population de départ est une population aléatoire qui subit par la suite des opérations génétiques successives, chacune aboutissant à une nouvelle génération. À chaque génération, on mesure le degré d'adaptation de chaque individu. Pour cela, on utilise une fonction de fitness qui va évaluer la solution suivant certains critères (dans le cas de la sélection de paramètres cela peut être la précision d'un classifieur appris ou le coût d'extraction d'un paramètre...). D'une génération à l'autre, on cherche alors à conserver les meilleures solutions. Il existe de nombreuses opérations génétiques, mais les plus connues sont :

- La reproduction : le nombre de descendants d'un individu dépend de son adaptation au problème.
- Le croisement (cross-over) avec un ou plusieurs points
- La mutation

La méthode de la roulette est une méthode de reproduction dans laquelle, pour chaque individu, on calcule son degré d'adaptation  $f_i$  et la probabilité qu'il a de se reproduire  $p_i = \frac{f_i}{\sum f_i}$ . On réalise alors  $k$  tirages parmi la population, via une roulette biaisée où chaque individu a une probabilité  $p_i$  d'être sélectionné, pour constituer la nouvelle génération. La mutation d'un individu consiste en la modification de la valeur d'un bit choisi aléatoirement dans un vecteur (individu). Le croisement (Figure 2.12) est une méthode qui fait intervenir deux individus de taille  $n$ . On choisit aléatoirement un entier  $j$  compris entre 1 et  $n - 1$ . Cet entier est appelé point de croisement. À partir des deux individus de départ, on crée deux nouveaux individus en intervertissant les valeurs, de chacun des vecteurs, situées entre  $j + 1$  et  $n$ . Le croisement et la mutation ne sont pas des processus systématiques lors du passage d'une génération à l'autre

**Un point de croisement**



**Deux points de croisement**

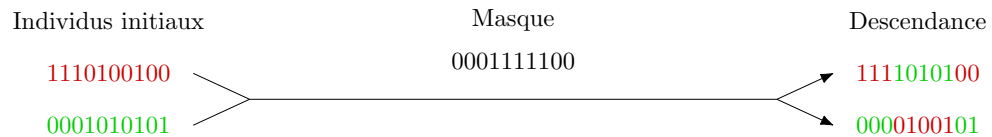


FIGURE 2.12 – Exemple de croisement entre deux individus codés sur 10 bits avec respectivement 1 point (5) et 2 points (3 et 8) de croisement

et interviennent avec une certaine probabilité. Le taux de mutation d'un caractère est généralement très faible : de l'ordre de  $1/1000$ . Dans les algorithmes génétiques, on souhaite travailler avec une population de taille constante, or, après avoir appliqué les opérations génétiques sur une population, celle-ci est alors constituée des individus de la précédente génération et de leurs descendants. Il existe donc plusieurs stratégies de sélection des individus de la nouvelle génération :

- Une stratégie dite générationnelle où les descendants remplacent les parents.
- Une stratégie dite «  $k$ -élitiste » où l'on garde seulement les  $k$  meilleurs individus sans distinction entre parents et descendants.
- Une stratégie où, à chaque génération, un certain pourcentage des parents est remplacé. Pour choisir les individus qui sont conservés, on peut alors soit décider de conserver strictement les meilleurs soit d'utiliser la méthode de la roulette.

L'utilisation d'algorithmes génétiques appliquée à la sélection de paramètres nécessite d'apprendre et de tester un classifieur pour chaque individu à chaque génération pour mesurer son degré d'adaptation ce qui peut entraîner des temps de calcul importants. En revanche, cela permet d'avoir une sélection adaptée au type de classifieur choisi. Cette méthode permet d'obtenir une solution avec de bonnes performances de classification, mais pas forcément d'obtenir une solution performante avec le moins de paramètres possible. Le croisement entre individus permet de mettre en place un « échange d'informations » entre les solutions du problème. La mutation, du fait de sa dimension aléatoire, permet d'éviter d'oublier de tester certains paramètres et peut permettre de sortir d'un minimum local. Les processus de reproduction et de sélection permettent de conserver les informations importantes présentes dans les individus d'une génération.

### 2.2.3 Conclusion sur les algorithmes de sélection de paramètres

Deux types d'algorithmes de sélection de paramètres ont été identifiés dans cette section : les méthodes par filtrage et les méthodes par encapsulation. Le Tableau 2.3 résume les avantages et inconvénients de ces deux méthodes. Les méthodes par filtrage sont généralement plus rapides que les méthodes par encapsulation qui sont extrêmement coûteuses à exécuter pour des ensembles de données avec un grand nombre de paramètres. Cependant, elles offrent de meilleures solutions avec des performances de classification plus élevées. Dans la suite de ce Chapitre, nous avons décidé de nous limiter à deux méthodes de filtrage, le score de Fisher (FS) et le gain d'information (IG), car facilement réalisables sous MATLAB<sup>®</sup>, et une méthode par encapsulation, le Sequential Forward Search (SFS), car elle permet de contrôler et donc minimiser le nombre de paramètres sélectionnés.

## 2.3 Sélection de paramètres pour le NUWS

Les deux dictionnaires d'ondelettes présentés plus tôt, ont été utilisés dans le processus d'extraction de paramètres de notre base de données pour ensuite être évalués avec différents classificateurs : les réseaux de neurones (Neural Network, NN), les forêts aléatoires (Random Forest, RF) et les machines à vecteurs de support (Support Vector Machine, SVM). Comme expliqué plus tôt, la fenêtre d'analyse des signaux ECG est fixée à une taille  $N = 256$  échantillons. Les 107 044 périodes de 256 échantillons ont donc été mélangées avec toutes les ondelettes des deux dictionnaires. Les données obtenues sont divisées en deux sous-ensembles : un ensemble d'apprentissage, constitué de 70% des patients, et d'un ensemble de test, constitué des 30% des patients restant. Dans cette étude, une classification binaire est effectuée pour détecter la présence ou non d'anomalies.

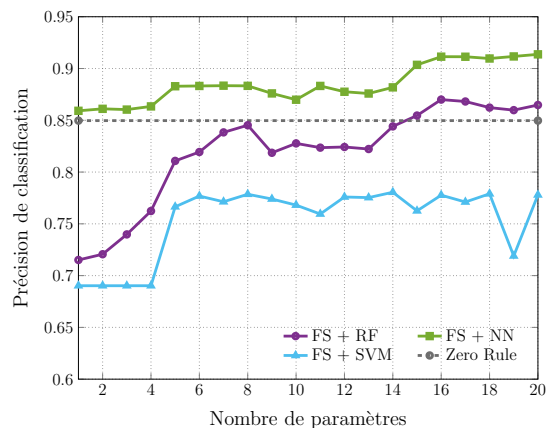
TABLE 2.3 – Avantages et inconvénients des différentes méthodes de sélection de paramètres

	Méthodes par filtrage	Méthodes par encapsulation
<b>Avantage(s)</b>	<ul style="list-style-type: none"> <li>• Simples et rapides</li> </ul>	<ul style="list-style-type: none"> <li>• Solutions adaptées à un type de classifieur</li> <li>• Solutions avec de meilleures performances de classification</li> <li>• Tiennent compte des interactions entre les paramètres</li> </ul>
<b>Inconvénient(s)</b>	<ul style="list-style-type: none"> <li>• Solutions avec de plus faibles performances de classification</li> </ul>	<ul style="list-style-type: none"> <li>• Extrêmement coûteuses quand le nombre de paramètres devient trop grand</li> </ul>

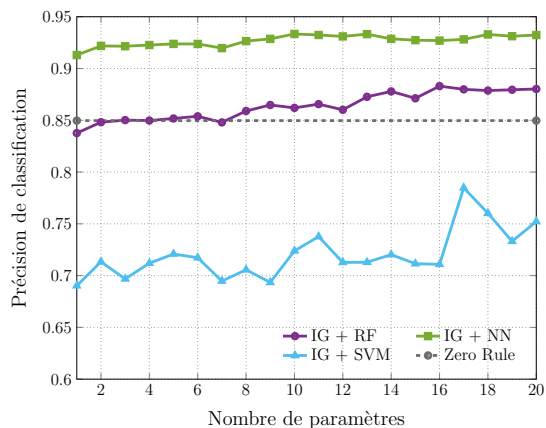
### 2.3.1 Comparaison des méthodes de sélection et de différents types d'ondelettes

Nous allons, dans un premier temps, nous intéresser aux performances de classification obtenues grâce à l'utilisation de méthodes de sélection comme celles présentées précédemment. Une étude similaire a été réalisée dans [8], dans cette dernière le score de Fisher et le critère d'importance de forêt sont testés et évalués avec des réseaux de neurones (NN) et des forêts aléatoires (RF). Nous avons ici décidé d'explorer de nouvelles pistes avec d'autres algorithmes de sélection. Les méthodes ainsi utilisées sont celles du Gain d'Information (IG), du score de Fisher (FS) et de la recherche séquentielle (SFS). Les paramètres sélectionnés sont également testés avec plus de modèles de classification puisque les réseaux de neurones (NN), les forêts aléatoires (RF) et les machines à vecteurs de support (SVM) sont utilisées ici.

La Figure 2.13 montre les performances de classification obtenues avec les ondelettes de Haar en fonction du nombre de paramètres sélectionnés grâce aux méthodes par filtrage IG et FS et en utilisant les différents modèles de classification précédemment évoqués. Ces trois méthodes de classification ont d'abord été choisies, car elles présentent deux avantages : elles sont facilement réalisables sous MATLAB<sup>®</sup> et restent relativement simple pour être implémenté au niveau d'un capteur. Les réseaux de neurones appris sont des réseaux de neurones à propagation avant, en anglais *Feedforward neural network*, avec une couche cachée de dix neurones et les forêts sont composées de cinquante arbres. Nous avons choisi d'utiliser des SVM linéaires afin de limiter la complexité, en effet, l'apprentissage de ce type de classifieur n'est pas adapté pour des ensembles de données de grande taille. L'utilisation de noyaux plus complexes augmenterait d'autant plus la complexité. Un classifieur Zero Rule (ZR) est utilisé comme référence. ZR attribue à chaque exemple testé la classe la plus fréquente dans l'ensemble de données d'entraînement.



(a) Précision de classification avec sélection par score de Fisher



(b) Précision de classification avec sélection par gain d'information

FIGURE 2.13 – Précision de classification des méthodes par filtrage en utilisant les ondelettes de Haar

Dans notre ensemble de données de test, environ 85% des données étaient normales, donc ZR a raison 85% du temps. On peut noter que les SVM n'atteignent pas les performances ZR, que les NN permettent d'obtenir les meilleures performances et que le IG semble offrir de meilleures performances que le FS. Par la suite, toutes ces méthodes ont été testées avec des ondelettes de Gabor et ont fourni des résultats similaires. Comme les SVM ne permettent pas d'atteindre les performances du classificateur ZR, ils ne seront pas retenus pour la suite des études. Les méthodes par encapsulation sont extrêmement coûteuses à exécuter pour les données avec un grand nombre de paramètres, afin de réduire le temps de sélection de paramètres, la taille du jeu d'ondelettes a été réduite en utilisant une méthode par filtrage, passant, dans le cas des ondelettes de Haar, de 502 ondelettes à 100 ondelettes.

La Figure 2.14 montre le gain de performances fourni par la combinaison de l'algorithme SFS avec des méthodes par filtrage. Les performances de classification obtenues avec IG et NN, qui étaient les meilleures performances obtenues en utilisant simplement une méthode par filtrage, sont utilisées comme référence dans la Figure 2.14. Dans tous les cas, l'ajout de l'algorithme SFS permet d'améliorer la précision. En utilisant seulement une sélection par filtrage, la précision maximale obtenue était de 93.3% en utilisant 10 paramètres. Avec l'ajout du SFS, la précision maximale des quatre combinaisons est comprise entre 97.3% et 98.4%, pour un nombre de paramètres compris entre 6 et 10. Cela représente une augmentation d'au moins de 4% de précision en sélectionnant autant, voire moins de paramètres. Nous pouvons noter que les meilleures performances sont également obtenues avec les NN, nous obtenons des performances de classification supérieures à 98% avec au moins six paramètres en utilisant l'algorithme SFS et les NN, quel que soit le filtrage effectué. Cette combinaison a donc été testée sur des ondelettes de Gabor et comparée aux ondelettes de Haar, en utilisant également une étape de pré-filtrage réduisant le nombre de paramètres à cent.

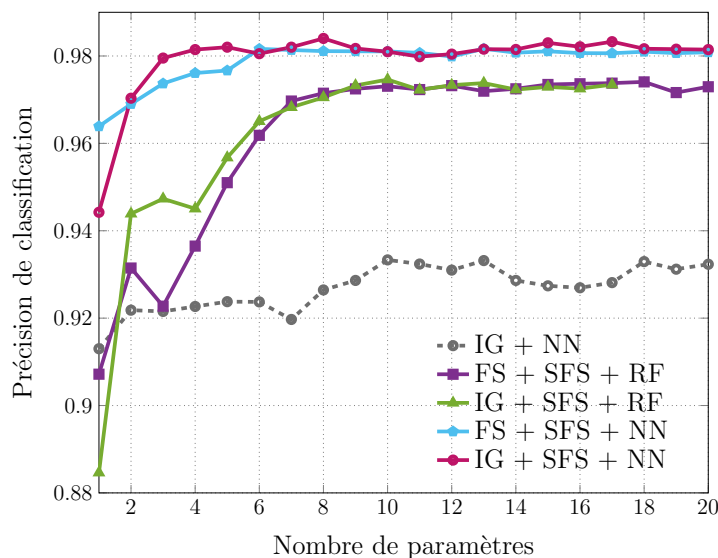


FIGURE 2.14 – Influence de l'algorithme SFS sur les performances de classification en utilisant les ondelettes de Haar

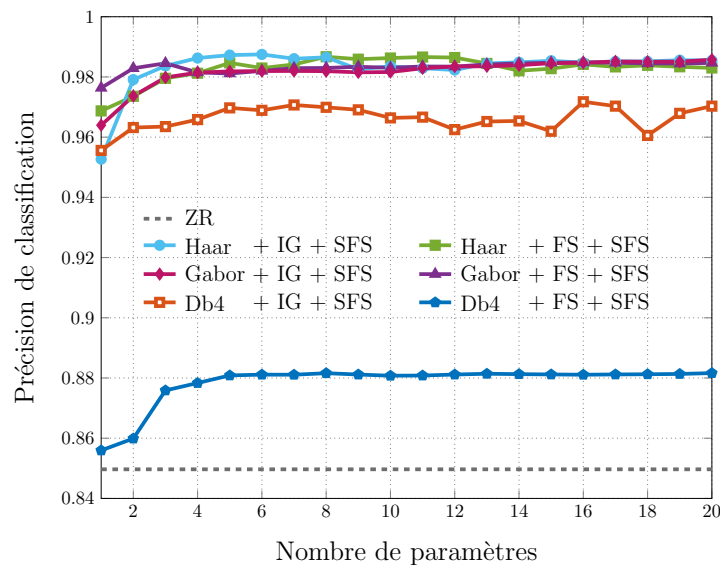


FIGURE 2.15 – Comparaison des performances obtenues avec les ondelettes de Gabor, de Haar et de Daubechies

La Figure 2.15 permet de comparer les performances de classification obtenues avec les ondelettes de Haar, de Gabor et de Daubechies en fonction du nombre de paramètres sélectionnés par l'algorithme de recherche séquentielle utilisant des réseaux de neurones, après une présélection via une méthode par filtrage. Cette étude a été réalisée dans [8], mais il semble pertinent de la reconduire du fait de l'utilisation de nouvelles méthodes de sélection. De plus, les ondelettes de Daubechies sont également testées ici. La précision de classification obtenue avec les familles d'ondelettes de Haar et Gabor est similaire, la précision maximale atteinte pour ces deux familles d'ondelettes est d'environ 98.7% pour les ondelettes de Haar (avec sept paramètres) et 98.5% pour les ondelettes de Gabor (avec dix-sept paramètres), tandis que les ondelettes de Daubechies permettent d'obtenir une précision maximale de 97.2% pour seize paramètres sélectionnés. L'inconvénient de la décomposition en ondelettes de Daubechies est qu'elle est obtenue par un processus récursif, qui ne correspond pas à l'architecture d'extraction de caractéristiques analogiques mentionnée précédemment. Étant donné que les performances obtenues avec les ondelettes de Haar et de Gabor sont similaires et que les ondelettes de Haar sont de simples signaux carrés, ce qui permet de les générer facilement de manière numérique et de faciliter l'étape de mixage, elles semblent donc être un meilleur choix pour notre application.

### 2.3.2 Synchronisation du complexe QRS

Jusqu'à présent, toutes les simulations ont été réalisées en considérant que le complexe QRS est centré autour du centième échantillon de la fenêtre d'analyse, comme expliqué dans la Section 2.1.3.2 et illustré dans la Figure 2.5. Ceci a pour but de connaître et de localiser, pour chaque battement annoté, la totalité de l'information contenue dans les différentes ondes

(P, Q, R, S et T) du complexe. Du fait de la très courte période de certaines ondelettes, une variation de la position du complexe QRS aurait pour conséquence la modification de l'information acquise par ces ondelettes, et donc potentiellement une perte de performances de classification plus ou moins importante. L'effet d'une désynchronisation est donc évaluée ici.

Pour cela, les blocs de 256 échantillons utilisés précédemment sont réutilisés en introduisant un décalage temporel du complexe QRS. Pour chaque bloc, le décalage est choisi aléatoirement suivant une loi normale de moyenne  $\mu = 0$ , c'est-à-dire centré sur le centième échantillon de la fenêtre, et d'écart-type  $\sigma$ . L'estimation de l'effet de la désynchronisation réalisée dans [8], utilise les résultats de sélection obtenus avec le score de Fisher. Dans notre cas, la sélection des paramètres est réalisée, comme précédemment, à l'aide d'une recherche séquentielle de paramètres après une présélection basée sur le gain d'information. De plus, notre étude est réalisée pour des écarts de désynchronisation beaucoup plus grands et bien plus réalistes, comme nous le verrons plus tard. En effet, l'étude réalisée ici explore des écarts-type compris entre  $\sigma = 20$  et  $\sigma = 50$  échantillons (soit entre 55 et 139 ms) alors que l'étude de [8], ne considère que des variances comprises entre  $\sigma^2 = 0$  et  $\sigma^2 = 10$ . Les résultats de cette recherche sont présentés à la Figure 2.16. Cette Figure montre les effets d'une désynchronisation du complexe QRS, pour quatre valeurs différentes d'écart-type  $\sigma$ , sur la précision de classification, dans le cas où l'apprentissage est réalisé sur des données parfaitement synchronisées.

On constate que lorsque les données de l'ensemble d'apprentissage sont synchronisées, la précision chute considérablement, en effet la meilleure précision obtenue est de 92.4% alors que les meilleures performances obtenues précédemment étaient supérieures à 98%. On constate également que, globalement, les performances se dégradent au fur et à mesure que l'on ajoute des paramètres. Lorsqu'un décalage aléatoire du même type est ajouté aux données d'entraî-

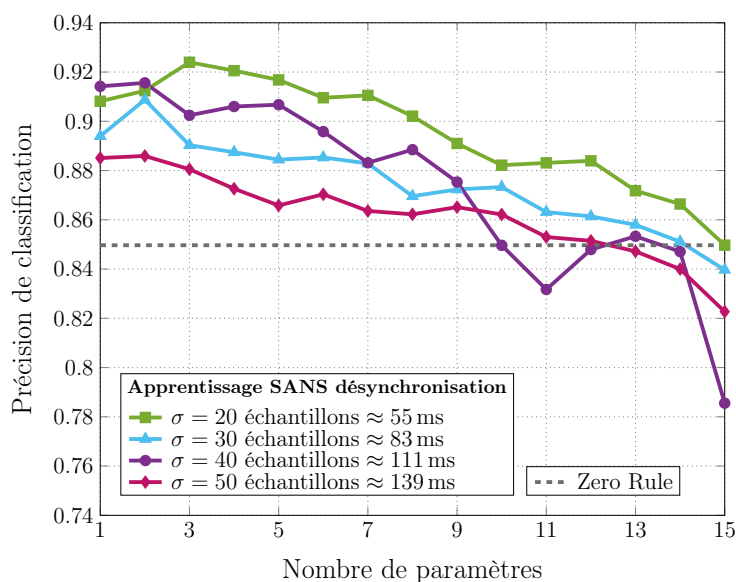


FIGURE 2.16 – Effet de la désynchronisation du complexe QRS sur les performances de classification



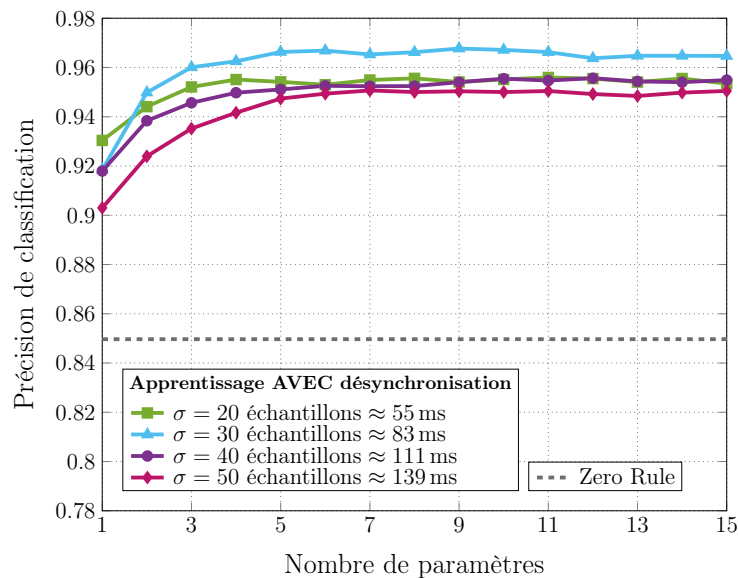
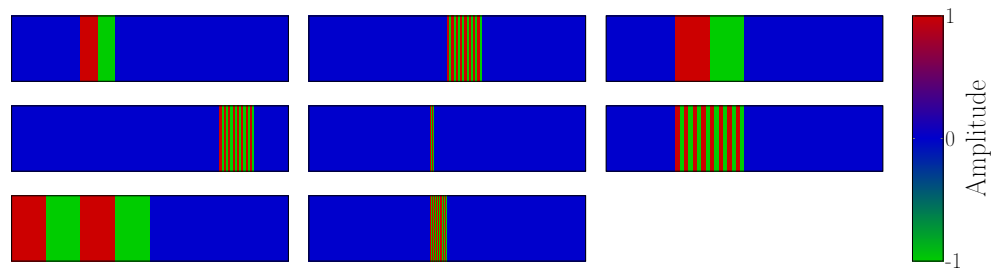


FIGURE 2.17 – Précision de classification lorsque les données d’entraînement sont désynchronisées

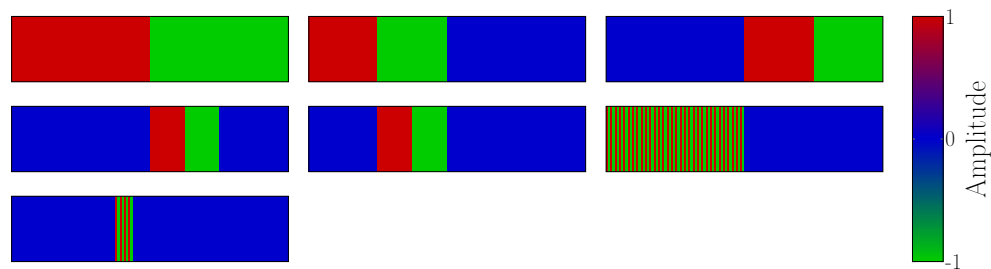
nement, on constate que cela permet de corriger en partie cette perte de performances, comme le montre la Figure 2.17. La précision maximale ainsi obtenue se situe entre 95% et 96.7% de bonnes classifications, ce qui reste tout de même limité lorsque l’on considère que celle-ci était supérieure à 98% dans lorsque les données sont parfaitement synchronisées.

Du fait de la localisation temporelle des ondelettes et de la très courte période que celles-ci peuvent avoir, la désynchronisation du complexe QRS a donc pour conséquence une perte de performances. La Figure 2.18 compare les ondelettes sélectionnées par l’algorithme SFS dans les cas où le complexe QRS est parfaitement centré sur le centième échantillon et où sa position varie suivant une loi normale  $\mathcal{N}(0, 50)$ . Les ondelettes sélectionnées dans le cas idéal sont celles dont les performances sont présentées à la Figure 2.14, en utilisant la combinaison IG, SFS et NN. On constate que lorsque l’on introduit un décalage aléatoire de la position du complexe QRS, alors les ondelettes sélectionnées ont un support temporel bien plus grand que dans le cas idéal où les complexes sont synchronisés. Ce qui peut s’expliquer par le fait que les informations contenues dans les différentes ondes sont alors situées dans une plage temporelle plus étalée.

Bien que cette étape de synchronisation soit relativement complexe, celle-ci est nécessaire, car le système doit pouvoir générer des ondelettes avant l’arrivée d’une onde R. Il est d’abord nécessaire de détecter les complexes QRS successifs. Des circuits pour détecter le complexe QRS ont été proposés dans [112, 113] et [114] dont les caractéristiques sont présentées dans le Tableau 2.4. Les performances de détection de ces circuits sont évaluées en fonction de leur sensibilité (Se) et de leur valeur prédictive positive (+P). Le circuit proposé dans [114] se divise en deux sous-parties : la première analogique et la seconde numérique. La partie analogique, illustrée à la Figure 2.19, se compose tout d’abord d’un bloc dérivateur et d’une fonction



(a) Ondelettes sélectionnées par l'algorithme SFS utilisant des données d'entraînement synchronisées



(b) Ondelettes sélectionnées par l'algorithme SFS utilisant des données d'entraînement désynchronisées

FIGURE 2.18 – Comparaison des ondelettes sélectionnées AVEC et SANS variation de la position du complexe QRS

carrée. Le signal résultant est passé au travers deux filtres passe-bas avec des fréquences de coupure différentes. Le premier, avec une bande passante large permet de réduire le niveau de bruit et de réduire les "pics" provoqué par les ondes T de grande amplitude. Le second filtre possède une bande passante plus petite, il fournit une estimation de l'énergie moyenne du signal. Une version amplifiée de ce signal est utilisée comme seuil de détection des complexes QRS. De fausses détections positives de complexes QRS peuvent être induites par du bruit ou des ondes T de grandes amplitudes. La partie numérique du circuit permet ainsi de définir un intervalle de temps minimum entre deux ondes R.

Deux approches peuvent alors être envisagée pour réaliser la synchronisation : détecter le complexe QRS et réaliser, en parallèle, l'extraction des paramètres sur un signal suffisamment retardé pour pouvoir déclencher la génération des ondelettes à l'instant adéquat. Cette solu-

TABLE 2.4 – Caractéristiques de différents circuits pour la détection de complexes QRS.

Réf.	Technologie ( $\mu\text{m}$ )	Tension (V)	Puissance (nW)	Surface ( $\text{mm}^2$ )	Se (%)	+P (%)
[112]	0.13	0.3	220	0.36	97.76	98.59
[113]	0.13	1.2	622	0.136	98.89	99.4
[114]	0.13	1.2	31.5	0.097	99.17	99.36

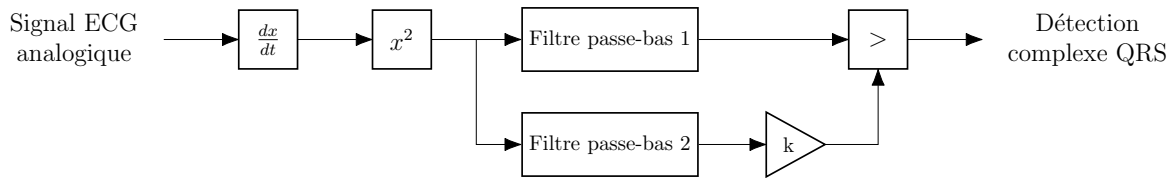


FIGURE 2.19 – Schéma du circuit analogique de détection de complexe QRS

tion permettrait une synchronisation précise, mais elle est difficilement envisageable pour des retards aussi longs, en effet le retard minimum à implémenter est d'environ 277 ms, ce qui correspond au décalage de cent échantillons entre le début de la fenêtre d'analyse et l'arrivée de l'onde R. La deuxième solution est de détecter les complexes successifs et d'utiliser un modèle afin de prédire l'instant de la prochaine onde R. Réaliser un modèle de prédiction de l'intervalle R-R (voir Figure 2.20) constitue en soit un problème de détection d'arythmie.

Différentes approches ont été testées pour créer un modèle de prédiction de l'intervalle R-R : deux approches naïves, l'une réalisant la moyenne des N précédents intervalles, l'autre donnant la médiane, et une approche basée sur l'apprentissage d'un réseau de neurones récurrents (Recurrent Neural Network, RNN). Les réseaux de neurones récurrents sont des réseaux de neurones artificiels présentant des connexions récurrentes. Ils sont particulièrement adaptés pour l'analyse de séries temporelles et sont notamment utilisés pour des tâches comme la reconnaissance de la parole ou de l'écriture manuscrite. Il existe différents types de RNN, nous utiliserons ici un réseau neuronal auto-régressif non linéaire (Nonlinear AutoRegressive, NAR)[115]. Le réseau NAR est un réseau récurrent dont la sortie à un instant donné dépend de ses valeurs passées. Il peut soit être utilisé en boucle ouverte ou fermée. Dans notre cas, il est utilisé en boucle ouverte puisque les intervalles R-R sont mesurés grâce à un système dédié.

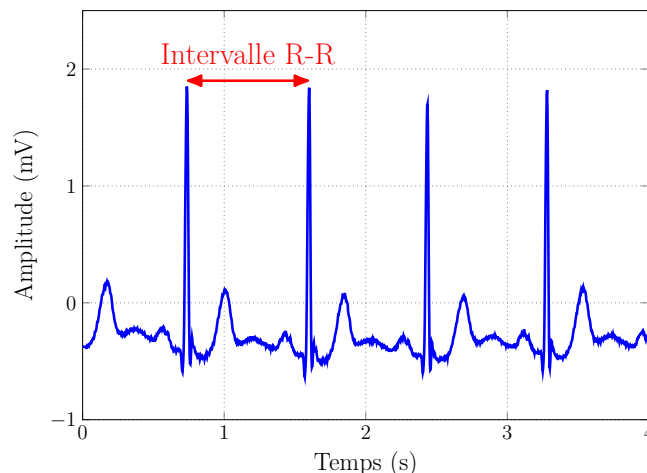


FIGURE 2.20 – Définition de l'intervalle R-R

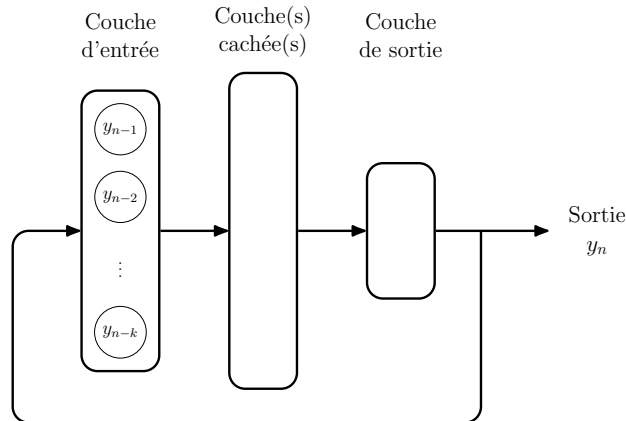


FIGURE 2.21 – Schéma d'un réseau de neurones récurrents

La Figure 2.21 représente un schéma simplifié d'un NAR. Le réseau NAR peut être appliqué à la prévision efficace de séries chronologiques et peut s'écrire comme suit :

$$\hat{y}_n = f(y(n-1), y(n-2), \dots, y(n-k)) + \varepsilon(n) \quad (2.21)$$

où  $f$  est une fonction non linéaire qui peut être approchée lors de la phase d'entraînement du réseau,  $\hat{y}_n$  est la valeur prédite de la série de données  $y$  à un instant  $n$ ,  $k$  représente le nombre de valeurs passées de la série utilisées pour la prédiction et  $\varepsilon(n)$  est l'erreur à l'instant  $n$ . La couche d'entrée d'un tel réseau est construite de manière que si l'on souhaite déterminer la valeur de sortie grâce aux  $k$  précédentes valeurs alors la couche d'entrée possède  $k$  neurones.

Le Tableau 2.5 présente les résultats obtenus grâce aux trois approches de prédiction de l'intervalle R-R. Le prédicteur ayant la meilleure précision est celui basé sur les réseaux NAR utilisant la position des 10 derniers complexes, l'écart type obtenu est de 148 ms ce qui correspond environ à 53 échantillons dans la base de données. La Figure 2.17 nous montre que la meilleure précision de classification obtenue pour une valeur de  $\sigma = 50$  est de 95%. On

TABLE 2.5 – Comparaison de l'erreur de prédiction des intervalles R-R pour les trois approches testées

		Valeur moyenne	Valeur médiane	RNN
N = 5	Erreur Moyenne (ms)	-0.34	3.1	0.22
	Écart Type (ms)	172	176	152
N = 10	Erreur Moyenne (ms)	-0.05	2.7	-0.22
	Écart Type (ms)	165	175	148

peut maintenant se demander si une telle précision est suffisante pour cette application. Si ce n'est pas le cas, il sera alors nécessaire de trouver un système de prédiction plus performant ou d'utiliser un modèle de classification plus robuste.

### 2.3.3 Niveau de quantification

Après l'extraction analogique des paramètres, ces derniers sont numérisés par le convertisseur A2F, via l'utilisation d'un CAN. La fréquence d'échantillonnage du CAN n'est pas régulière et dépend des ondelettes utilisées pour l'extraction. Étant donné la fréquence minimum de celles-ci, la fréquence d'échantillonnage ne peut pas être supérieure à 180 Hz. Dans la Figure 2.14, nous avons pu constater qu'une précision de 98% pouvait être atteinte en utilisant seulement six paramètres pour une période de signal ECG, alors que 256 échantillons étaient acquis un échantillonnage uniforme au taux de Nyquist (360 Hz). La fréquence d'échantillonnage moyenne ainsi obtenue est d'environ 8.5 Hz et le taux de compression est d'environ 97.5%. Cette réduction du nombre d'échantillons permet ainsi d'utiliser de manière plus parcimonieuse l'étage de transmission. Pour réduire la quantité de données à transmettre, on peut également réduire la taille d'un échantillon. Jusqu'à présent, le niveau de quantification du CAN n'était pas pris en compte lors des simulations, ces dernières étant réalisées avec des données codées sur 64 bits (virgule flottante double précision). Une nouvelle évaluation du niveau de quantification nécessaire des paramètres est donc réalisée. Ce qui la différencie de celle menée dans [8], réside dans le fait que la sélection des paramètres basée sur le SFS est réalisée en utilisant les données quantifiées. Cela permet ainsi de prendre en compte d'éventuelles pertes d'information, engendrée par le processus de quantification. Certains paramètres sont susceptibles de ne plus fournir d'informations pertinentes pour la classification.

L'algorithme SFS est appliqué sur les données quantifiées après avoir réalisé une présélection par Gain d'information, cela signifie que les données d'apprentissage et de test du réseau de neurones sont dans les deux cas, quantifiées. Les résultats de cette sélection sont présentés à la Figure 2.22. On peut constater une forte baisse de performances pour des niveaux de quantification en dessous de six bits. En comparaison, les performances obtenues pour des niveaux de quantification sur six, huit et dix bits sont relativement proches, ce qui signifie qu'une précision de six bits est suffisante pour l'application considérée.

On constate également que la précision obtenue pour une quantification sur six ou huit bits peut être légèrement meilleure que lorsque l'on utilise une base de données non quantifiées. Ceci peut être expliqué par un phénomène de sur-apprentissage du classifieur lorsque celui-ci utilise des données non quantifiées ou alors par la variabilité des résultats obtenus due à l'initialisation des réseaux de neurones. En effet, ces dernières peuvent contenir du bruit, similaire pour tous les signaux, que le classifieur tente donc d'apprendre. Il n'apprend donc pas uniquement les spécificités des signaux des différentes classes, ce qui nuit ainsi aux performances de classification. Le processus de quantification induit une perte de d'information, l'effet obtenu est similaire à celui d'un filtre.

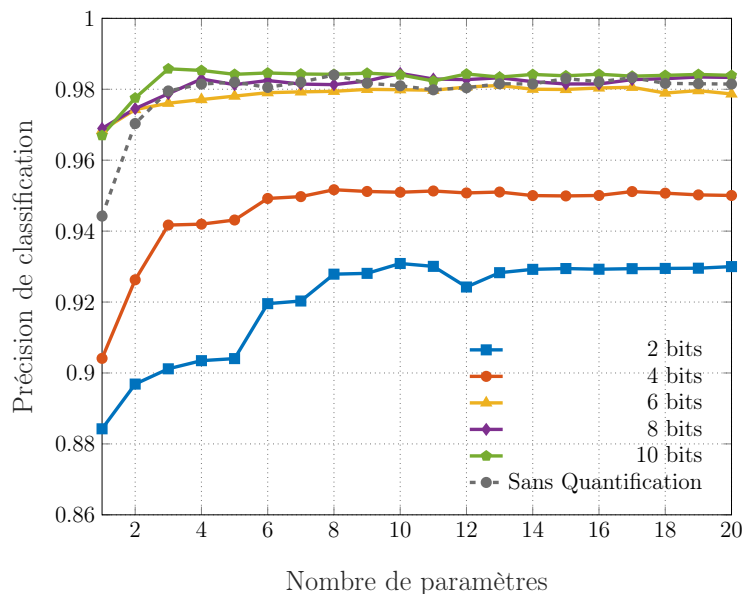


FIGURE 2.22 – Performances obtenues avec la recherche séquentielle pour différents niveaux de quantification

### 2.3.4 Reconnaissance d'anomalie

L'architecture de convertisseur A2F, présenté à la Figure 2.2, utilise un détecteur de contexte qui permet de reconfigurer le convertisseur et ainsi activer, désactiver ou modifier le jeu de paramètres extraits par chaque extracteur. Ce système permet de rendre le convertisseur A2F flexible et ainsi utilisable pour différentes applications. Cela permet également, pour une application donnée, de modifier la précision du convertisseur : on peut en effet décider d'extraire peu de paramètres afin de réaliser une classification binaire, ou alors extraire plus de paramètres afin de pouvoir détecter le type d'évènement. Nous allons ici réaliser une étude préliminaire portant sur l'amélioration de la précision du convertisseur A2F dans le contexte des signaux ECG afin de détecter le type d'anomalie.

Jusqu'à présent, la classification réalisée était une classification binaire permettant de détecter la présence ou non d'une anomalie. Nous souhaitons maintenant pouvoir détecter différents types d'anomalies. La base de données des signaux ECG MIT-BIH contient, outre les battements normaux, 19 classes différentes pour catégoriser les différents complexes ECG. Parmi ces classes, seul les cinq ayant le plus d'exemples ont été conservées :

- Bloc de branche gauche (L)
- Bloc de branche droit (R)
- Contraction auriculaire prématurée (A)
- Contraction ventriculaire prématurée (V)
- Battement "rythmé" (/)

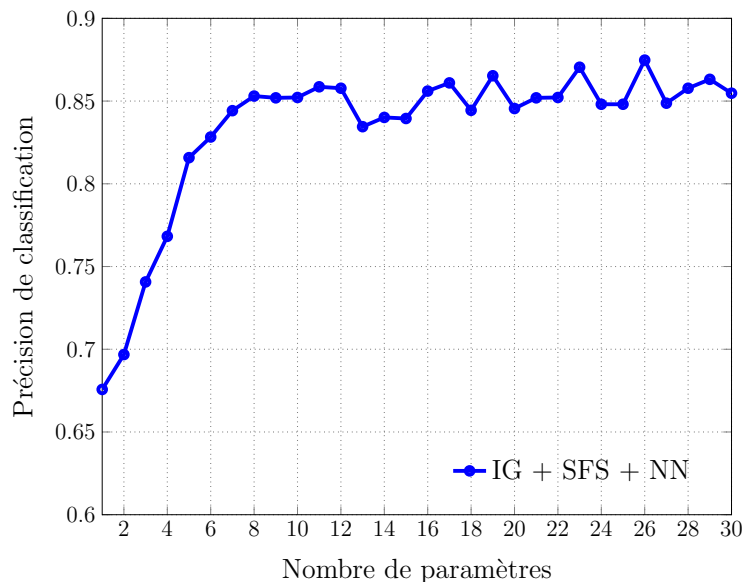


FIGURE 2.23 – Précision de classification obtenue lors de la reconnaissance du type d'anomalie sur des données synchronisées et non quantifiées

Ces cinq classes ont été regroupées en deux catégories : la première contenant les exemples des classes "Bloc de branche gauche" et "Bloc de branche droit", caractérisant un défaut de "forme" dans le complexe QRS, et une seconde contenant les exemples des trois autres classes, c'est-à-dire : contractions auriculaires et ventriculaires prématurées et battements "rythmés". Les deux ensembles créés sont d'une taille équivalente : soit 15 327 exemples appartenant à la première classe (anomalies L et R) et 16 699 exemples appartenant à la seconde classe. La tâche à réaliser ici est donc toujours binaire, mais l'objectif est de distinguer deux catégories d'anomalies. La Figure 2.23 montre les performances de classification obtenues pour cette tâche en effectuant une sélection de paramètres, sur des données synchronisées et non quantifiées, combinant l'IG et le SFS et en utilisant des réseaux de neurones. On constate que l'utilisation d'un réseau de neurones, pour la détection de la catégorie d'une anomalie, permet d'obtenir une précision de 85% en utilisant huit paramètres et que la précision maximale obtenue est de 87.5% avec 26 paramètres. Ces chiffres peuvent être comparés aux performances d'un classifieur ZR qui aurait une précision de 52%.

## 2.4 Conclusion

Ce Chapitre présente une nouvelle méthode d'acquisition des signaux, appelée conversion analogique-paramètre (A2F), dont le but est de diminuer la quantité de données acquise par les capteurs sans-fil en extrayant, de manière analogique, des informations utiles du signal pour la réalisation d'une classification (ou d'une régression). Cette méthode permet de réduire la consommation des capteurs en permettant une utilisation plus parcimonieuse du système de communication RF. La technique de l'échantillonnage non-uniforme en ondelettes (NUWS),

issue du CS, est proposée comme méthode d'extraction de paramètres. Le NUWS possède plusieurs degrés de liberté, ce qui offre une certaine flexibilité et permet ainsi d'obtenir un convertisseur A2F générique pouvant acquérir différents types de signaux à basses fréquences (ECG, EEG, EMG, voix, ...).

Les différents degrés de liberté du NUWS imposent l'utilisation d'algorithmes de sélection de paramètres afin d'identifier un ensemble réduit de paramètres pertinents pour l'étape de classification. Dans le contexte de la détection d'anomalies dans des signaux ECG, différentes méthodes (par filtrage ou par encapsulation) ont été combinées afin de tirer parti des avantages de chacune. L'utilisation de tels algorithmes permet de réduire l'ensemble de recherche à sept paramètres et d'obtenir une précision de classification de 98.7% avec des ondelettes de Haar. De nombreuses pistes sont étudiées afin de minimiser la quantité de données et maximiser les performances de détection. Une étude sur le choix du type d'ondelettes a permis de constater les performances relativement proches des ondelettes de Gabor et de Haar, permettant ainsi de conclure que ces dernières sont un choix judicieux étant donné leur simplicité. Une évaluation du niveau de quantification nécessaire permet de montrer qu'une précision de 6 bits est suffisante pour obtenir une bonne précision de classification.

L'exemple des signaux ECG apporte un problème spécifique concernant la synchronisation de la génération des ondelettes avec l'apparition d'un complexe QRS. En effet, cette étape est nécessaire du fait de la localisation temporelle réduite de certaines ondes du complexe QRS et, car les informations contenues dans les différentes ondes doivent être extraites au cours de la même "fenêtre" d'analyse. Plusieurs modèles relativement simples sont proposés afin de prédire la position du prochain complexe QRS, mais les performances de ces modèles restent limitées : un modèle basé sur un réseau de neurones NAR a permis d'obtenir une variance de la position d'environ 148 ms. Ce décalage entraîne une perte de précision qui peut être limitée si l'apprentissage est réalisé avec des données désynchronisées. Étant donné la précision de la prédiction, la précision de classification chute à 95%. L'amélioration de la prédiction du prochain complexe QRS peut alors constituer une bonne piste d'exploration. Globalement, l'étude menée dans ce Chapitre a permis de réaliser une sélection affinée des paramètres, par rapport aux résultats proposés dans [8], permettant ainsi d'obtenir un bien meilleur ratio précision/nombre de paramètres. L'étude menée sur les effets de la désynchronisation du complexe QRS est largement améliorée grâce à la réalisation d'un modèle de prédiction de l'intervalle R-R, permettant d'étudier une plage de variation beaucoup plus importante et plus réaliste.

En résumé, la méthodologie appliquée dans ce Chapitre pour le dimensionnement d'un convertisseur A2F, permettant de réduire au maximum le nombre d'échantillons tout en fournissant de bonnes performances de classification, est la suivante :

1. Choisir la taille de la fenêtre d'analyse, afin de découper le signal en plusieurs vecteurs de taille  $N$ . Dans le cas des signaux ECG, la valeur de  $N$  est de 256 échantillons,



chaque vecteur correspond à un battement annoté et l'onde R est centrée sur le centième échantillon.

2. Constituer différents dictionnaires d'ondelettes, en considérant la dimension des vecteurs précédemment constitués.
3. Extraire les paramètres des différents vecteurs grâce aux dictionnaires réalisés.
4. Réaliser une présélection des meilleurs paramètres en utilisant une méthode par filtrage.
5. Sélectionner, parmi les paramètres restants, ceux fournissant les meilleures performances grâce à une méthode par encapsulation.
6. Tester différents modèles de classification afin de déterminer celui qui est le plus adapté à l'application choisie.
7. Déterminer la précision du CAN en évaluant l'influence du niveau de quantification sur la précision de classification.
8. Considérer les potentiels sources d'erreur : dans le cas des signaux ECG, un défaut de synchronisation peut constituer une source d'erreur de classification.

Afin d'optimiser au mieux la consommation d'énergie, il est nécessaire de prendre en considération l'architecture matérielle lors de la sélection des paramètres. La précision du CAN nécessaire est une piste qui est explorée. Dans le Chapitre suivant, une adaptation de l'algorithme SFS est proposée afin de limiter le nombre d'extracteurs à utiliser en mutualisant, lorsque cela est possible, l'acquisition de plusieurs paramètres avec un seul extracteur. Une étape supplémentaire dans la réduction de la consommation consiste à prendre en compte le coût énergétique de l'extraction de chaque paramètre lors de la sélection.



## Chapitre 3

# Évaluation et optimisation de la consommation d'un convertisseur analogique-paramètre

### Introduction

Nous avons d'abord mis en évidence le coût énergétique du système RF dans la consommation des capteurs sans-fil, et donc le gain énergétique que peut apporter une utilisation parcimonieuse de celui-ci grâce à une diminution de la quantité de données. Dans le Chapitre 2, les convertisseurs A2F sont proposés comme alternatives aux convertisseurs A2I basés sur le CS, dont le taux de compression est limité, pour les applications utilisant des modèles basés sur l'apprentissage automatique. En utilisant des ondelettes de Haar et des données synchronisées, une précision de classification de 98.7% peut être atteinte en extrayant sept paramètres du signal ECG.

Afin de montrer l'intérêt des convertisseurs A2F par rapport aux approches classiques et aux convertisseurs A2I, et d'adapter au mieux leur consommation d'énergie, un modèle de consommation doit être réalisé. Il est donc nécessaire de déterminer la consommation des différentes parties du convertisseur. Ce modèle de consommation réalisé dans ce Chapitre est obtenu à partir de données provenant de l'état de l'art et de simulations, dans une technologie CMOS 0.18  $\mu\text{m}$ . Il permettra ainsi de déterminer le coût d'extraction de chaque paramètre.

Pour optimiser la consommation du convertisseur A2F, il est intéressant de prendre en compte l'architecture du convertisseur A2F, ainsi que le coût d'extraction des paramètres lors de la sélection de ces derniers en plus de la précision de classification. Un nouvel algorithme de sélection, basé sur l'algorithme SFS, adapté à l'architecture et un nouveau critère d'évaluation permettant de prendre en compte le coût d'extraction de chaque paramètre, sont proposés, l'objectif étant de minimiser le coût global, tout en maximisant la précision lors de la détection des anomalies dans le signal ECG.

### 3.1 Sélection adaptée à l'architecture

Afin de réduire la consommation énergétique du système, nous souhaitons minimiser le nombre d'extracteurs de paramètres de l'architecture du convertisseur A2F en les utilisant pour extraire plusieurs paramètres dans la fenêtre de  $N = 256$  échantillons. Pour cela, il est nécessaire que les différents paramètres extraits utilisent des ondelettes qui ne se superposent pas temporellement, comme illustré à la Figure 3.1. L'algorithme SFS a donc été adapté dans ce sens, comme le montre l'Algorithme 2.

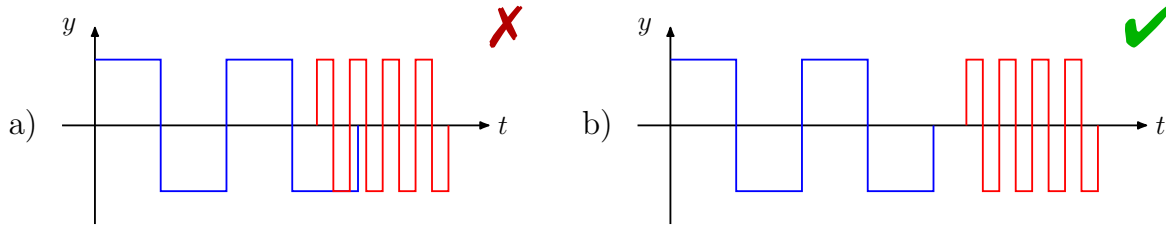


FIGURE 3.1 – Exemples de a) deux ondelettes superposées qui ne peuvent pas être acquises par le même extracteur et b) deux ondelettes non superposées qui peuvent être acquises par le même extracteur

Ce concept de mutualisation de l'extraction de paramètres est évoqué dans [8], mais n'est réalisée que pour l'utilisation d'un unique extracteur. Nous avons ici généralisé ce concept en définissant en amont de la sélection un nombre d'extracteurs maximum. La nouvelle version de l'algorithme SFS prend à présent comme paramètre d'entrée le nombre maximum d'extracteurs ( $N_O$ ) que l'on souhaite obtenir. À l'étape  $i$ , après avoir évalué toutes les combinaisons possibles, l'algorithme classe les paramètres par ordre décroissant ( $M_i$ ). À partir de

---

#### Algorithm 2 SFS pour limiter le nombre d'extracteurs

---

- 1: **Input:**  $N_O$  ▷  $N_O$  est le nombre d'extracteurs
  - 2:  $S \leftarrow \emptyset$
  - 3: **repeat**
  - 4:   **for all**  $X_i \notin S$  **do**
  - 5:      $J_i \leftarrow J(S \cup \{X_i\})$
  - 6:    $M \leftarrow \text{sort}(J_i, \text{decreasing})$
  - 7:    $i' \leftarrow \emptyset$
  - 8:   **repeat**
  - 9:     **for all**  $E_s \in \text{Extracteurs}$  **do**
  - 10:      **if**  $(W_i \cap E_s) == \emptyset$  **then** ▷  $W_i$  est l'ondelette associée à  $X_i$  et  $J_i$
  - 11:        $i' \leftarrow M_i$
  - 12:        $S' \leftarrow S \cup \{X_{i'}\}$
  - 13:       **break**
  - 14:   **until**  $i' \neq \emptyset$
  - 15: **until**  $|S| == D$
-

ce classement, il choisit le meilleur paramètre, et donc l'ondelette ( $W_i$ ), avec les meilleures performances. L'algorithme tente alors d'attribuer ce paramètre à un extracteur ( $E_s$ ). Pour cela, il faut que l'ondelette sélectionnée ne soit pas superposée temporellement avec les autres ondelettes attribuées à cet extracteur (comme illustré à la Figure 3.1). Lorsque sur un intervalle de temps donné dans la fenêtre d'analyse, tous les extracteurs génèrent une ondelette, et donc extraient un paramètre, alors cet intervalle de temps devient un intervalle interdit dans le processus de sélection. Les ondelettes restantes dans l'ensemble des solutions possibles et qui sont définies sur cet intervalle de temps sont supprimées des solutions à tester lors des étapes suivantes, car elles ne peuvent être attribuées à aucun extracteur.

La Figure 3.2 présente les résultats obtenus avec cet algorithme. Sur ce graphique les courbes représentent la précision de classification en fonction du nombre de paramètres sélectionnés et chaque paramètre est identifié par un point dont la couleur et la forme varient en fonction de l'extracteur auquel le paramètre est attribué. Comme dans le chapitre précédent, l'ensemble des ondelettes candidates de départ est réduit aux 100 ondelettes ayant les plus hauts gains d'information. Lorsqu'on limite la recherche à trois extracteurs, on constate rapi-

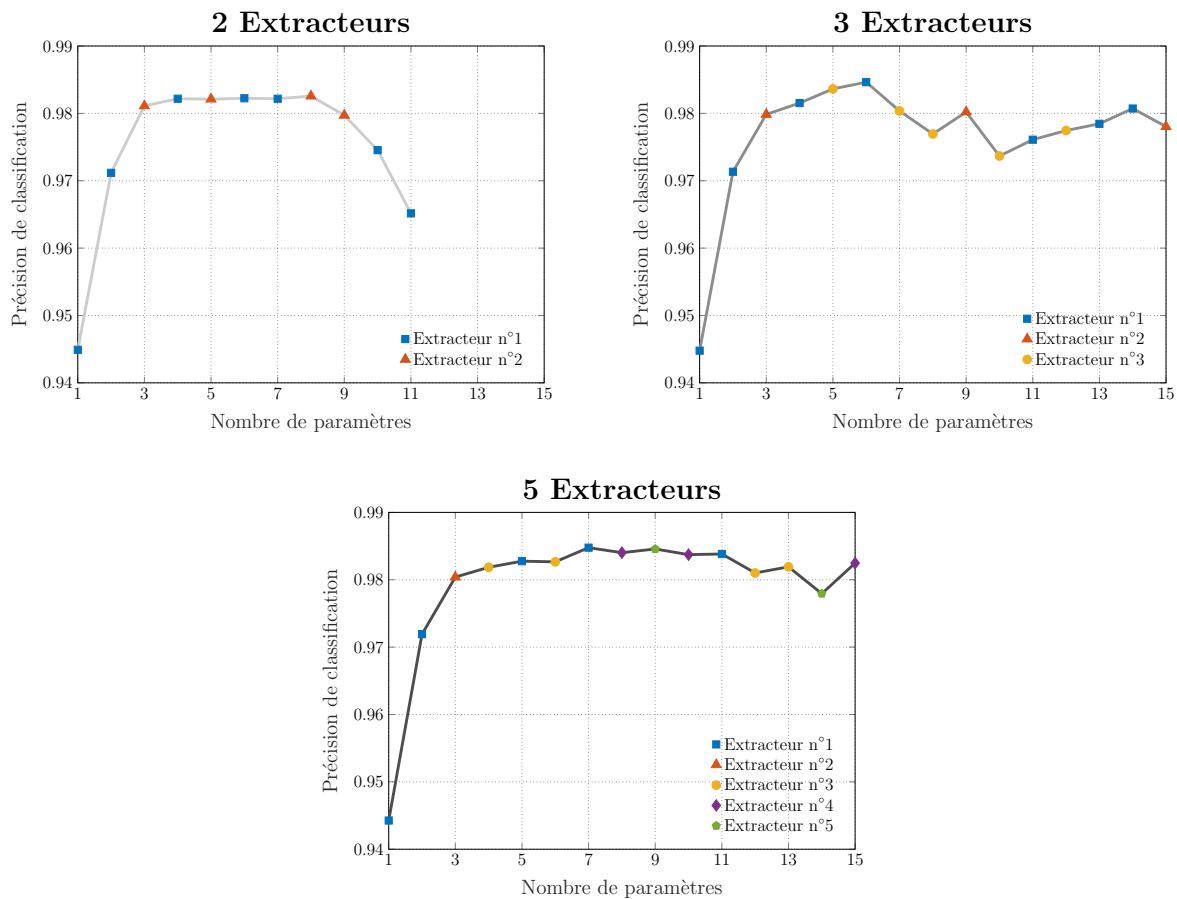


FIGURE 3.2 – Performances obtenues avec la recherche séquentielle optimisée pour 2, 3 et 5 extracteurs

dement un effet de sur-apprentissage et donc qu'après six paramètres sélectionnés, l'ajout de nouveaux paramètres entraîne une dégradation des performances. On constate également un effet de sur apprentissage après huit paramètres sélectionnés lorsqu'on limite à deux extracteurs et que la recherche s'arrête après avoir sélectionné onze paramètres, car il n'y a plus de solution possible. Lorsque la limite du nombre d'extracteurs est fixée à cinq, on remarque que la précision maximale est obtenue alors que seulement trois des cinq extracteurs sont utilisés. Cette précision n'était cependant pas atteinte lorsque la limite était fixée à trois. Ce résultat peut s'expliquer par un choix de paramètres différents à une certaine étape ce qui a influencé le reste de la sélection. On peut ainsi, avec trois extracteurs, faire l'acquisition de sept paramètres et obtenir une précision de classification des anomalies de 98.5%. En limitant le nombre d'extracteurs, les performances de classification obtenues sont équivalentes à celles obtenues à la Figure 2.14, en extrayant seulement un paramètre de plus.

Pour optimiser au mieux le rapport performance/consommation du convertisseur A2F, il est intéressant de tenir compte du coût énergétique de chaque ondelette lors de l'étape de sélection des paramètres, afin de sélectionner des paramètres nécessitant peu de ressources tout en fournissant des informations pertinentes pour la classification. Un modèle de consommation est donc proposé afin d'évaluer ce coût énergétique.

## 3.2 Modèle de consommation du circuit analogique

Le modèle de consommation du circuit analogique est réalisé à partir de données de provenant de l'état de l'art. L'objectif premier dans cette thèse était de réaliser ce modèle de consommation, au niveau transistor, pour la globalité de l'architecture, mais faute de temps et d'expérience, les travaux réalisés dans ce domaine n'ont pas abouti. Comme expliqué dans la Section 2.1, un extracteur de paramètre, illustré à la Figure 3.3, nécessite un circuit frontal, un mixeur et un intégrateur analogique, un convertisseur analogique-numérique, ainsi qu'un générateur d'ondelettes. L'utilisation des ondelettes de Haar pour l'extraction des paramètres permet de simplifier la réalisation de ce dernier avec une architecture numérique. De plus, les signaux ECG ayant un ordre de grandeur de quelques millivolts, un étage d'amplification est nécessaire.

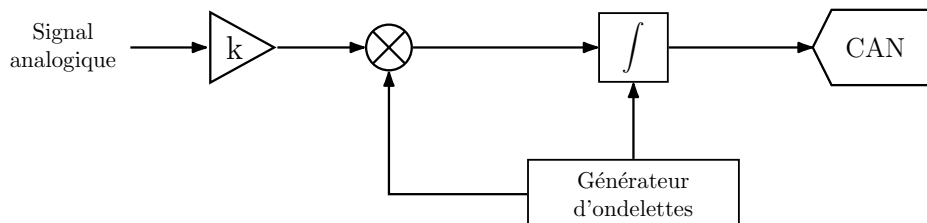


FIGURE 3.3 – Schéma d'un extracteur de paramètre

### 3.2.1 Frontal analogique

Pour implémenter la partie analogique des extracteurs de paramètres, nous proposons d'utiliser la chaîne d'acquisition de signaux biologiques présentée dans [80], et illustrée à la Figure 3.4. Ce circuit est composé d'un amplificateur faible-bruit (Low Noise Amplifier, LNA), d'un amplificateur à gain variable (Programmable Gain Amplifier, PGA), d'un filtre passe-bas (Low Pass Filter, LPF) et d'un CAN.

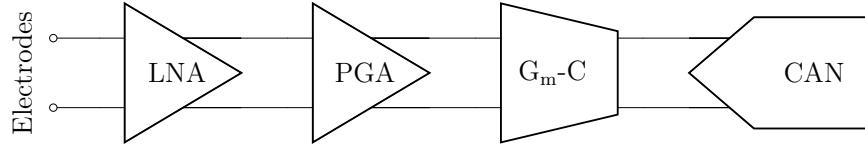


FIGURE 3.4 – Schéma du frontal d'acquisition proposé dans [80]

L'étage d'amplification est donc composé d'un LNA et d'un PGA ayant respectivement un gain de 38 dB (80 V/V) et un gain programmable sur sept niveaux, avec un gain maximal de 34 dB. De manière globale, l'étage d'amplification possède un gain compris entre 38 et 72 dB, est alimenté avec une tension de 1.4 V, consomme un courant de 3.6  $\mu$ A et donc une puissance de  $P_{LNA} + P_{PGA} = 5.04 \mu$ W. L'intérêt d'un tel étage d'amplification est d'obtenir un système flexible et reconfigurable pour traiter différents types de signaux. Le filtre est un filtre passe-bas de type  $G_m$ -C avec une fréquence de coupure paramétrable entre 150 Hz et 10 kHz, son gain maximum, dans les basses fréquences, est de 0 dB. Il est alimenté avec une tension de 1.4 V, consomme un courant de 0.5  $\mu$ A et donc une puissance de 0.7  $\mu$ W. Le CAN utilisé implémente une architecture SAR avec une précision de 10 bits et une fréquence d'échantillonnage maximale de 40 kHz. Il est alimenté avec une tension de 1 V et consomme un courant de 0.3  $\mu$ A. Le rapport signal sur bruit et distorsion, pour une sinusoïde proche de la fréquence de Nyquist, a été mesuré à 56.1 dB, ce qui donne un nombre effectif de bits (ENOB) de 9.02 bits et donc une figure de mérite  $E_{sampling} = 14.3$  fJ/échantillon.

Le Tableau 3.1 résume les caractéristiques des différents éléments du circuit frontal d'acquisition proposé dans [80]. Dans cette étude, les données concernant l'étage d'amplification et le CAN peuvent être utilisées. En revanche, les caractéristiques du filtre passe-bas ne corres-

TABLE 3.1 – Consommation des différents éléments du circuit [80]

LNA et PGA		LPF		CAN	
Tension d'alimentation (V)	1.4	Tension d'alimentation (V)	1.4	Tension d'alimentation (V)	1
Courant ( $\mu$ A)	3.6	Courant ( $\mu$ A)	0.5	Courant ( $\mu$ A)	0.3
Gain (dB)	38 - 72	Gain (dB)	0	SNDR (dB)	56.1
		Fréquence de coupure (Hz)	150 - 10k	ENOB (bits)	9.02

pondent pas aux caractéristiques attendues pour la réalisation d'un intégrateur, dans le cadre du NUWS, comme nous le verrons dans le paragraphe suivant.

### 3.2.2 Mixeur analogique

Le choix des ondelettes de Haar pour l'extraction de paramètres via le NUWS, facilite non seulement la réalisation d'un générateur d'ondelettes, mais également l'étape de mixage entre l'ondelette et le signal entrant. En effet, celles-ci ont pour valeurs -1 ou 1 lorsqu'elles sont définies, ce qui revient donc à intégrer alternativement  $V_{in}$  et  $-V_{in}$ . La chaîne d'acquisition proposée dans [80] est une chaîne différentielle, l'étape de mixage peut ainsi se résumer en l'inversion des signaux positif et négatif à la sortie du PGA. Pour cela, on peut utiliser des interrupteurs pour choisir le signal à connecter sur chacune des entrées. Pour implémenter ces interrupteurs, on peut utiliser des portes de transmission CMOS, dont le schéma et le symbole sont représentés à la Figure 3.5.

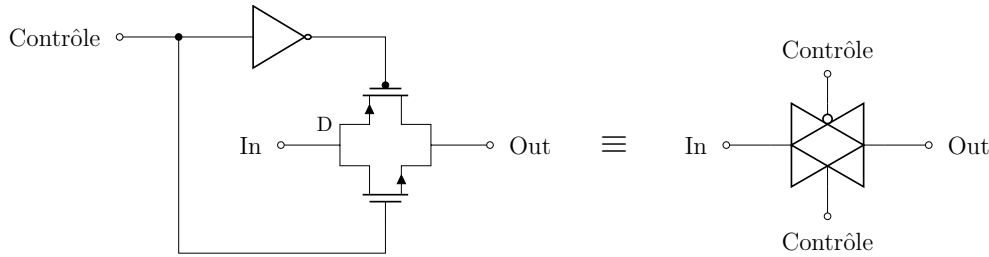


FIGURE 3.5 – Schéma et symbole d'une porte de transmission CMOS

Un seul transistor, NMOS ou PMOS, peut être utilisé comme interrupteur commandé en tension, mais dans les deux cas, le transistor seul ne permet pas de transmettre le signal sans perte. En effet, si la grille d'un transistor NMOS est commandée par un signal  $V_{\text{contrôle}}$  alors le transistor ne peut laisser passer que des signaux dont la valeur est inférieure à  $V_{\text{contrôle}}$  moins la tension de seuil  $V_{\text{seuil}}$ . De manière inverse, la tension  $V_{\text{contrôle}}$  appliquée à la grille d'un transistor PMOS, doit être nettement inférieure à la tension de seuil  $V_{\text{seuil}}$  du transistor pour le désactiver complètement. Ainsi, l'association des transistors NMOS et PMOS d'une porte de transmission CMOS permet la transmission du signal sans dégradation. La Figure 3.6 illustre le mixeur analogique réalisé à partir de portes de transmission CMOS, où C est le signal de contrôle des portes, piloté par le signal provenant du générateur d'ondelettes, et  $\bar{C}$  est son complément.

### 3.2.3 Intégrateur

Après avoir été amplifié et mixé avec une ondelette le signal doit être intégré. La fonction de transfert d'un intégrateur idéal est de la forme  $H(p) = 1/p$ , et sa réponse en fréquence est donc une pente de  $-20$  dB par décade, comme illustré dans la Figure 3.7. Mais dans ce cas, le gain de l'intégrateur tend vers l'infini quand la fréquence tend vers zéro. En pratique, la réponse



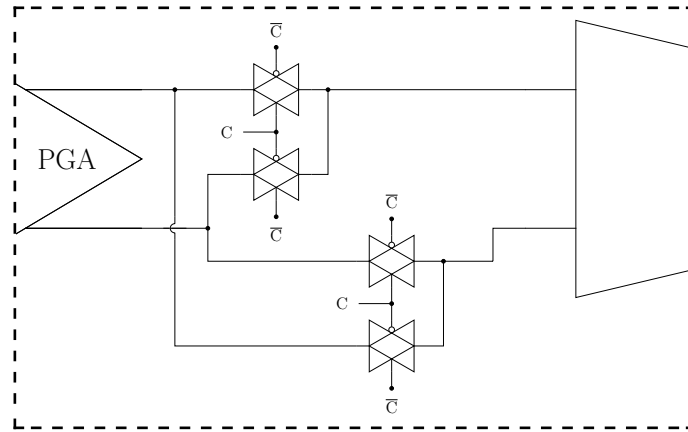


FIGURE 3.6 – Schéma du mixeur analogique

en fréquence d'un intégrateur est celle d'un filtre passe-bas, avec un gain maximum  $G_0$  dans les fréquences basses. Mais pour que celui-ci ait bien un comportement d'intégrateur, il faut que sa fréquence de coupure  $F_c$  soit largement inférieure à la fréquence du signal à intégrer. Étant donné la période maximale d'observation du signal ECG ( $T_{obs} \simeq 0.711$  s), la fréquence de coupure du filtre doit être inférieure à 1.4 Hz pour avoir un comportement intégrateur dans les fréquences désirées.

Le filtre passe-bas proposé dans [80], est un filtre passe-bas de type  $G_m$ -C d'ordre 3, avec une fréquence de coupure paramétrable entre 150 Hz et 10 kHz, qui ne possède donc pas les caractéristiques nécessaires pour notre application. Il est donc nécessaire de proposer un filtre passe-bas ayant une très faible fréquence de coupure. La Figure 3.8 représente trois types de filtres passe-bas du premier ordre : le filtre passif, le filtre actif avec un amplificateur opérationnel et le filtre actif avec un amplificateur à transconductance. Le filtre passe-bas

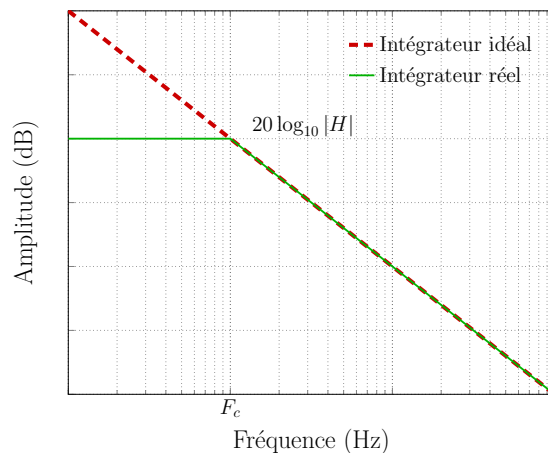


FIGURE 3.7 – Réponse en fréquence d'un intégrateur

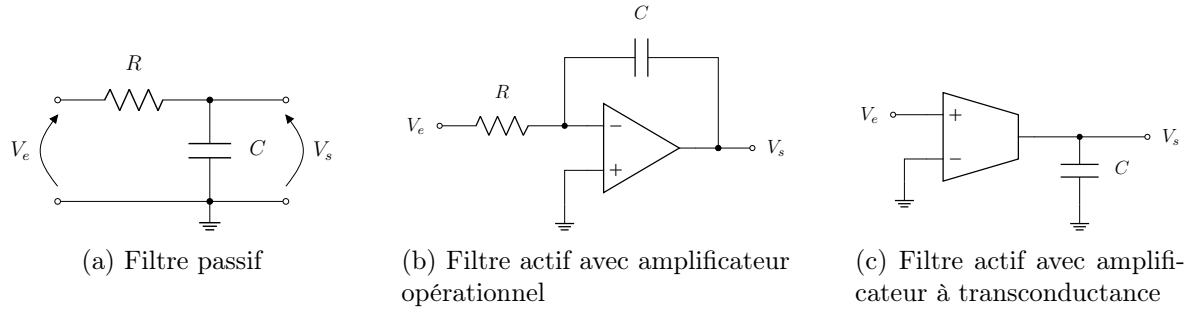


FIGURE 3.8 – Filtres passe-bas du premier ordre

basé sur un amplificateur à transconductance (Figure 3.8c) a pour avantages d'être moins complexe et d'avoir une consommation inférieure à celle qu'un amplificateur opérationnel. Il permet d'implémenter un gain et possède une impédance de sortie élevée.

La manière la plus simple de réaliser un amplificateur consiste à moduler l'intensité d'un courant en fonction de la tension à amplifier. Le courant ainsi modulé circule dans une résistance afin d'obtenir le signal amplifié. Un transistor NMOS peut être utilisé comme un modulateur linéaire tension/courant. En effet, il peut se comporter comme une source de courant d'intensité  $I_D = K(V_{gs} - V_T)^2$ , où  $K$  est une constante qui dépend de la technologie utilisée et des dimensions du transistor, lorsque  $V_{gs} > V_T$  et  $V_{ds} > V_{gs} - V_T$ .

Il est en général préférable d'utiliser des amplificateurs différentiels plutôt que des amplificateurs asymétriques, car les circuits différentiels sont beaucoup moins sensibles au bruit et aux interférences que les circuits asymétriques. En effet, si l'on considère deux fils transportant un petit signal différentiel et que l'on suppose qu'ils sont physiquement proches l'un de l'autre et affectés de manière égale par un signal d'interférence, alors les interférences seront égales sur les deux fils. Puisque, dans un système différentiel, seul le signal de différence entre les deux fils est détecté, il ne contiendra aucune composante d'interférence.

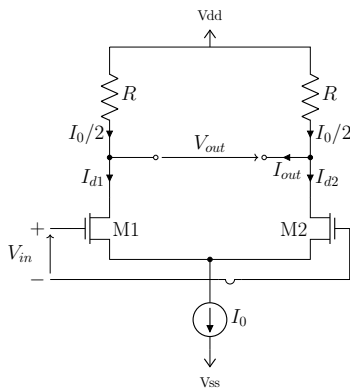


FIGURE 3.9 – Amplificateur MOS en configuration de paire différentielle

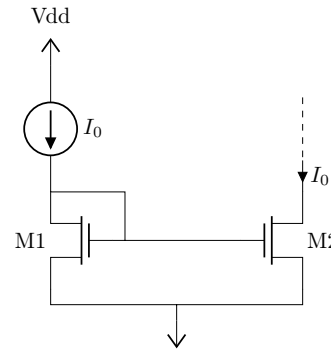


FIGURE 3.10 – Miroir de courant

TABLE 3.2 – Exemples de filtres passe-bas à très faible fréquence de coupure

Paramètre	[116]	[117]	[118]
Technologie ( $\mu\text{m}$ )	0.18	0.18	0.35
$V_{\text{supply}}$ (V)	1.8	1.8	1
Consommation ( $\mu\text{W}$ )	1.08	2.7	0.005
Surface ( $\text{mm}^2$ )	0.0156	0.051	0.07
$F_c$ (Hz)	0.22 - 39.1k	0.114 - 2.5k	0.002 - 90
Réglage	Tensions	Tension	Courant + Horloge

L'amplificateur différentiel le plus basique est celui utilisant une configuration de paire différentielle, comme illustré à la Figure 3.9. Il est constitué de deux transistors appariés, ici appelés  $M1$  et  $M2$ , dont les sources sont reliées et polarisées par une source de courant constant  $I_0$ . Cette dernière est généralement mise en œuvre par un circuit MOSFET comme un miroir de courant, illustré à la Figure 3.10. Bien que le drain de chaque transistor soit connecté à l'alimentation positive ( $V_{dd}$ ) via une résistance  $R$ , dans la plupart des cas, des charges actives comme des sources de courant, sont utilisées.

La problématique dans le contexte de la réalisation de cet intégrateur est de concevoir un filtre du premier ordre avec une fréquence de coupure très basse, c'est-à-dire de l'ordre du Hertz. En effet, pour réduire la fréquence de coupure, il est nécessaire d'utiliser une plus grande capacité, dont la taille est limitée dans un circuit intégré, ou bien de réduire la valeur de la transconductance, ce qui est difficile à réaliser avec une simple paire différentielle. Le Tableau 3.2 présente les caractéristiques de différentes propositions de filtres du premier ordre permettant d'obtenir une fréquence de coupure sous le Hertz. Le filtre proposé dans [116] et illustré à la Figure 3.11, semble être un choix intéressant. Tout d'abord, il présente une fréquence de coupure minimale de 220 mHz, ce qui répond à nos exigences. Sa consommation est inférieure à celle du filtre de [117] mais supérieure à celui de [118], en revanche il a pour avantage d'utiliser une technologie en 0.18  $\mu\text{m}$ , ce qui permet de garder une certaine cohérence par rapport au circuit frontal.

L'objectif de cette architecture [116] est de maximiser la capacité  $C$  et de minimiser la transconductance  $G_m$ . La Figure 3.12 présente le circuit, au niveau transistor, de la transconductance proposée dans [116]. Le circuit comprend plusieurs leviers de configuration différents :

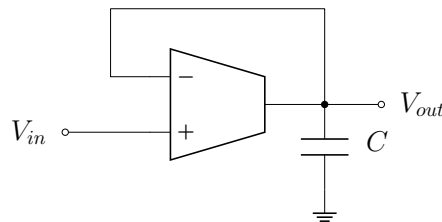


FIGURE 3.11 – Intégrateur de type  $G_m$ - $C$  proposée dans [116]

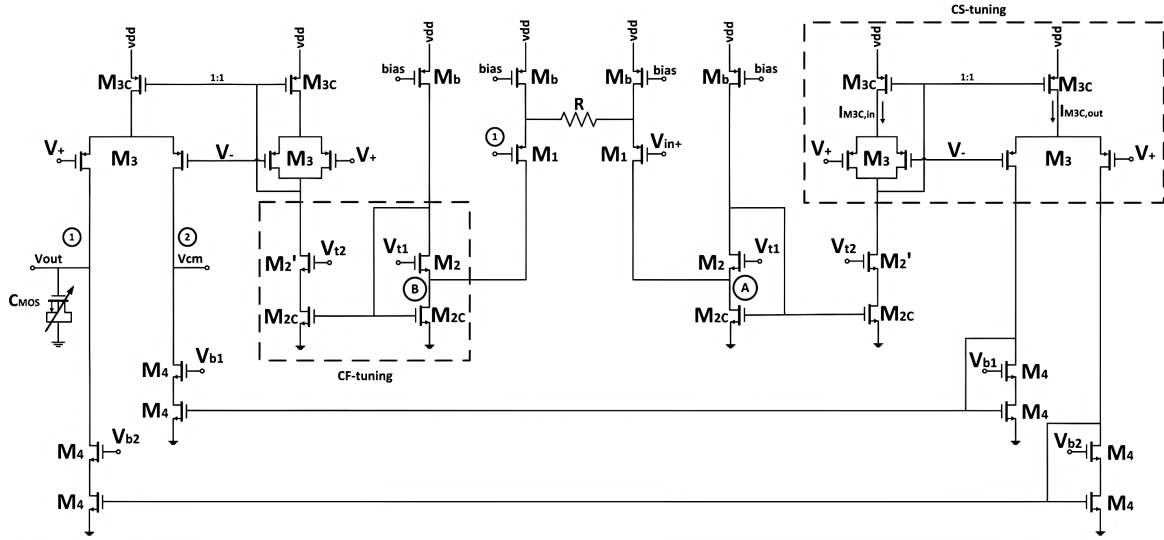


FIGURE 3.12 – Architecture complète de la cellule  $G_m$ -C proposée dans [116]

la tension  $V_{t2}$ , visible au niveau du bloc CF-tuning de la Figure 3.12, et la tension  $V_{gc}$  qui agit sur le circuit par l'intermédiaire des tensions  $V_+$  et  $V_-$ , visibles au niveau du bloc CS-tuning, suivant la relation :  $V_{\pm} = V_c \pm V_{gc}$ . Ces leviers permettent d'ajuster la transconductance et donc la fréquence de coupure du filtre. Le circuit possède également deux condensateurs de charge programmables numériquement différents (5 pF et 50 pF), pour étendre encore davantage la fréquence de coupure. Les graphiques de la Figure 3.13, issus de [116], montrent les différentes fréquences de coupure atteintes pour les différentes tensions  $V_{t2}$  et  $V_{gc}$ . Ils montrent que l'influence de ces deux tensions n'est pas la même sur la valeur de la fréquence de coupure : la tension  $V_{gc}$  permet d'atteindre une large gamme de fréquences, tandis que la tension  $V_{t2}$  permet d'effectuer un réglage beaucoup plus fin de la fréquence de coupure.

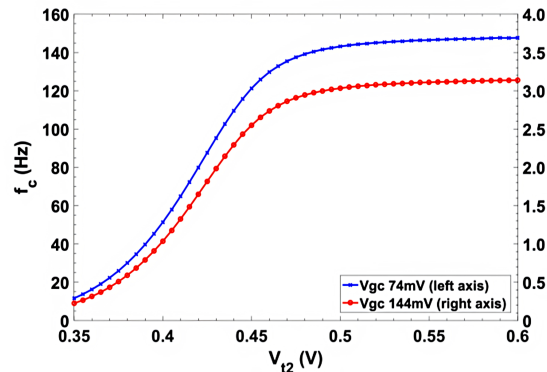
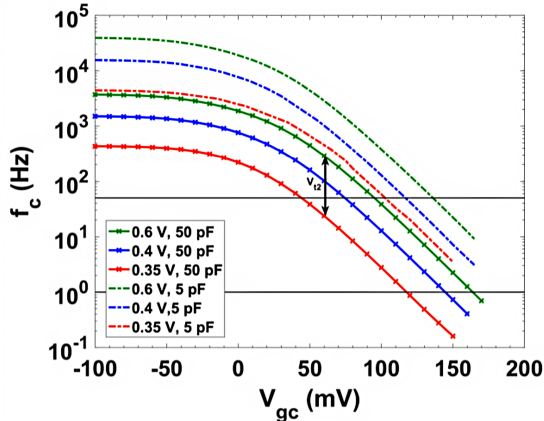


FIGURE 3.13 – Résultats présentés dans [116] montrant l'évolution de la fréquence de coupure du filtre en fonction des tensions  $V_{gc}$  (gauche) et  $V_{t2}$  (droite)

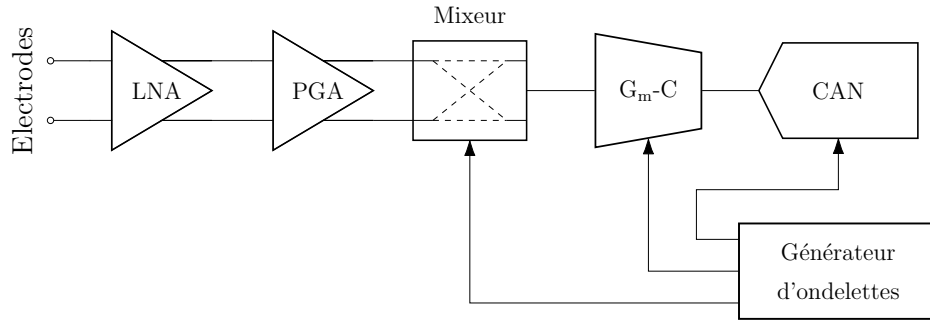


FIGURE 3.14 – Schéma matériel d'un extracteur de paramètre

Même si les caractéristiques de ce filtre correspondent aux critères fixés par l'application, il pose tout de même un problème. En effet, comme le montre le schéma de la Figure 3.11, le filtre fonctionne avec une entrée et une sortie simple. Il est donc nécessaire d'utiliser un bloc tampon entre la sortie du mélangeur et l'entrée du filtre, permettant ainsi de passer d'un signal différentiel à un signal simple (Single-ended).

### 3.2.4 Bilan

Comme illustré à la Figure 3.14, le circuit pour l'extraction de paramètres est constitué de plusieurs composants analogiques dont un étage d'amplification (LNA + PGA), un intégrateur de type  $G_m$ -C et un mélangeur basé sur des portes de transmission CMOS utilisées comme interrupteurs. Les caractéristiques de différents composants analogiques de la chaîne d'acquisition sont résumées dans le Tableau 3.3. La puissance nécessaire au fonctionnement de l'étage d'amplification et de l'intégrateur est de  $6.12 \mu\text{W}$ . Après l'extraction des paramètres de manière analogique, ceux-ci sont numérisés grâce à un CAN dont la figure de mérite est de  $14.3 \text{ fJ/échantillon}$ . Il ne reste maintenant plus qu'à estimer le coût énergétique de la génération des ondelettes en modélisant un générateur avec une architecture numérique.

TABLE 3.3 – Consommation du circuit front-end analogique

LNA et PGA [80]		LPF [116]		CAN [80]	
Technologie ( $\mu\text{m}$ )	0.18	Technologie ( $\mu\text{m}$ )	0.18	Technologie ( $\mu\text{m}$ )	0.18
Tension d'alimentation (V)	1.4	Tension d'alimentation (V)	1.8	Tension d'alimentation (V)	1
Courant ( $\mu\text{A}$ )	3.6	Courant ( $\mu\text{A}$ )	0.6	Courant ( $\mu\text{A}$ )	0.3
Gain (dB)	38 - 72	Capacité (pF)	5 - 50	SNDR (dB)	56.1
		Fréquence de coupure (Hz)	0.22 - 39.1k	ENOB (bits)	9.02
				$F_s$ (kHz)	40

### 3.3 Modèle de consommation du générateur d'ondelette

Dans le Chapitre précédent, nous avons décidé d'utiliser les ondelettes de Haar pour l'extraction de paramètres basé sur le NUWS. Ces ondelettes étant des signaux carrés, cela signifie qu'elles peuvent être facilement générées par un système électronique numérique.

#### 3.3.1 Architecture

Dans cette section, nous verrons dans un premier temps le principe de fonctionnement du générateur d'ondelettes, puis, dans un second temps, nous verrons plus en détails le fonctionnement des blocs numériques utilisés.

##### 3.3.1.1 Fonctionnement général du générateur

Les ondelettes de Haar sont des signaux à trois états : en effet, comme expliqué plus tôt, celles-ci alternent entre -1 et 1 sur la période où elles sont définies et valent zéro sinon. Cette définition permet de développer un générateur d'ondelettes avec un fonctionnement simple. La Figure 3.15 est un schéma simplifié du fonctionnement de notre générateur.

Les ondelettes sont ici décrites par trois paramètres : leur fréquence, leur longueur et leur position temporelle. Le système est composé d'un premier diviseur d'horloge qui permet de générer un signal d'horloge ayant une fréquence de 360 Hz, fréquence qui correspond à la fréquence d'échantillonnage de la base de données MIT-BIH et donc à la précision utilisée pour générer notre dictionnaire d'ondelettes. L'horloge à 360 Hz, couplée à un compteur permet de déterminer les instants auxquels les ondelettes doivent commencer et les instants auxquels elles se terminent. Le signal *START* permet de déclencher le début de la fenêtre d'analyse du signal ECG, le compteur est alors réinitialisé. La configuration des ondelettes est stockée

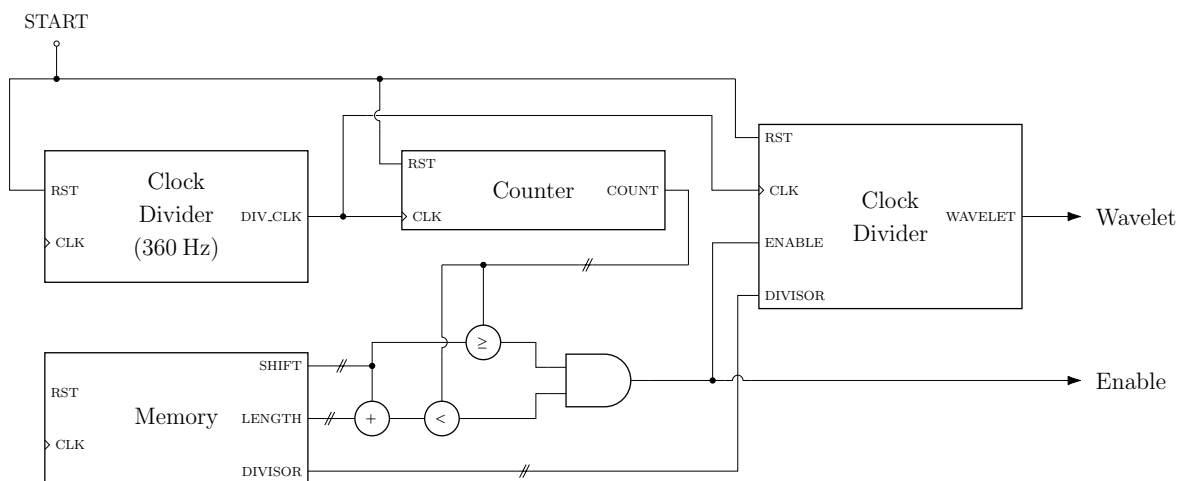


FIGURE 3.15 – Schéma simplifié du fonctionnement du générateur d'ondelettes

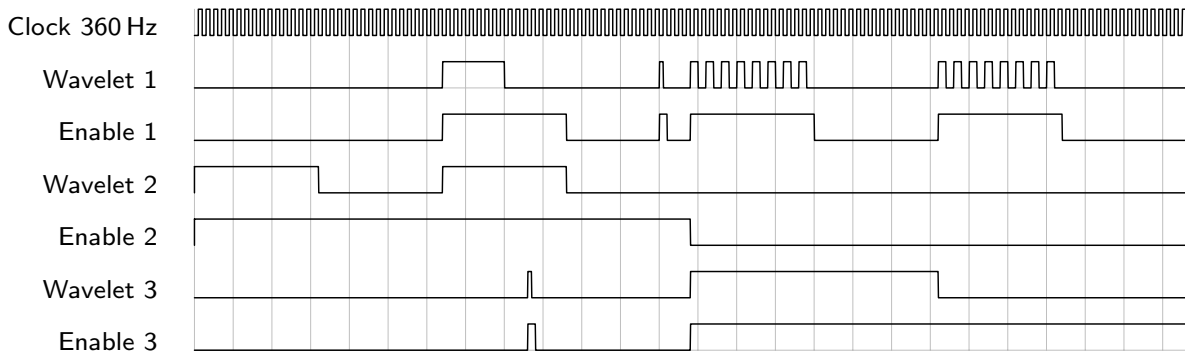


FIGURE 3.16 – Exemple de chronogramme obtenu avec le générateur d'ondelettes

dans une mémoire : le décalage temporel ( $SHIFT$ ) et la longueur d'une ondelette ( $LENGTH$ ) sont exprimés en nombre d'échantillons équivalent de la base de données et la fréquence est exprimée par le nombre par lequel l'horloge à 360 Hz doit être divisé pour obtenir la fréquence d'ondelette souhaitée ( $DIVISOR$ ). La somme des valeurs  $LENGTH$  et  $SHIFT$  correspond donc à l'instant auquel finit l'ondelette. Lorsque la valeur du compteur est donc comprise entre les valeurs  $SHIFT$  et  $SHIFT+LENGTH$ , alors le signal  $Enable$  prend la valeur logique "1". Cet intervalle indique donc l'intervalle de temps durant lequel l'ondelette est non-nulle et permet donc de délimiter le début et la fin de la période d'intégration. Ce signal permet également de signaler au CAN que le paramètre doit être numérisé. Durant cette période où l'ondelette est non-nulle, le second diviseur d'horloge est activé afin de générer la "forme" de l'ondelette à la fréquence souhaitée, afin d'activer ou désactiver les interrupteurs du mixeur analogique. La Figure 3.16 est un exemple de chronogramme obtenu dans une configuration utilisant trois générateurs d'ondelettes. Une ondelette est donc définie par deux signaux binaires, dont le codage peut être résumé par le Tableau 3.4.

### 3.3.1.2 Détails sur l'architecture numérique

Le fonctionnement d'un diviseur d'horloge est illustré à la Figure 3.17. Il est composé d'un compteur, d'un comparateur, d'une bascule  $D$ , et d'une porte logique  $NOT$ . Le compteur est incrémenté de 1 à chaque front montant de l'horloge  $CLK$ , le comparateur compare la sortie du compteur avec la valeur par laquelle on souhaite diviser la fréquence de l'horloge. Si la

TABLE 3.4 – Correspondance entre les signaux numériques générés et les valeurs théoriques des ondelettes

<i>Enable</i>	<i>Wavelet</i>	Valeur physique codée
0	X	0
1	0	-1
1	1	1

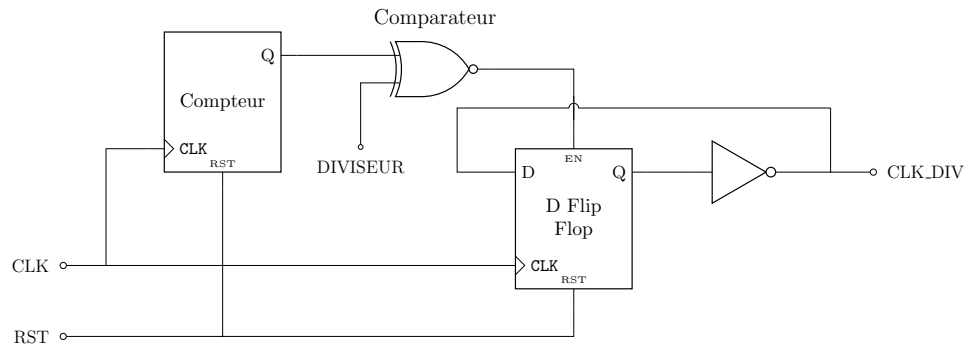


FIGURE 3.17 – Schéma d'un diviseur d'horloge

sortie du compteur est égale à cette valeur, alors la bascule  $D$  est autorisée à mettre à jour sa valeur : la sortie de cette bascule étant rebouclée sur son entrée à travers une porte  $NOT$ , on l'autorise donc à faire alterner sa sortie d'un état à l'autre.

La Figure 3.18 est un schéma matériel d'un compteur : un compteur de  $N$  bits est constitué d'autant de bascules et d'additionneurs complets qu'il y a de bits. Un additionneur comprend deux entrées, dont il fait la somme, et deux sorties, le résultat de la somme et une retenue. Un additionneur complet possède une troisième entrée pour la retenue, ce qui permet le chaînage d'additionneur. Les  $N$  additionneurs complets sont donc chaînés pour transmettre la retenue, chacun calculant la valeur d'un bit. Les sorties des bascules renvoient à la première entrée des additionneurs, tandis que la seconde entrée correspond à la valeur de l'incrément. Dans le cas de la Figure 3.18 et de notre étude, l'incrément à une valeur de 1, ce qui correspond à incrémenter de 1 le bit de poids faible à chaque front montant de l'horloge. Un second diviseur est utilisé pour générer les ondelettes à la fréquence souhaitée.

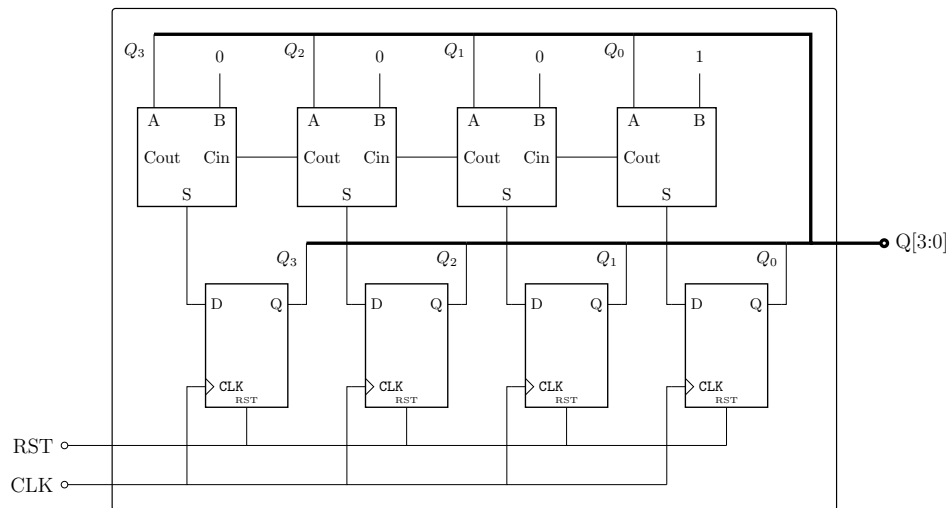


FIGURE 3.18 – Schéma d'un compteur 4 bits



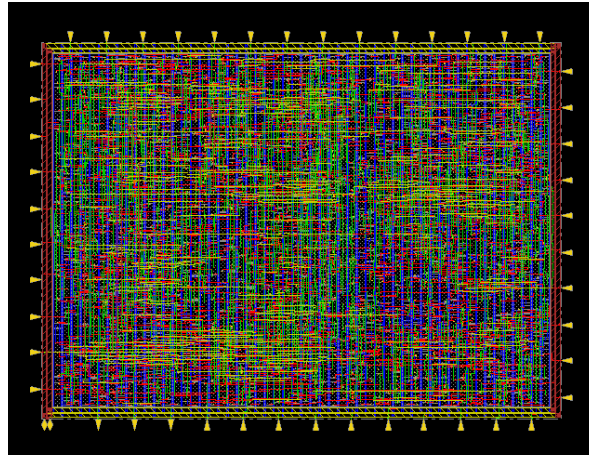
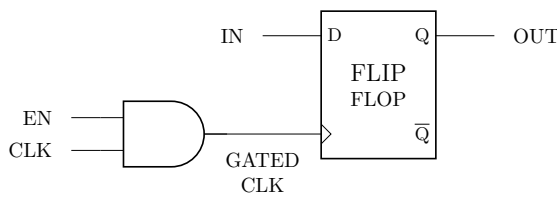


FIGURE 3.19 – Schéma de placement routage d'un générateur d'ondelettes dans la technologie XFAB 0.18  $\mu\text{m}$

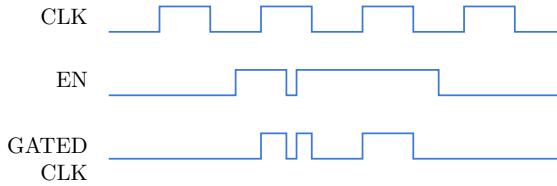
La mémoire utilisée pour stocker la configuration des ondelettes est une mémoire semblable à une mémoire séquentielle de type FIFO (First In — First Out). Les mémoires FIFO sont des mémoires que l'on peut voir comme des files d'attente : les données sont lues et écrites dans l'ordre de leur arrivée. Elles sont extrêmement courantes en électronique et sont notamment utilisées pour l'interfaçage de composants fonctionnant à des vitesses différentes qui doivent communiquer entre eux. Sur les mémoires FIFO, seules deux opérations sont possibles : écrire, c'est-à-dire ajouter une donnée à la file d'attente et lire, c'est-à-dire faire sortir la donnée la plus ancienne de la file d'attente. Une FIFO peut être fabriquée à partir d'une mémoire RAM, une mémoire volatile où l'information est conservée tant que celle-ci est alimentée, à laquelle on ajoute des circuits pour gérer les adresses de la donnée la plus ancienne et de la plus récente. La mémoire RAM peut être vue comme une matrice où chaque ligne correspond à un mot codé sur  $N$  bits. Une FIFO possède plusieurs entrées et sorties : une entrée pour l'horloge, deux bits de contrôle servant à déclencher l'écriture ou la lecture, une entrée pour le mot à écrire, une sortie pour le mot à lire et deux bits de contrôle indiquant respectivement si la mémoire est pleine ou vide. De manière analogue, la mémoire utilisée dans ce générateur d'ondelette stocke et lit les données dans l'ordre d'arrivée, mais contrairement à une FIFO, lorsque toutes les données sont lues, celles-ci ne sont pas perdues, on revient au début de la file d'attente pour relire une nouvelle fois les données. La lecture d'une nouvelle donnée dans la file d'attente est déclenchée par l'évènement de fin d'une ondelette, c'est-à-dire à un front descendant du signal *Enable*.

### 3.3.2 Réalisation du générateur en technologie CMOS

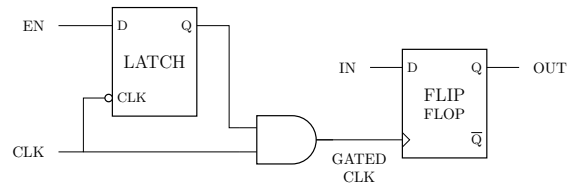
Ce générateur d'ondelette est décrit dans le langage de description matériel Verilog pour être simulé. Sa consommation d'énergie est alors estimée, après les étapes de synthèse et de placement routage dans la technologie XFAB 0.18  $\mu\text{m}$ , avec une fréquence de fonctionnement de



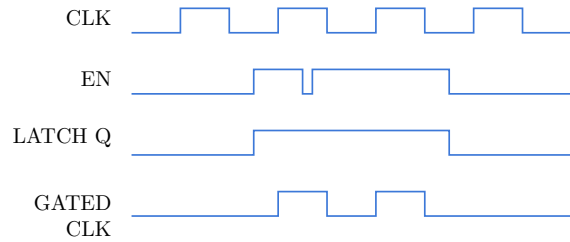
(a) Circuit basé sur une porte ET



(b) Chronogramme avec erreur



(a) Circuit basé sur un verrou



(b) Horloge d'activation sans erreurs

FIGURE 3.20 – Clock Gating basé sur une porte ET

FIGURE 3.21 – Clock gating basé sur un verrou

2 MHz. Pour cela, les outils GENUS et INNOVUS de la suite CADENCE® ont été utilisés. Après l'étape de synthèse, la complexité du circuit est de 1270 cellules parmi lesquels on compte 1017 portes logiques et 253 blocs séquentiels (bascules). Après placement routage, l'estimation de la consommation, durant la génération des ondelettes, est d'environ 22.5  $\mu$ W. La surface de silicium occupée par un générateur est de 0.033 mm<sup>2</sup>. Le résultat de la phase de placement routage est représenté dans la Figure 3.19.

### 3.3.3 Réduction de la consommation grâce au Clock Gating

Lors de la conception de circuits numériques synchrones, c'est-à-dire que l'état logique des sorties du système est piloté par une horloge, la consommation d'énergie du système est induite par trois composantes majeures : l'énergie consommée par les blocs de logique combinatoire, l'énergie consommée par les bascules et l'énergie consommée par l'arbre d'horloge. Pour réduire la consommation de ce bloc numérique, on peut utiliser une méthode très connue en électronique numérique, appelée *déclenchement d'horloge*, ou *Clock Gating* [119, 120]. Cette technique consiste à désactiver le signal d'horloge d'une partie du circuit lorsque celle-ci est inutilisée. Cette technique permet tout d'abord de diminuer la consommation de l'arbre d'horloge. En effet, en désactivant l'horloge des parties inutilisées, on diminue la charge de l'arbre d'horloge et donc le nombre de buffers utilisés dans celui-ci. La consommation des bascules (flip-flop), dont la consommation représente environ 68.5% de la consommation totale du système, est également réduite, car sans horloge, celles-ci ne sont pas déclenchées et donc l'énergie de fonctionnement dynamique est donc économisée. La contrepartie de cette technique est l'ajout de blocs logiques supplémentaires au circuit.

La méthode la plus basique pour le déclenchement de l'horloge est d'utiliser une porte *ET* ou *AND*, comme illustré dans la Figure 3.20a. Le signal permettant d'activer l'horloge (*EN*) est généré par un circuit de logique combinatoire et décide des instants auxquels l'horloge est fournie à la bascule en aval. Le problème de cette méthode est que l'intégrité du signal d'horloge d'activation généré n'est pas garanti. En effet, il peut apparaître des erreurs, également appelés glitches, comme illustré dans la Figure 3.20b. Ces glitches entraînent l'apparition de fronts montants supplémentaire qui peuvent déclencher des événements de manière prématurée.

Ce problème peut être évité en ajoutant un verrou, aussi appelé *Latch*, en amont de la porte *ET*, comme illustré dans la Figure 3.21. Un verrou est une bascule dont la sortie dépend du niveau logique de l'entrée de contrôle, c'est-à-dire que la bascule mémorise ou recopie l'entrée en fonction du niveau logique de l'entrée de contrôle, contrairement à une bascule qui recopie l'entrée lors d'un front (montant ou descendant) de l'entrée de contrôle. Dans une cellule de clock gating, le verrou recopie l'entrée *D* sur la sortie *Q* lorsque l'horloge (*CLK*) est à l'état bas (niveau logique 0) et garde en mémoire le niveau logique que possédait *D* au moment de la transition de l'horloge vers l'état haut (niveau logique 1). À la place d'un verrou, une bascule *D* aurait pu être utilisée, mais cette méthode est beaucoup moins populaire, car la bascule *D* a une consommation plus importante et occupe une plus grande surface que le verrou.

Le Tableau 3.5 met en lumière les différences de résultats obtenues avec et sans l'utilisation du Clock Gating. Il présente le nombre de cellules de différentes catégories (Séquentielle, logique, ...), nécessaire à la réalisation du générateur d'ondelettes, après synthèse. On constate ainsi que le nombre total de cellules nécessaire diminue alors que des instances liées aux clock gating ont été ajoutées, ce qui a pour conséquence de faire diminuer la surface du circuit. Grâce à cette technique, la consommation globale d'un générateur d'ondelette, pendant la phase de génération, a pu être réduite de 60%, passant de 22.5  $\mu\text{W}$  à  $P_{Generator} = 9 \mu\text{W}$ . La puissance nécessaire au fonctionnement du générateur est identique pour toutes les ondelettes de Haar du dictionnaire considéré.

TABLE 3.5 – Comparaison des résultats de synthèse, de surface et de consommation *AVEC* et *SANS* Clock Gating

	Sans Clock Gating	Avec Clock Gating
<b>Cellules Séquentielles</b>	253	253
<b>Inverseurs</b>	95	87
<b>Cellules logiques</b>	922	779
<b>Cellules de Clock Gating</b>	0	11
<b>Total</b>	1270	1130
<b>Surface (mm<sup>2</sup>)</b>	0.033	0.030
<b>Consommation (<math>\mu\text{W}</math>)</b>	22.5	9

### 3.4 Optimisation de la consommation

Nous souhaitons à présent prendre en considération le coût énergétique de l'extraction de chaque paramètre lors du processus de sélection. Un modèle de consommation de convertisseur A2F a été proposé dans ce sens. L'algorithme de sélection SFS optimisé, présenté dans l'Algorithme 2 de la Section 3.1, dérivé de l'Algorithme 1 de la Section 2.2.2.1, subit une nouvelle évolution. Cet algorithme a d'abord été modifié pour tenir compte du nombre d'extracteurs utilisé en parallèle et ainsi sélectionner des paramètres qui peuvent être extraits par un même générateur. Nous souhaitons maintenant introduire un nouveau critère de sélection qui ne tiendra pas uniquement compte de la précision de classification de chaque paramètre, comme c'était le cas jusqu'à présent, mais également du coût énergétique de l'extraction de celui-ci.

Pour minimiser la consommation, tout en maximisant les performances de classification, nous proposons d'utiliser un nouveau critère d'évaluation pour l'algorithme SFS. Ce nouveau critère consiste à faire le rapport entre le gain de précision qu'apporte le paramètre testé et son coût en énergie. Pour un gain de précision équivalent, une ondelette avec un coût énergétique plus faible sera donc favorisée. Ce nouveau critère peut ainsi s'écrire de la façon suivante :

$$J_i = \frac{A(S \cup \{X_i\}) - A(S)}{E_{feature}^i} \quad (3.1)$$

où  $S$  est l'ensemble des paramètres déjà sélectionné,  $X_i$  est le paramètre testé,  $A(S)$  correspond à la précision de classification de l'ensemble  $S$  et  $E_{feature}^i$  est l'énergie requise pour l'extraction du paramètre  $X_i$ . La valeur initiale de  $A$  est fixée à  $A(0) = 85\%$ , qui correspond à la précision du classifieur ZR, utilisé précédemment comme référence.

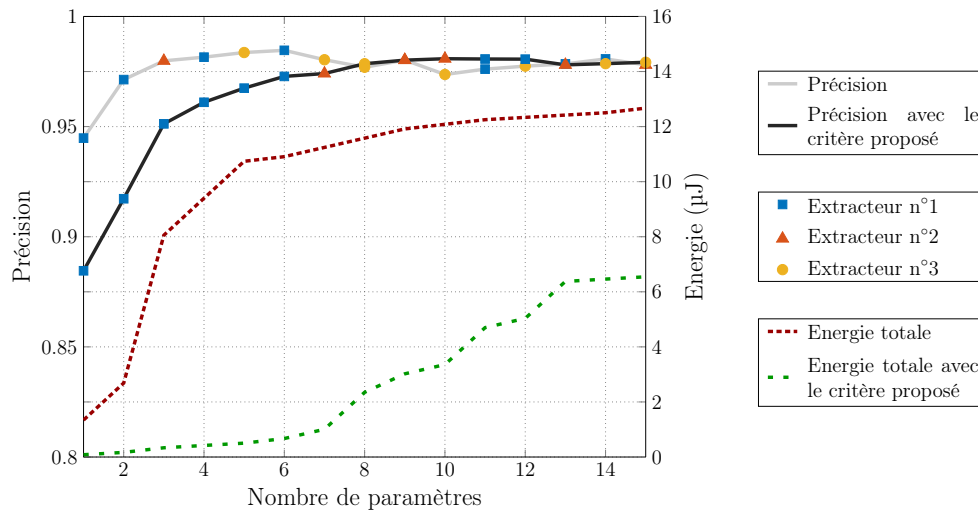


FIGURE 3.22 – Comparaison de l'évolution de la précision de classification et du coût d'extraction des paramètres lorsque le coût d'extraction est pris en compte lors de la sélection

L'estimation du coût d'extraction de chaque paramètre est calculée selon la formule suivante :

$$E_{feature} = E_{sampling} + E_{analog} + E_{digital} \quad (3.2)$$

où  $E_{sampling}$  correspond à l'énergie requise pour la conversion d'un échantillon par le CAN et qui est constante quel que soit le paramètre,  $E_{analog}$  correspond à l'énergie requise par la partie analogique de l'extracteur (LNA, PGA et intégrateur) et qui est proportionnelle à la longueur de l'ondelette, et  $E_{digital}$  correspond à la quantité d'énergie dont la partie numérique a besoin pour générer l'ondelette. Ces trois grandeurs sont définies par les relations suivantes :

$$E_{sampling} = \frac{P_{CAN}}{2^{ENOB} \cdot F_s} \quad (3.3)$$

$$E_{analog} = \Delta t \cdot (P_{LNA} + P_{PGA} + P_{LPF}) \quad (3.4)$$

$$E_{digital} = \Delta t \cdot P_{Generator} \quad (3.5)$$

où  $\Delta t$  est la période de temps durant laquelle l'ondelette est non-nulle et  $P_{CAN}$ ,  $P_{LNA}$ ,  $P_{PGA}$ ,  $P_{LPF}$  et  $P_{Generator}$  correspondent respectivement aux puissances du CAN, de l'étage d'amplification (LNA + PGA), de l'intégrateur et du générateur d'ondelettes. La Figure 3.22 montre les performances de classification, ainsi que la somme cumulée du coût énergétique des paramètres sélectionnés par l'algorithme SFS utilisant ce nouveau critère. Nous pouvons voir que la précision atteinte avec l'ancien critère, qui ne tenait compte que de la précision de classification, est d'environ 98.4% pour six paramètres extraits, avec trois extracteurs, et que le coût énergétique qui en résulte est d'environ 10.9  $\mu$ J. Avec le nouveau critère, on atteint une précision de 98% pour dix paramètres extraits, avec toujours trois extracteurs utilisés, mais avec un coût énergétique d'environ 3.35  $\mu$ J. Lorsque l'on compare les performances maximales obtenues, l'utilisation de ce nouveau critère permet de diviser le coût énergétique de l'extraction des paramètres par 3.3 alors que le nombre de paramètres nécessaire augmente, passant de six à dix. On constate également que le choix des premiers paramètres réalisé avec l'ancien critère est plus performant en termes de classification, mais entraîne rapidement une perte de performance qui peut être due à un effet de sur-apprentissage, par rapport aux paramètres sélectionnés avec le critère proposé.

### 3.5 Comparaison de la consommation avec l'état de l'art

Nous pouvons à présent faire une comparaison entre les trois différentes approches de capteurs sans fils et ainsi montrer les avantages apportés par les capteurs utilisant les convertisseurs A2F par rapports à ceux utilisant les convertisseurs A2I, basés sur le CS, et les approches classiques, réalisant une acquisition à la fréquence de Nyquist. Pour cela, nous pouvons réutiliser

TABLE 3.6 – Tableau regroupant les données considérées pour l'estimation de la consommation des différents types de capteurs

		Capteur classique	Capteur A2I	Capteur utilisant un convertisseur A2F
Acquisition	Amplification	LNA + PGA [80] 5.04 $\mu$ W	Architecture [4] (32 canaux) 0.9 $\mu$ W	LNA + PGA [80] 5.04 $\mu$ W
	Filtrage	Filtre $G_m$ -C [80] 0.7 $\mu$ W		–
	Extraction de paramètres/ information	–		Intégrateur $G_m$ -C [116] + Générateur d'ondelettes 10.08 $\mu$ W
	Conversion	CAN [80] 0.3 $\mu$ W		CAN [80] 0.3 $\mu$ W
Transmission RF		Bluetooth Low Energy [42] 3.7 nJ/bit		

les données déjà utilisées pour comparer ces deux dernières approches dans la Section 1.3.5 et illustrées à la Figure 1.25, où sont considérées l'approche classique présentée dans [80] et l'architecture A2I de [4].

La chaîne d'acquisition considérée pour l'approche classique est donc celle présentée dans [80]. Cette chaîne est composée d'un étage d'amplification (LNA + PGA), d'un filtre passe-bas et d'un CAN SAR ayant une précision de 10 bits, la puissance totale de cette chaîne d'acquisition est ainsi de 6.04  $\mu$ W. Cette chaîne d'acquisition consomme donc 3.02 nJ par échantillon, pour une fréquence d'échantillonnage  $F_s$  de 2 kHz.

L'architecture proposée dans [4], est considérée pour l'estimation de la consommation d'un capteur utilisant un convertisseur A2I. Cette architecture possède 64 branches pouvant fonctionner en parallèle et a une puissance de 1.8  $\mu$ W. Pour l'acquisition de signaux ECG, celui-ci est configuré pour fonctionner avec  $M = 32$  branches et une fenêtre temporelle d'analyse équivalente à  $N = 128$  cycles d'échantillonnage de Nyquist. Ce qui correspond à un taux de compression de 4.

L'estimation de la consommation du convertisseur A2F est réalisée en considérant la consommation des différents éléments nécessaires à l'extraction des paramètres (voir Figure 3.14) : ce modèle prend donc en considération la consommation de l'étage d'amplification (LNA + PGA) de [80], du filtre passe-bas, utilisé comme intégrateur, présenté dans [116], du générateur d'ondelettes présenté dans la Section 3.3, et du CAN de [80]. La configuration du convertisseur A2F considérée ici est celle obtenue à la Figure 3.22, où l'extraction de 10 paramètres permet d'obtenir une précision de 98% avec un coût énergétique d'environ 3.35  $\mu$ J. La consommation induite par le système de synchronisation pour les signaux ECG, n'est pas considérée ici. En effet, si des circuits pour la détection des complexes QRS sont proposés,

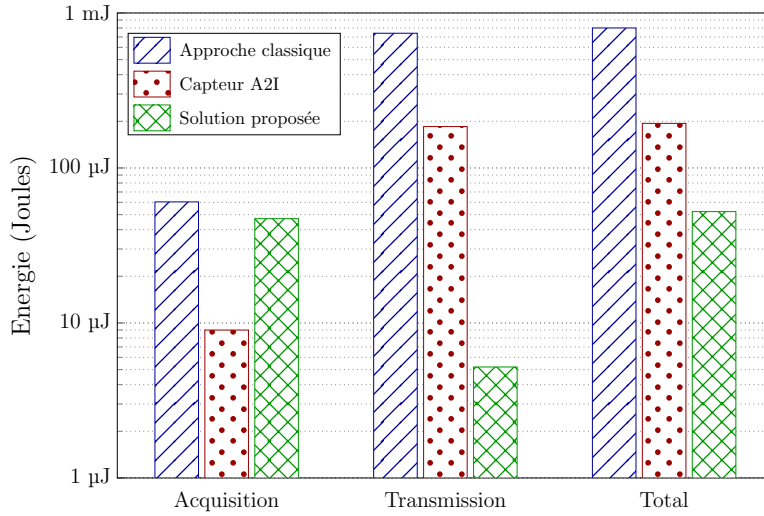


FIGURE 3.23 – Comparaison de l'énergie requise pour l'acquisition de 10 secondes de signal avec 3 types de capteurs

comme dans [114] où le circuit a une consommation de 31.5 nW, l'estimation de la consommation d'un système de prédiction, ou d'une cellule de retard, reste à faire. De plus, nous considérons que le capteur utilisant un convertisseur A2F transmet les paramètres extraits et l'étape de classification n'est donc pas réalisée au niveau du capteur.

Le système de transmission sans-fil considéré dans les trois situations est issu de [42]. Il utilise le standard Bluetooth Low Energy (BLE) et consomme 3.7 nJ par bit transmis. Dans les trois cas, le convertisseur utilisé possède une précision de 10 bits, la consommation est donc de 37 nJ par échantillon transmis. Le Tableau 3.6 résume toutes les données de consommation considérées pour cette comparaison.

L'approche au taux de Nyquist requiert 3.02 nJ par échantillon et fonctionne à la fréquence  $F_s = 2$  kHz. L'architecture A2I est composée de  $M = 32$  branches, chacune ayant une puissance de 28 nW et analyse le signal sur une fenêtre temporelle de  $N = 128$  échantillons. Ici l'énergie nécessaire pour l'étape de reconstruction n'est pas prise en compte. Le système de transmission nécessite 3.7 nJ par bit transmis, soit 37 nJ par échantillon de 10 bits. À partir de ces données, l'énergie nécessaire à l'acquisition et à la transmission de 10 secondes de signal ECG est évalué pour ces trois types de capteurs. La Figure 3.23 présente les résultats obtenus : pour l'approche classique, 800 μJ sont nécessaires, tandis que le convertisseur A2I requiert 194 μJ. La solution de convertisseur A2F proposée ne consomme que 52.3 μJ, l'énergie nécessaire est pour le traitement de 10 secondes de signal est donc divisée par 15.3 par rapport à l'approche classique et par 3.7 par rapport au capteur utilisant un convertisseur A2I. Par rapport à la solution basée sur le convertisseur A2I, la consommation due à la transmission est divisée par 35 et la consommation du circuit multipliée par 5.4. La consommation du processus d'acquisition, dans la solution utilisant un convertisseur A2F, représente 93.5% de la consommation totale.

Une implémentation physique de convertisseur A2F complète et optimisée permettrait ainsi de réduire la consommation du circuit. Ces données montrent bien l'intérêt de cette solution pour diminuer la consommation des capteurs sans-fil et ainsi augmenter leur durée de vie. Ces chiffres sont toutefois à relativiser, en effet, ils sont obtenus dans le cas relativement simple d'une détection d'anomalie binaire.

Nous souhaitons à présent évaluer l'impact sur la consommation du capteur de l'implémentation d'un réseau de neurones au niveau du capteur. L'utilisation d'un tel système permettrait de réduire encore plus la quantité de données transmises, en transmettant uniquement le résultat de classification. Le réseau de neurones implémenté dans [121], est un perceptron multicouches avec une couche cachée, avec respectivement sept, six et cinq neurones sur ses différentes couches. Le réseau est implémenté sur un FPGA pour une consommation de 120 mW et un temps de réponse de 270 ns. Les données utilisées dans cette étude sont issues de [121] et n'ont pas été réévaluées par nos soins. Le FPGA utilisé est un FPGA XILINX ARTIX-7® [122] sur une carte ARTIX-7 35T. Cette implémentation a été choisie, car la topologie du réseau de neurones est proche de celle utilisée pour notre application. D'autres solutions d'implémentation peuvent également être considérées comme des réseaux de neurones implémentés sur des microcontrôleurs [123] ou des réalisations physiques en technologie CMOS [124].

Dans le contexte du convertisseur A2F, l'ajout du réseau de neurones au niveau du capteur ne modifie pas la consommation induite par l'extraction des paramètres, celle-ci reste donc inchangée par rapport au résultat obtenu précédemment, soit 47.1  $\mu$ J. Concernant la transmission des données, le capteur utilisant un convertisseur A2F, considéré précédemment, transmettait 10 paramètres codés sur 10 bits pour un coût total de 5.2  $\mu$ J. Si l'on considère à présent que seul le résultat de classification est transmis par le capteur et que l'on se trouve dans le cas d'une détection binaire d'anomalies ECG, comme c'est le cas dans cette étude, la quantité de données transmise est ainsi divisée par cent, en effet le nombre d'échantillons et le nombre de bit sont tous deux divisés par dix. On peut alors estimer l'énergie consommée par l'étage de transmission à 52 nJ. Concernant le coût énergétique du réseau de neurones proposé dans [121], celui-ci peut être évalué en connaissant la puissance nécessaire à son fonctionnement, le temps nécessaire pour réaliser une classification et le nombre de classifications réalisées pendant ces dix secondes. Les deux premières valeurs ont été évoquées précédemment, la puissance est de 120 mW et le temps de réponse est de 270 ns. Étant donné la fréquence d'échantillonnage du signal (360 Hz) et la taille de la fenêtre d'analyse du signal ECG, soit 256 échantillons, nous pouvons déterminer le nombre de classifications réalisées pendant dix secondes qui est donc d'environ 14.0625. Le coût énergétique pour la classification de dix secondes de signal est donc évalué à 455 nJ.

Au total, l'énergie nécessaire pour l'acquisition, le traitement et la transmission des données est d'environ 47.6  $\mu$ J. L'implémentation d'un réseau de neurones au niveau du capteur permettrait ainsi de diminuer la consommation du capteur d'environ 9% par rapport à un capteur transmettant les paramètres extraits. La Figure 3.24 permet de mettre en perspective



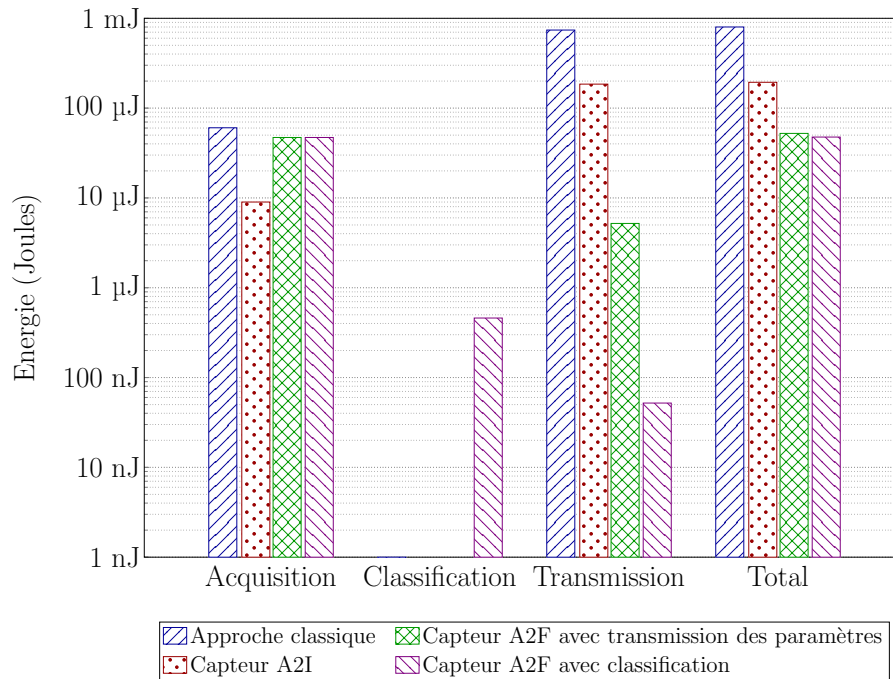


FIGURE 3.24 – Comparaison de l'énergie requise pour l'acquisition de 10 secondes de signal entre les capteurs de l'état de l'art et les solutions à base de convertisseur A2F

les résultats obtenus précédemment et présentés à la Figure 3.23 et les résultats de l'estimation de consommation d'un capteur réalisant la classification localement. La consommation du processus d'acquisition représente à présent environ 99% de la consommation totale du capteur. Ceci laisse place à de fortes améliorations sur la consommation du système, en effet, un travail d'optimisation sur la consommation du circuit d'acquisition et d'extraction des paramètres nous garantirait une optimisation de la consommation globale du capteur.

### 3.6 Au-delà de la conversion analogique-paramètre

L'objectif des convertisseurs A2F est de numériser le résultat d'une phase d'extraction de paramètres dans le domaine analogique pour réaliser une étape de classification (ou de régression) de manière numérique. Dans la Section 3.5, nous avons évoqué une solution d'implémentation, au niveau du capteur, d'un réseau de neurones utilisant une architecture numérique, permettant ainsi d'estimer une réduction de la consommation d'environ 9%. Des avancées dans le domaine des réseaux de neurones (NN) artificiels, analogiques et reconfigurables, pourraient permettre de réduire encore plus la consommation des capteurs sans-fil. En effet, l'utilisation d'un NN analogique permettrait de ne numériser que le résultat fourni par le modèle d'apprentissage automatique. Cela réduirait les contraintes de bande passante du CAN, de plus, cela permettrait de réduire la quantité de données à transmettre par le capteur. On peut également considérer, comme dans le cas de la détection d'anomalies ECG, que le capteur ne transmette le résultat de classification que lorsqu'une anomalie est détectée.

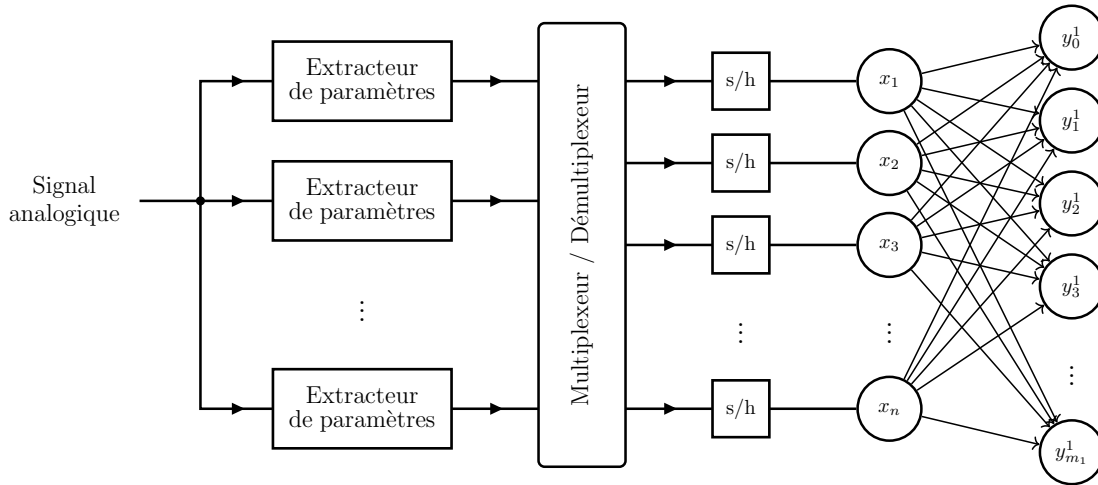


FIGURE 3.25 – Convertisseur A2F avec réseau de neurones analogique

Des implémentations matérielles de NN analogiques ou mixtes sont proposés dans [125, 126, 127]. Ces circuits sont basés sur des systèmes, appelés *memristor*, qui permettaient de surmonter les problèmes de surface et de puissance liés aux implémentations classiques en technologie CMOS. Cet élément, proposés en 1971 par Leon Chua [128] et implémenté la première fois en 2008 [129], est un élément de circuit passif qui agit comme une résistance variable dont la valeur peut être pilotée grâce à un courant ou à une tension. La résistance  $R$  à un instant  $t$  d'un memristor, également appelé memristance, est défini par la relation :

$$R(t) = M(t) = \frac{v(t)}{i(t)} = \frac{d\phi}{dq} \quad \text{avec} \quad \begin{cases} q(t) = \int_{-\infty}^t i(\tau) d\tau \\ \phi(t) = \int_{-\infty}^t u(\tau) d\tau \end{cases} \quad (3.6)$$

où  $\phi(t)$  et  $q(t)$  sont respectivement le flux et la charge électrique. Du point de vue physique, un memristor est composé de deux électrodes entre lesquelles se trouvent deux couches, l'une dopée et l'autre non dopée. Lorsqu'une tension ou un courant est appliqué, la ligne de séparation entre les deux couches se décale en fonction de la tension ou du courant appliqué. En conséquence, la résistance entre les deux électrodes est modifiée. Les memristors permettent d'implémenter des mémoires non volatiles et avec une taille extrêmement petite de quelques nanomètres, ce qui les rend idéales pour régler les poids synaptiques de réseaux de neurones artificiels.

L'utilisation de memristors dans les réseaux de neurones analogiques permet non seulement d'obtenir des réseaux reconfigurables, mais également la mise en œuvre sur puce d'algorithmes d'apprentissage tels que les algorithmes de rétro-propagation [125, 127]. Dans le contexte des convertisseurs A2F, l'utilisation de ce type de réseaux permettrait d'obtenir des réseaux reconfigurables permettant de s'adapter à l'application ou au contexte, mais aussi de réaliser une phase d'apprentissage plus précise qui tiendrait compte des spécificités du circuit, mais également des signaux étudiés. Dans le cas des signaux ECG, une phase d'apprentissage réalisée sur le signal du patient observé permettrait d'obtenir une meilleure précision lors de la phase

de détection des anomalies. La Figure 3.25 représente le nouveau système proposé où la sortie des extracteurs de paramètres sont connectées à des blocs échantillonneurs, ces derniers sont utilisés afin de stocker les valeurs des paramètres dans le cas où les extracteurs doivent être remis à zéro pour extraire d'autres paramètres. Ces blocs échantillonneurs constituent alors les entrées du réseau de neurones analogique à base de memristors.

### 3.7 Conclusion

Les méthodes de sélection de paramètres permettent de maximiser le taux de compression d'un convertisseur A2F basé sur le NUWS, et donc de réduire au maximum la consommation du capteur. Pour optimiser cette dernière, un nouvel algorithme de sélection est proposé : il permet de tenir compte de l'architecture du convertisseur A2F et du coût énergétique de l'extraction de chaque paramètre. En effet, cet algorithme permet, dans un premier temps, de fixer le nombre maximal d'extracteurs utilisés par le convertisseur A2F, il sélectionne alors des paramètres de façon à ce qu'un même extracteur puisse acquérir plusieurs paramètres en se basant sur la taille et la position du support temporel des ondelettes associées. De plus, un modèle de consommation de convertisseur A2F est proposé afin d'évaluer le coût d'extraction de chaque paramètre : ce modèle est réalisé pour une technologie CMOS 0.18  $\mu\text{m}$ , à partir de circuits analogiques issus de l'état de l'art, pour une consommation de 6.42  $\mu\text{W}$ , et de la simulation d'un générateur d'ondelettes de Haar numérique ayant une consommation de 9  $\mu\text{W}$ . Un nouveau critère d'évaluation pour l'algorithme de sélection SFS est donc proposé afin de maximiser le gain de précision de classification tout en minimisant le surcoût en énergie qu'entraîne l'extraction d'un nouveau paramètre.

Dans le Chapitre 2, la précision atteinte était d'environ 98.7% pour sept paramètres extraits. En utilisant cette nouvelle méthode de sélection, une précision de 98% est atteinte pour dix paramètres extraits, avec un coût énergétique d'environ 3.3  $\mu\text{J}$ . En limitant le nombre d'extracteurs à trois, l'ajout d'un critère d'évaluation tenant compte de l'énergie nécessaire à l'extraction permet de diviser la quantité d'énergie nécessaire par 3.3 en extrayant quatre paramètres de plus, tout en perdant seulement 0.4% de précision de classification par rapport au processus de sélection qui limite seulement le nombre d'extracteurs. La comparaison du convertisseur A2F avec d'autres techniques de l'état de l'art, utilisant l'échantillonnage de Nyquist ou l'échantillonnage compressif, a permis de montrer que l'utilisation d'un convertisseur A2F pour le traitement et la transmission de dix secondes de signal ECG, permettrait de diviser par 15.3 la quantité d'énergie nécessaire par rapport à l'approche classique et par 3.7 par rapport au capteur utilisant un convertisseur A2I. La part dédiée au circuit dans la consommation totale du capteur représente 93.5%, ce qui laisse place à une réelle amélioration de l'efficacité énergétique du circuit via une implémentation physique complète et optimisée du convertisseur A2F.



# Conclusions et Perspectives

## Conclusions sur les travaux réalisés

Les différentes évolutions dans le domaine des semi-conducteurs apparues lors des dernières années a permis de considérablement réduire la taille et la consommation des systèmes intégrés. Ces deux aspects ont permis de démocratiser et de rendre plus abordables les technologies liées à l'internet des objets et aux objets connectés en général. Les capteurs corporels constituent un bon exemple d'application ayant de très fortes contraintes notamment en ce qui concerne la taille et la consommation. Les recherches concernant les réseaux de capteurs sans-fil visent à rendre ces derniers plus intelligents et à leur conférer une plus grande autonomie. C'est dans ce contexte qu'a été proposée la conversion analogique/paramètres.

L'analyse de la chaîne d'acquisition et de traitement de l'information des capteurs sans-fil a permis de mettre en évidence le coût énergétique des différents processus impliqués et de conclure que la transmission des données constitue la majeure partie des dépenses énergétiques de ces derniers. L'échantillonnage compressif est une approche qui permet de réduire la quantité de données à transmettre, par rapport aux approches classiques fonctionnant au taux de Nyquist. Elle permet de faire l'acquisition de données compressées et ensuite de reconstruire le signal d'origine grâce à un algorithme de reconstruction. Le problème de cette approche réside dans la complexité de ce type d'algorithmes [3] et dans le fait que le taux de compression est limité [4].

L'intérêt de l'extraction de paramètres dans le domaine analogique pour diminuer la consommation d'énergie grâce à la réduction de la quantité de données a été démontré dans de précédents travaux [5]. La conversion analogique-paramètre permet d'extraire de l'information du signal étudié, pour des applications de détection utilisant l'apprentissage automatique, et d'ainsi utiliser de façon plus parcimonieuse l'étage de transmission des capteurs. La technique de l'échantillonnage non-uniforme en ondelettes [57] est proposée comme méthode d'extraction de paramètres pour concevoir un convertisseur A2F générique pour le traitement d'une large gamme de signaux basse fréquence. Cette thèse propose une méthode pour sélectionner les paramètres pertinents pour une application donnée tout en tenant compte de l'architecture et de l'implémentation d'un convertisseur A2F pour en optimiser la consommation.

La première étape a été de mesurer les potentielles performances, en termes de précision de classification et de réduction de la quantité de données, d'un convertisseur A2F utilisant le NUWS dans le cas d'une application de détection d'anomalie dans des signaux ECG. Des algorithmes de sélection de paramètres sont utilisés pour identifier les paramètres pertinents pour la phase de classification, parmi une base de données de 502 paramètres obtenus via l'échantillonnage non-uniforme d'une transformée en ondelettes de Haar. La combinaison d'algorithmes de sélection comme le gain d'information, le score de Fisher ou le Sequential forward search, ont permis d'obtenir une précision de classification de 98.7%, grâce à un réseau de neurones, en utilisant seulement sept paramètres et d'ainsi réduire de plus de 98% la quantité de données par rapport à un échantillonnage au taux de Nyquist. Des études sur le choix du type d'ondelettes et sur le niveau de quantification nécessaire pour la numérisation des paramètres ont d'abord permis de conclure que les ondelettes de Haar étaient un choix judicieux étant donné leur simplicité et la précision de classification obtenue, puis de montrer qu'une quantification sur 6 bits est suffisante pour obtenir une bonne précision. La localisation temporelle réduite de certaines ondes du complexe QRS nécessite une synchronisation entre la génération des ondelettes et l'arrivée de ces complexes, afin d'extraire toutes les informations d'un même complexe au cours de la même "fenêtre" d'analyse. L'utilisation d'un réseau de neurones récurrent comme prédicteur et la réalisation d'une étape de sélection avec des données désynchronisée permettent d'obtenir une précision de classification de "seulement" 95%.

Afin de démontrer l'utilité des convertisseurs A2F par rapport aux approches de Nyquist et aux convertisseurs A2I, une méthode de sélection des paramètres est proposée afin de minimiser la consommation d'un convertisseur A2F à base de NUWS. L'objectif de cette nouvelle méthode de sélection est, en plus de tenir compte de la précision de classification des paramètres, de tenir compte également de l'architecture du convertisseur A2F, en mutualisant l'extraction de plusieurs paramètres avec un même extracteur afin d'en limiter le nombre, et du coût énergétique de l'extraction de chaque paramètre. L'évaluation du coût des différents paramètres est rendue possible par la réalisation d'un modèle de consommation du convertisseur A2F en technologie CMOS 0.18  $\mu\text{m}$ . Ce modèle est réalisé à partir de données de composants analogiques issus de l'état de l'art et de la réalisation d'un générateur d'ondelettes numérique ayant une consommation de 9  $\mu\text{W}$ . Cette nouvelle étape de sélection a permis de mettre en évidence de nouveaux paramètres permettant l'obtention de performances de classification similaire à ce qui avait pu être obtenu précédemment, soit environ 98% de précision, tout en divisant par 3.3 la quantité d'énergie nécessaire pour l'extraction des paramètres alors que l'on extrait plus de paramètres. Ce modèle de consommation permet également de montrer l'intérêt de la conversion analogique-paramètre par rapport à l'échantillonnage au taux de Nyquist ou l'échantillonnage compressif. L'utilisation du convertisseur A2F pour l'acquisition et la transmission de dix secondes de signal ECG permettrait de diviser la quantité d'énergie nécessaire par 15.3, par rapport à l'approche classique, et par 3.7 par rapport aux capteurs utilisant un convertisseur A2I.

Par rapport aux travaux menés dans [8], cette thèse a permis de réaliser un affinage de la sélection des paramètres permettant ainsi d'obtenir un bien meilleur ratio précision/nombre de paramètres sélectionnés, de proposer une évaluation plus réaliste des effets sur la précision de classification d'une mauvaise synchronisation des signaux ECG en considérant des écarts beaucoup plus réalistes grâce à un modèle de prédiction de l'intervalle R-R. Une étude sur la généralité du système, exploitant la reconfigurabilité du convertisseur A2F, avec l'étude d'une autre application pour la catégorisation d'anomalies ECG. Cette thèse propose également une méthode de sélection permettant l'optimisation de la consommation d'un convertisseur A2F basé sur le NUWS, grâce à la réalisation d'un modèle de consommation basé sur des données de l'état de l'art et de la réalisation d'un circuit numérique personnalisé pour la génération des ondelettes de Haar.

## Perspectives

Ces travaux ouvrent de nombreuses perspectives de recherche. Le choix du type d'ondelettes à utiliser pour le traitement de signaux ECG et la sélection adaptée à l'architecture ont permis de définir les contours d'une implémentation physique d'un convertisseur A2F. La consommation du circuit utilisé pour l'acquisition et l'extraction des paramètres représente 93.5% du budget énergétique total du capteur dans le cas où les paramètres sont transmis et 99% dans le cas où la classification est réalisée au niveau du capteur. Cela laisse place à une réelle amélioration de l'efficacité énergétique du circuit via une implémentation physique complète et optimisée du convertisseur A2F. Une seconde piste peut être d'affiner le modèle de classification afin de reconnaître le type d'anomalie et plus seulement détecter la présence d'anomalie ou détecter une catégorie d'anomalie. Cela permettrait ainsi de mieux exploiter la reconfigurabilité des extracteurs et permettre de choisir un plus large compromis entre performances de classification et consommation. Cette reconfigurabilité peut également être exploitée pour le traitement d'autres types de signaux comme les électroencéphalogrammes (EEG) ou les signaux audio. Une implémentation d'un réseau de neurones analogique ou numérique peut également être envisagée comme évoqué dans les Sections 3.5 et 3.6.

### Sélection de paramètre pour les signaux EEG

La reconfigurabilité du convertisseur A2F peut également être exploitée pour traiter d'autres types de signaux comme les signaux électroencéphalogramme (EEG). L'électroencéphalographie est une méthode permettant de mesurer l'activité électrique du cerveau grâce à différentes électrodes placées sur le cuir chevelu du patient. Le tracé ainsi obtenu est appelé électroencéphalogramme (EEG). Nous allons considérer ici la base de donnée *CHB-MIT Scalp EEG* [130]. La Figure 4.1 représente une minute de signal EEG en indiquant la partie du signal à identifier

comme étant de l'épilepsie. L'étude réalisée ici est une étude préliminaire permettant de mettre en évidence les contraintes liées aux signaux EEG et pour évoquer certaines pistes pour les surmonter.

Cette base de données est composée de 23 enregistrements EEG de personnes souffrant de crises d'épilepsies. Tous les signaux ont été échantillonnés à une fréquence de 256 Hz, avec une résolution de 16 bits. Pour chaque patient l'enregistrement est réalisé sur plusieurs heures et est découpé sur plusieurs canaux, chacun caractérisant l'activité d'une région du cerveau comme illustré à la Figure 4.2. Chaque électrode est identifiée par un code : les différentes lettres permettent d'identifier le lobe ou la zone du cerveau dont le signal est issu. L'annotation indiquant la présence d'épilepsie est valable pour chacun des canaux disponibles, comme illustré à la Figure 4.2. Des signes d'épilepsie peuvent donc être détectés dans une région spécifique du cerveau et sera donc visibles seulement sur certains canaux.

Comme pour les signaux ECG, nous proposons de découper les signaux en blocs de 256 échantillons auxquels est appliqué le processus d'extraction de paramètres grâce aux ondelettes de Haar. Contrairement à l'extraction de paramètres pour les signaux ECG, les signaux EEG ne nécessitent pas de synchronisation, mais ils possèdent d'autres points d'attention.

Le problème que pose l'utilisation de cette base de donnée est la très faible occurrence des évènements à détecter : parmi les enregistrements utilisés pour la phase d'apprentissage, seulement 0.2% des exemples sont positifs. Ce déséquilibre dans le nombre d'exemples de chaque classe augmente considérablement la difficulté de l'apprentissage des modèles d'apprentissage automatique. Le modèle appris est biaisé par le nombre excessif d'évènements négatifs et est donc moins robuste. Pour faire face à ce déséquilibre, on peut pré-traiter l'ensemble de don-

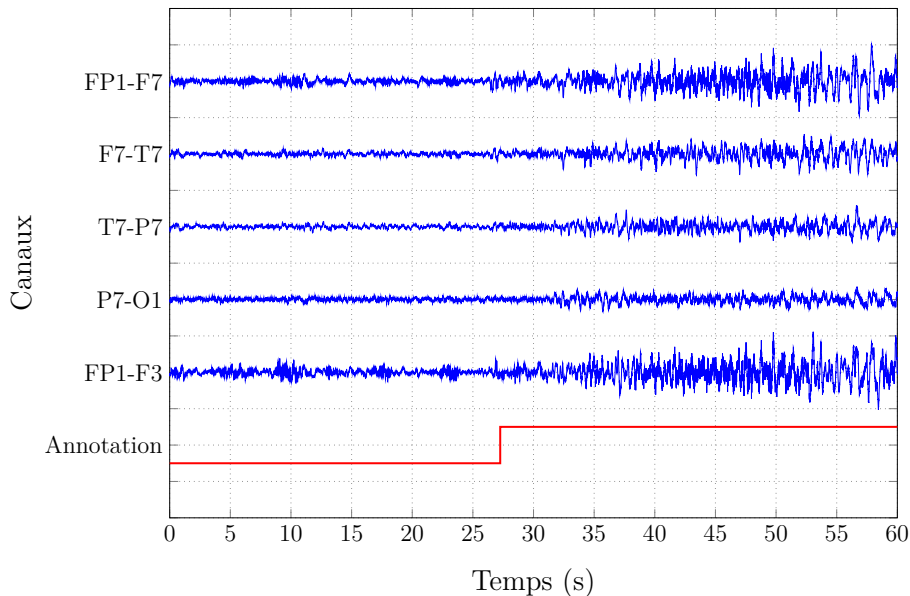


FIGURE 4.1 – Exemple de signal EEG avec l'annotation correspondante



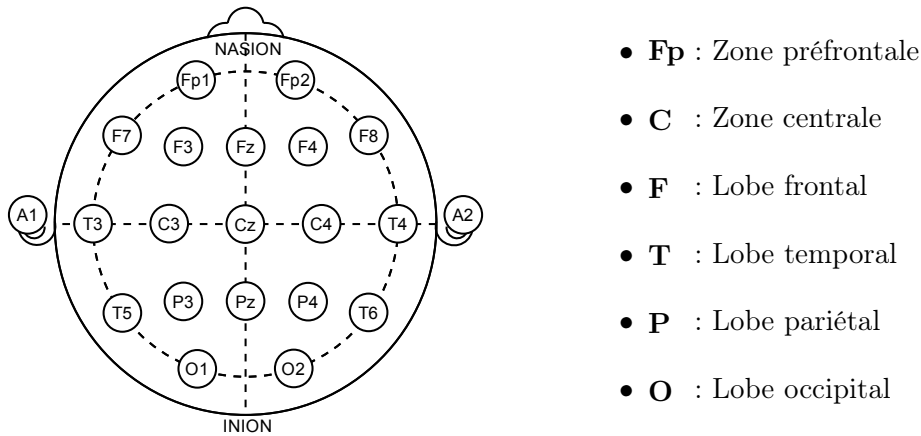


FIGURE 4.2 – Répartition des différentes électrodes au niveau du cuir chevelu. Source : [131]

née d’entraînement pour rééquilibrer le nombre d’exemples de chaque classe. Pour cela on peut utiliser une technique comme le sous-échantillonnage aléatoire, qui consiste à supprimer aléatoirement des exemples de la classe majoritaire. Cette méthode est utilisée ici car, étant donné la longueur importante des différents signaux (plusieurs heures pour chaque patient), le nombre d’exemples dans l’ensemble d’apprentissage reste important. Au contraire, l’ensemble de test ne subit aucun traitement et conserve sa répartition originale d’exemple de chaque classe.

La Figure 4.3 représente la précision de classification, en fonction du nombre de paramètres sélectionnés, obtenue pour les signaux EEG. La sélection des paramètres est réalisée grâce à l’algorithme SFS après un pré-filtrage de 100 paramètres avec le gain d’information et le score

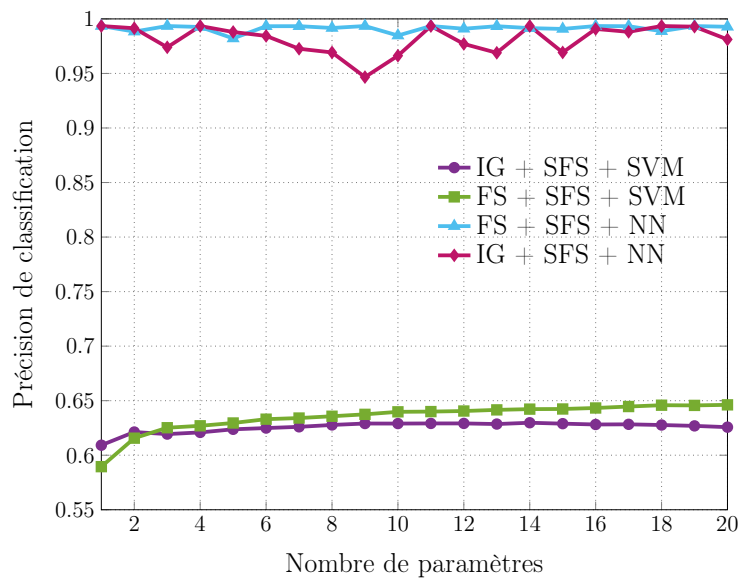


FIGURE 4.3 – Performances de classification pour la détection d’épisodes épileptiques dans des signaux EEG avec des ondelettes de Haar

de Fisher. La précision de classification est obtenue grâce à des classifieurs de type NN, qui ont montré de bons résultats pour les signaux ECG, et de type SVM utilisant un noyau basé sur une fonction de base radiale. Ces derniers ont été choisis, car ils ont permis d'obtenir de bonnes performances dans [1, 132]. L'apprentissage des classifieurs de type SVM étant fortement impacté par les ensembles de données contenant beaucoup d'exemples, l'ensemble d'entraînement a été limité au dix premiers patients de la base de donnée *CHB-MIT Scalp EEG*. L'ensemble de test est constitué avec les signaux de cinq autres patients. Les signaux EEG étant constitués de plusieurs canaux, l'extraction de paramètres est réalisée sur ces différents canaux : le nombre de paramètres obtenu pour une fenêtre temporelle d'analyse correspond ainsi au produit du nombre de canaux avec le nombre d'ondelettes dans la base choisie. Le nombre de paramètres considérés ici est donc de 11 546.

Dans la Figure 4.3, on constate ainsi que les SVM fournissent de très faibles performances, avec au maximum 64.5% de bonnes classifications, alors que la proportion d'exemples positifs est de 0.65%. Les NN permettent quant à eux, d'obtenir une précision maximum de 99.35%. Ce chiffre est à mettre en perspective par rapport à la faible proportion d'exemples positifs. Il est nécessaire de regarder d'autres critères comme la sensibilité et la spécificité. Dans le cas des signaux EEG, la sensibilité représente la probabilité que la prédiction soit positive s'il y a effectivement des signes d'épilepsie et la spécificité représente, au contraire, la probabilité que la prédiction soit négative si les signaux ne montrent pas de signes d'épilepsie. L'évaluation de ces deux critères dans le cas de NN, montrent une spécificité comprise entre 98.7 et 100%, mais une sensibilité maximum de 5%, comme illustré à la Figure 4.4. Ce qui traduit la difficulté, pour les NN appris, de détecter les épisodes épileptiques.

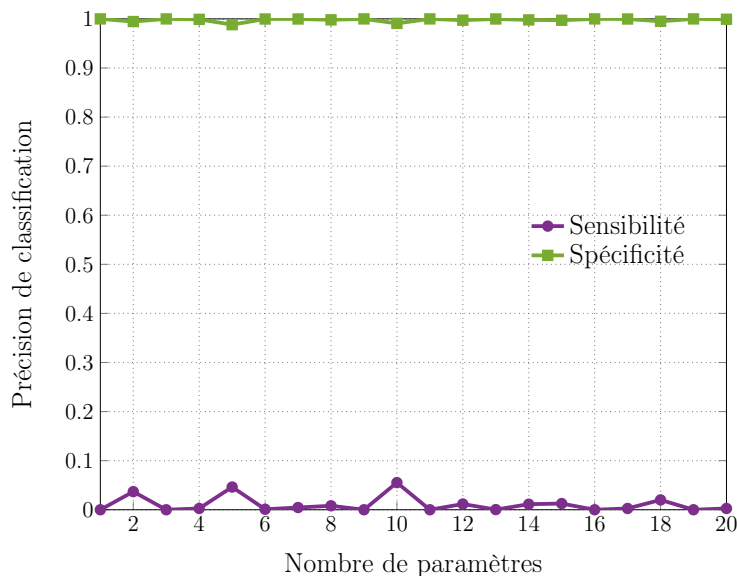


FIGURE 4.4 – Sensibilité et spécificité obtenues en utilisant des NN et avec une sélection par FS et SFS

Il est donc nécessaire d'utiliser d'autres modèles de classification ou d'appliquer d'autres méthodes que le sous-échantillonnage aléatoire et ainsi mieux distinguer la limite entre les deux classes pour améliorer la détection des épisodes épileptiques. En effet, la sélection aléatoire d'exemples de la classe majoritaire peut avoir pour conséquence de sélectionner des exemples dont l'appartenance à cette classe est floue et ainsi avoir pour conséquence de rendre la détection plus difficile. Pour cela, il existe différentes approches : soit en diminuant le nombre d'exemples de la classe majoritaire (sous-échantillonnage), soit en augmentant artificiellement le nombre d'exemples de la classe minoritaire (sur-échantillonnage). Parmi les méthodes de sous-échantillonnage, on peut citer la méthode des liens de Tomek [133, 134] qui permet de supprimer des exemples à la limite de deux (ou plusieurs) classes, ce qui permet de rendre la limite entre ces classes plus nette et donc d'améliorer les performances de classification.

De manière analogue, on trouve parmi les méthodes de sur-échantillonnage, une technique appelée sur-échantillonnage aléatoire qui consiste à compléter l'ensemble de données d'apprentissage en dupliquant des exemples de la classe minoritaire. Pour synthétiser des exemples de la classe minoritaire, on peut également utiliser une méthode appelée le sur-échantillonnage des minorités synthétiques (Synthetic Minority Oversampling Technique, SMOTE) [135]. Considérons un ensemble de données d'apprentissage dont les paramètres sont continus. L'algorithme SMOTE sélectionne un exemple de la classe minoritaire et recherche ses  $k$  plus proches voisins. Pour générer un nouvel exemple, le vecteur entre l'exemple considéré et l'un de ses  $k$  plus proches voisins est multiplié par un nombre aléatoire compris entre 0 et 1.

Une autre technique peut également être utilisée pour surmonter le problème de l'apprentissage avec des données inégalement réparties. Cette technique s'appuie sur le concept de réseaux de neurones complémentaires (Complementary Neural Network, CMTNN)[136, 135, 137]. Elle utilise donc deux réseaux de neurones, appelés *Truth Neural Network (TNN)* et *Falsity Neural Network (FNN)*, qui sont entraînés avec le même ensemble de données, mais le second réseau utilise les données de classe complémentaires de celle du premier réseau. Le réseau TNN est ainsi entraîné pour déterminer le degré d'appartenance à la classe "Vrai" et le réseau FNN est donc entraîné pour déterminer le degré d'appartenance à la classe "Faux". La méthode proposée dans [136] et [135], est donc de comparer, lors de la phase de test, les prédictions de ces deux réseaux et de supprimer les exemples qui ont été mal classés dans les deux cas. En effet, le degré d'appartenance à la classe "Vrai" est supposé être le complément du degré d'appartenance à la classe "Faux". Cependant, les deux valeurs d'appartenance prédites peuvent ne pas être complémentaires à 100%. Un flou peut apparaître dans la frontière entre ces deux appartenances : en éliminant ces exemples, la limite entre les deux classes est plus nette, ce qui permet d'améliorer les performances de classification. Les auteurs de [135] proposent de combiner la méthode des CMTNN avec l'algorithme SMOTE pour traiter les ensembles de données inégalement répartis pour améliorer les performances de classification.

## Amélioration de l'autonomie grâce aux techniques de récupération d'énergie

Pour augmenter l'autonomie des capteurs, on peut également s'intéresser aux systèmes de récupération d'énergie. L'intérêt pour ces systèmes est croissant notamment pour le développement des bio-capteurs implantables. En effet, ce genre de capteur nécessite une autonomie extrêmement longue du fait de leur positionnement. Différentes techniques de récupération d'énergie pour les bio-capteurs sont détaillées dans [138] comme les systèmes de récupération d'énergie cinétique basés, par exemple sur l'effet piézoélectrique, les systèmes utilisant la thermoélectricité, en exploitant par exemple l'effet Seebeck, les systèmes à base de pile à combustible ou encore les systèmes exploitant les rayonnements infrarouges.

Parmi les systèmes de récupération d'énergie adaptés aux systèmes biomédicaux, les systèmes thermoélectriques [139] sont particulièrement adaptés, car ils possèdent une forte densité de puissance. Une interface de récupération d'énergie thermoélectrique, réalisée dans une technologie CMOS 0.18  $\mu\text{m}$  est proposée dans [140]. En utilisant un générateur thermoélectrique *TGP-651* de la marque MICROPELT®<sup>®</sup>, le système permet de récupérer une puissance de 23  $\mu\text{W}$ , avec une différence de température de 2.5 K et avec une efficacité de 85%. Une autre technique particulièrement adaptée pour les bio-capteurs implantables, sont les systèmes à base de pile à combustible. La principale différence entre une pile à combustible et une pile traditionnelle réside dans le fait que la pile à combustible peut produire de l'énergie tant que les réactifs nécessaires sont présents. Les systèmes de récupération à base de pile à combustible représentent donc une bonne piste pour les bio-capteurs implantables, car ils peuvent exploiter des réactifs présents dans le corps humain, ce qui offre des avantages considérables. L'utilisation du glucose comme réactif est probablement l'une des pistes les plus étudiées, en raison de sa forte disponibilité. L'interface de récupération d'énergie, proposée dans [141], exploite deux sources d'énergie : une pile à combustible à base de glucose et un générateur thermoélectrique. Cette interface est réalisée dans une technologie CMOS 0.18  $\mu\text{m}$  et possède un rendement maximal de 89.5% lorsque les deux sources fournissent une puissance combinée de 66  $\mu\text{W}$ . Ce système de récupération d'énergie permettrait de récupérer une quantité d'énergie de 660  $\mu\text{J}$  sur une période de 10 secondes. Cette quantité d'énergie est donc supérieure aux données de consommation d'un capteur utilisant un convertisseur A2F obtenues dans la Section 3.5, où les deux solutions proposées consommaient respectivement 47.6  $\mu\text{J}$  et 47.6  $\mu\text{J}$  pour l'acquisition, le traitement et la transmission des données. Ce système de récupération d'énergie pourrait donc potentiellement donner une autonomie infinie à notre capteur. Ceci est tout de même à relativiser car, nous ne prenons pas en compte l'interfaçage entre le capteur et le système de récupération d'énergie, le stockage de l'énergie et la puissance instantanée requise pour le fonctionnement du capteur.

# Annexe A : Code MATLAB pour le SFS

```
1 function [selected,perf,classifier] = SFS(dataset,classe,datasetTest,
    classeTest,available,nMax)
2
3     selected = [];
4     perf = zeros(1,3);
5     classifier = cell(1,nMax);
6     while(length(selected)<nMax)
7         scores = zeros(1,length(available));
8         netList = cell(1,length(available));
9         parfor i=1:length(available)
10             subSet = dataset(:,[selected available(i)]);
11             subSetTest = datasetTest(:,[selected available(i)]);
12             for k = 1:50
13                 net = patternnet(10);
14                 net = configure(net,subSet',full(ind2vec(classe')));
15                 net.layers{1}.transferFcn = "poslin";
16                 net.layers{2}.transferFcn = "softmax";
17                 net = train(net,subSet',full(ind2vec(classe')));
18                 y = net(subSetTest');
19                 x = sum(vec2ind(y)==classeTest')/length(classeTest);
20                 if(x>scores(i))
21                     scores(i) = x;
22                     netList{i} = net;
23                 end
24             end
25         end
26         id = find(scores==max(scores),1);
27         selected(n) = available(id);
28         perf(n) = scores(id);
29         classifier{n} = netList{id};
30         available(id) = [];
31     end
32 end
```



# Annexe B : SFS avec limite d'extracteur

```
1 function [selected,perf,classifier,resultBranches] = SFS_opt(dataset,
    classe,datasetTest,classeTest,available,nMax,nBranches,marge,
    dictionnaire)
2
3     n = 1;
4     selected = [];
5     perf = zeros(1,3);
6     classifier = cell(1,nMax);
7     noSolution = false;
8     resultBranches = zeros(1,nMax);
9     sizeBranches = size(dictionnaire);
10
11     table = cell(length(available),3);
12     table(:,1) = num2cell(available);
13     table(:,2) = dictionnaire(available);
14
15     for m = 1:size(table,1)
16         wavelet = cell2mat(table(m,2));
17         nonZero = find(wavelet);
18         first = nonZero(1);
19         last = nonZero(end);
20         table{m,3} = [first last];
21     end
22
23     branches = zeros(nBranches,sizeBranches);
24
25     while(length(selected)<nMax && ~noSolution)
26         scores = zeros(size(table,1),1);
27         netList = cell(size(table,1),1);
28         parfor i=1:size(table,1)
29             subSet = dataset(:,[selected table{i,1}]);
30             subSetTest = datasetTest(:,[selected table{i,1}]);
31             for k = 1:50
32                 net = patternnet(10);
33                 net = configure(net,subSet',full(ind2vec(classe')));
34                 net.layers{1}.transferFcn = "poslin";
```

```

35         net.layers{2}.transferFcn = "softmax";
36         net = train(net,subSet',full(ind2vec(classe')));
37         y = net(subSetTest');
38         x = sum(vec2ind(y)==classeTest')/length(classeTest);
39         if(x>scores(i))
40             scores(i) = x;
41             netList{i} = net;
42         end
43     end
44 end
45
46 id = 0;
47 newTable = [num2cell(scores) table netList];
48 newTable = sortrows(newTable,1,'descend');
49
50 for m = 1:size(newTable,1)
51     limit = newTable{m,4};
52     first = limit(1);
53     last = limit(2);
54     for a = 1:nBranches
55         b = branches(a,:);
56         if (first-marge)>0 && (last+marge)<=sizeBranches
57             b = b(first-marge:last+marge);
58         elseif (first-marge)<=0 && (last+marge)<=sizeBranches
59             b = b(1:last+marge);
60         elseif (first-marge)>0 && (last+marge)>sizeBranches
61             b = b(first-marge:end);
62         else
63             b = b(1:end);
64         end
65         if(sum(b)==0)
66             branches(a,first:last)=1;
67             id = m;
68             resultBranches(n) = a;
69             break
70         end
71     end
72     if(id~=0)
73         break
74     end
75 end
76 if(id==0)
77     noSolution = true;
78 else
79     perf(n) = newTable{id,1};

```



```
80         selected(n) = newTable{id,2};
81         classfier{n} = newTable{id,5};
82         table(cell2mat(table(:,1))==newTable{id,2},:) = [];
83
84         forbidden = ones(1,sizeBranches);
85         for a = 1:nBranches
86             forbidden = forbidden & branches(a,:);
87         end
88
89         deleteWaves = zeros(1,size(table,1));
90         for m = 1:size(table,1)
91             if sum(forbidden(table{m,3}))>0
92                 deleteWaves(m) = 1;
93             end
94         end
95         table(logical(deleteWaves),:) = [];
96         n = n+1;
97     end
98 end
99 end
```



# Bibliographie

- [1] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Guttag, and A. P. Chandrakasan, “A Micro-Power EEG Acquisition SoC With Integrated Feature Extraction Processor for a Chronic Seizure Detection System,” *IEEE Journal of Solid-State Circuits*, vol. 45, no. 4, pp. 804–816, Apr. 2010.
- [2] E. J. Candes and M. B. Wakin, “An Introduction To Compressive Sampling,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
- [3] E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, “A Review of Sparse Recovery Algorithms,” *IEEE Access*, vol. 7, pp. 1300–1322, 2019.
- [4] D. Gangopadhyay, E. G. Allstot, A. M. R. Dixon, K. Natarajan, S. Gupta, and D. J. Allstot, “Compressed Sensing Analog Front-End for Bio-Sensor Applications,” *IEEE Journal of Solid-State Circuits*, vol. 49, no. 2, pp. 426–438, Feb. 2014. [Online]. Available : <https://ieeexplore.ieee.org/document/6722935>
- [5] B. Rumberg, D. W. Graham, V. Kulathumani, and R. Fernandez, “Hibernets : Energy-Efficient Sensor Networks Using Analog Signal Processing,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 3, pp. 321–334, Sep. 2011.
- [6] K. Badami, S. Lauwereins, W. Meert, and M. Verhelst, “24.2 Context-aware hierarchical information-sensing in a 6 $\mu$ W 90nm CMOS voice activity detector,” in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*. San Francisco, CA, USA : IEEE, Feb. 2015, pp. 1–3. [Online]. Available : <http://ieeexplore.ieee.org/document/7063110/>
- [7] K. M. H. Badami, S. Lauwereins, W. Meert, and M. Verhelst, “A 90 nm CMOS, 6  $\mu$ W Power-Proportional Acoustic Sensing Frontend for Voice Activity Detection,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 291–302, Jan. 2016. [Online]. Available : <https://ieeexplore.ieee.org/document/7315025/>
- [8] P. Chollet, “Traitement parcimonieux de signaux biologiques,” Thèse de doctorat, Ecole nationale supérieure Mines-Télécom Atlantique Bretagne Pays de la Loire, Nov. 2017. [Online]. Available : <http://www.theses.fr/2017IMTA0024>
- [9] J. Burrell, T. Brooke, and R. Beckwith, “Vineyard computing : Sensor networks in agricultural production,” *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, Jan. 2004.
- [10] D. Davcev, K. Mitreski, S. Trajkovic, V. Nikolovski, and N. Koteli, “IoT agriculture system based on LoRaWAN,” in *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*, Jun. 2018, pp. 1–4.
- [11] “On Dijon.” [Online]. Available : <http://www.metropole-dijon.fr/Grands-projets/Les-grandes-realisations/OnDijon-metropole-intelligente-et-connectee>

- [12] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava, "NAWMS : Nonintrusive autonomous water monitoring system," in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '08. New York, NY, USA : Association for Computing Machinery, Nov. 2008, pp. 309–322. [Online]. Available : <https://doi.org/10.1145/1460412.1460443>
- [13] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, "Hardware design experiences in ZebraNet," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, ser. SenSys '04. New York, NY, USA : Association for Computing Machinery, Nov. 2004, pp. 227–238. [Online]. Available : <https://doi.org/10.1145/1031495.1031522>
- [14] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," in *Proceedings of the Second European Workshop on Wireless Sensor Networks, 2005.*, Feb. 2005, pp. 108–120.
- [15] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, no. 2, pp. 18–25, Mar. 2006.
- [16] X. Lai, Q. Liu, X. Wei, W. Wang, G. Zhou, and G. Han, "A Survey of Body Sensor Networks," *Sensors*, vol. 13, no. 5, pp. 5406–5447, May 2013. [Online]. Available : <https://www.mdpi.com/1424-8220/13/5/5406>
- [17] C. E. Shannon, "Communication in the Presence of Noise," *PROCEEDINGS OF THE IEEE*, vol. 86, no. 2, p. 11, 1998.
- [18] V. H. Chen and L. Pileggi, "An 8.5mW 5GS/s 6b flash ADC with dynamic offset calibration in 32nm CMOS SOI," in *2013 Symposium on VLSI Circuits*, Jun. 2013, pp. C264–C265.
- [19] Y.-S. Shu, "A 6b 3GS/s 11mW fully dynamic flash ADC in 40nm CMOS with reduced number of comparators," in *2012 Symposium on VLSI Circuits (VLSIC)*, Jun. 2012, pp. 26–27.
- [20] S. Zhu, B. Wu, Y. Cai, and Y. Chiu, "A 2GS/s 8b flash ADC based on remainder number system in 65nm CMOS," in *2017 Symposium on VLSI Circuits*, Jun. 2017, pp. C284–C285.
- [21] D. C. Daly and A. P. Chandrakasan, "A 6b 0.2-to-0.9V Highly Digital Flash ADC with Comparator Redundancy," in *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, Feb. 2008, pp. 554–635.
- [22] G. Van der Plas, S. Decoutere, and S. Donnay, "A 0.16pJ/Conversion-Step 2.5mW 1.25GS/s 4b ADC in a 90nm Digital CMOS Process," in *2006 IEEE International Solid State Circuits Conference - Digest of Technical Papers*, Feb. 2006, pp. 2310–.
- [23] B. Murmann, "ADC Performance Survey 1997-2019." [Online]. Available : <http://web.stanford.edu/~murmam/adcsurvey.html>

- [24] B. Verbruggen, M. Iriguchi, M. de la Guia Solaz, G. Glorieux, K. Deguchi, B. Malki, and J. Craninckx, "A 2.1 mW 11b 410 MS/s dynamic pipelined SAR ADC with background calibration in 28nm digital CMOS," in *2013 Symposium on VLSI Circuits*, Jun. 2013, pp. C268–C269.
- [25] B. Hershberg, B. v Liempd, N. Markulic, J. Lagos, E. Martens, D. Dermit, and J. Craninckx, "3.6 A 6-to-600MS/s Fully Dynamic Ringamp Pipelined ADC with Asynchronous Event-Driven Clocking in 16nm," in *2019 IEEE International Solid-State Circuits Conference - (ISSCC)*, Feb. 2019, pp. 68–70.
- [26] B. Verbruggen, M. Iriguchi, and J. Craninckx, "A 1.7 mW 11b 250 MS/s 2-Times Interleaved Fully Dynamic Pipelined SAR ADC in 40 nm Digital CMOS," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 12, pp. 2880–2887, Dec. 2012.
- [27] K.-J. Moon, H.-W. Kang, D.-S. Jo, M.-Y. Kim, S.-Y. Baek, M. Choi, H.-J. Ko, and S.-T. Ryu, "A 9.1 ENOB 21.7fJ/conversion-step 10b 500MS/s single-channel pipelined SAR ADC with a current-mode fine ADC in 28nm CMOS," in *2017 Symposium on VLSI Circuits*, Jun. 2017, pp. C94–C95.
- [28] B. Hershberg and U.-K. Moon, "A 75.9dB-SNDR 2.96mW 29fJ/conv-step ringamp-only pipelined ADC," in *2013 Symposium on VLSI Circuits*, Jun. 2013, pp. C94–C95.
- [29] H. S. Bindra, A.-J. Annema, S. M. Louwsma, and B. Nauta, "A 0.2 - 8 MS/s 10b flexible SAR ADC achieving 0.35 - 2.5 fJ/conv-step and using self-quenched dynamic bias comparator," in *2019 Symposium on VLSI Circuits*, Jun. 2019, pp. C74–C75.
- [30] S.-E. Hsieh and C.-C. Hsieh, "A 0.44-fJ/Conversion-Step 11-Bit 600-kS/s SAR ADC With Semi-Resting DAC," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 9, pp. 2595–2603, Sep. 2018.
- [31] —, "A 0.4V 13b 270kS/S SAR-ISDM ADC with an opamp-less time-domain integrator," in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb. 2018, pp. 240–242.
- [32] H.-Y. Tai, Y.-S. Hu, H.-W. Chen, and H.-S. Chen, "11.2 A 0.85fJ/conversion-step 10b 200kS/s subranging SAR ADC in 40nm CMOS," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb. 2014, pp. 196–197.
- [33] Y.-S. Hu, K.-Y. Lin, and H.-S. Chen, "A 510nW 12-bit 200kS/s SAR-assisted SAR ADC using a re-switching technique," in *2017 Symposium on VLSI Circuits*, Jun. 2017, pp. C238–C239.
- [34] P. Cenci, M. Bolatkale, R. Rutten, M. Ganzerli, G. Lassche, K. Makinwa, and L. Breems, "A 3.2mW SAR-assisted CT $\Delta\Sigma$  ADC with 77.5dB SNDR and 40MHz BW in 28nm CMOS," in *2019 Symposium on VLSI Circuits*, Jun. 2019, pp. C230–C231.
- [35] H. Chandrakumar and D. Markovic, "A 15.2-enob continuous-time  $\Delta\Sigma$  adc for a 7.3  $\mu$ W 200mvpp-linear-input-range neural recording front-end," in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb. 2018, pp. 232–234.

- [36] J. Liu, S. Li, W. Guo, G. Wen, and N. Sun, "A 0.029MM<sup>2</sup> 17-FJ/Conv.-Step CT  $\Delta\Sigma$  ADC with 2nd-Order Noise-Shaping SAR Quantizer," in *2018 IEEE Symposium on VLSI Circuits*, Jun. 2018, pp. 201–202.
- [37] C.-H. Weng, T.-A. Wei, H.-Y. Hsieh, S.-H. Wu, and T.-Y. Wang, "A 71.4dB SNDR 30MHz BW Continuous-Time Delta-sigma Modulator Using a Time-Interleaved Noise-Shaping Quantizer in 12-nm CMOS," in *2019 Symposium on VLSI Circuits*, Jun. 2019, pp. C228–C229.
- [38] T.-Y. Lo, C.-H. Weng, H.-Y. Hsieh, Y.-S. Shu, and P.-C. Chiu, "20.4 An  $8 \times$  - OSR 25MHz-BW 79.4dB/74dB DR/SNDR CT  $\Delta\sigma$  Modulator Using 7b Linearized Segmented DACs with Digital Noise-Coupling-Compensation Filter in 7nm FinFET CMOS," in *2019 IEEE International Solid-State Circuits Conference - (ISSCC)*, Feb. 2019, pp. 334–336.
- [39] "Bluetooth Core, Specification Version 5.2, Feature Overview."
- [40] "IEEE Standard for Low-Rate Wireless Networks," Jul. 2020.
- [41] "Zigbee Alliance." [Online]. Available : <https://zigbeealliance.org/>
- [42] F.-W. Kuo, S. Binsfeld Ferreira, H.-N. R. Chen, L.-C. Cho, C.-P. Jou, F.-L. Hsueh, I. Madadi, M. Tohidian, M. Shahmohammadi, M. Babaie, and R. B. Staszewski, "A Bluetooth Low-Energy Transceiver With 3.7-mW All-Digital Transmitter, 2.75-mW High-IF Discrete-Time Receiver, and TX/RX Switchable On-Chip Matching Network," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 4, pp. 1144–1162, Apr. 2017. [Online]. Available : <https://ieeexplore.ieee.org/document/7862859>
- [43] S. Oh, S. Kim, I. Ali, T. T. K. Nga, D. Lee, Y. Pu, S. Yoo, M. Lee, K. C. Hwang, Y. Yang, and K. Lee, "A 3.9 mW Bluetooth Low-Energy Transmitter Using All-Digital PLL-Based Direct FSK Modulation in 55 nm CMOS," *IEEE Transactions on Circuits and Systems I : Regular Papers*, vol. 65, no. 9, pp. 3037–3048, Sep. 2018.
- [44] H. Liu, Z. Sun, D. Tang, H. Huang, T. Kaneko, Z. Chen, W. Deng, R. Wu, and K. Okada, "A DPLL-Centric Bluetooth Low-Energy Transceiver With a 2.3-mW Interference-Tolerant Hybrid-Loop Receiver in 65-nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 12, pp. 3672–3687, Dec. 2018.
- [45] L.-H. Wang, T.-Y. Chen, K.-H. Lin, Q. Fang, and S.-Y. Lee, "Implementation of a wireless ecg acquisition soc for ieee 802.15.4 (zigbee) applications," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 247–255, 2015.
- [46] B. J. Thijssen, E. A. M. Klumperink, P. Quinlan, and B. Nauta, "30.4 a 370 $\mu$ w 5.5db-nf ble/bt5.0/ieee 802.15.4-compliant receiver with gt;63db adjacent channel rejection at gt;2 channels offset in 22nm fdsoi," in *2020 IEEE International Solid-State Circuits Conference - (ISSCC)*, 2020, pp. 466–468.
- [47] M. Rahman and R. Harjani, "A 2.4GHz IEEE 802.15.6 Compliant 1.52nJ/bit TX 1.32nJ/bit RX Multiband Transceiver for Low Power Standards," in *2018 25th IEEE*

- International Conference on Electronics, Circuits and Systems (ICECS)*, Dec. 2018, pp. 821–824.
- [48] A. Ba, M. Vidojkovic, K. Kanda, N. F. Kiyani, M. Lont, X. Huang, X. Wang, C. Zhou, Y. Liu, M. Ding, B. Büsze, S. Masui, M. Hamaminato, H. Sato, K. Philips, and H. de Groot, “A 0.33 nJ/bit IEEE802.15.6/Proprietary MICS/ISM Wireless Transceiver With Scalable Data Rate for Medical Implantable Applications,” *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 3, pp. 920–929, May 2015.
- [49] Y. L. Tsou, C. A. Gong, N. C. Cheng, Y. Lee, and C. F. Jou, “Integrated Biosensing Platform Based on a 1.74-mW -90-dBm Sensitivity Dual- Mode-Operation Receiver for IEEE 802.15.6 Human Body Communication Standard,” *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3317–3327, Jun. 2015.
- [50] J. Bae, K. Song, H. Lee, H. Cho, and H. Yoo, “A 0.24-nJ/b Wireless Body-Area-Network Transceiver With Scalable Double-FSK Modulation,” *IEEE Journal of Solid-State Circuits*, vol. 47, no. 1, pp. 310–322, Jan. 2012.
- [51] H. Cho, H. Kim, M. Kim, J. Jang, Y. Lee, K. J. Lee, J. Bae, and H. Yoo, “A 79 pJ/b 80 Mb/s Full-Duplex Transceiver and a 42.5  $\mu$ W 100 kb/s Super-Regenerative Transceiver for Body Channel Communication,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 310–317, Jan. 2016.
- [52] “IEEE Standard for Local and metropolitan area networks - Part 15.6 : Wireless Body Area Networks,” Feb. 2012.
- [53] E. J. Candès, “The restricted isometry property and its implications for compressed sensing,” *Comptes Rendus Mathématique*, vol. 346, no. 9, pp. 589–592, May 2008. [Online]. Available : <http://www.sciencedirect.com/science/article/pii/S1631073X08000964>
- [54] A. Kundu and P. K. Roy, “Sparse Signal Recovery from Nonadaptive Linear Measurements,” *arXiv e-prints*, p. 5, Oct. 2013. [Online]. Available : <https://ui.adsabs.harvard.edu/abs/2013arXiv1310.8468K>
- [55] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, “A Simple Proof of the Restricted Isometry Property for Random Matrices,” *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, Dec. 2008. [Online]. Available : <http://link.springer.com/10.1007/s00365-007-9003-x>
- [56] E. Candès and J. Romberg, “Sparsity and incoherence in compressive sampling,” *Inverse Problems*, vol. 23, no. 3, pp. 969–985, Jun. 2007. [Online]. Available : <http://stacks.iop.org/0266-5611/23/i=3/a=008?key=crossref.bdb1d0e3390f5616a4395b50273198a9>
- [57] M. Pelissier and C. Studer, “Non-Uniform Wavelet Sampling for RF Analog-to-Information Conversion,” *IEEE Transactions on Circuits and Systems I : Regular Papers*, vol. 65, no. 2, pp. 471–484, Feb. 2018.
- [58] M. Wakin, S. Becker, E. Nakamura, M. Grant, E. Sovero, D. Ching, J. Yoo, J. Romberg, A. Emami-Neyestanak, and E. Candes, “A Nonuniform Sampler for Wideband

- Spectrally-Sparse Environments,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 3, pp. 516–529, Sep. 2012.
- [59] D. E. Bellasi, L. Bettini, C. Benkeser, T. Burger, Q. Huang, and C. Studer, “VLSI Design of a Monolithic Compressive-Sensing Wideband Analog-to-Information Converter,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 3, no. 4, pp. 552–565, Dec. 2013.
- [60] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk, “Analog-to-Information Conversion via Random Demodulation,” in *2006 IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software*, Oct. 2006, pp. 71–74.
- [61] T. Ragheb, J. N. Laska, H. Nejati, S. Kirolos, R. G. Baraniuk, and Y. Massoud, “A prototype hardware for random demodulation based compressive analog-to-digital conversion,” in *2008 51st Midwest Symposium on Circuits and Systems*. Knoxville, TN, USA : IEEE, Aug. 2008, pp. 37–40. [Online]. Available : <http://ieeexplore.ieee.org/document/4616730/>
- [62] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, “Theory and Implementation of an Analog-to-Information Converter using Random Demodulation,” in *2007 IEEE International Symposium on Circuits and Systems*. New Orleans, LA, USA : IEEE, May 2007, pp. 1959–1962. [Online]. Available : <http://ieeexplore.ieee.org/document/4253049/>
- [63] J. Yoo, S. Becker, M. Monge, M. Loh, E. Candes, and A. Emami-Neyestanak, “Design and implementation of a fully integrated compressed-sensing signal acquisition system,” in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Kyoto, Japan : IEEE, Mar. 2012, pp. 5325–5328. [Online]. Available : <http://ieeexplore.ieee.org/document/6289123/>
- [64] M. Mishali, Y. C. Eldar, and A. J. Elron, “Xampling : Signal Acquisition and Processing in Union of Subspaces,” *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4719–4734, Oct. 2011.
- [65] M. A. Lexa, M. E. Davies, and J. S. Thompson, “Reconciling Compressive Sampling Systems for Spectrally Sparse Continuous-Time Signals,” *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 155–171, Jan. 2012. [Online]. Available : <https://ieeexplore.ieee.org/document/6026970>
- [66] M. Mishali and Y. C. Eldar, “From Theory to Practice : Sub-Nyquist Sampling of Sparse Wideband Analog Signals,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 375–391, Apr. 2010, comment : 17 pages, 12 figures, to appear in IEEE Journal of Selected Topics in Signal Processing, the special issue on Compressed Sensing. [Online]. Available : <http://arxiv.org/abs/0902.4291.pdf>



- [67] P. Daponte, L. D. Vito, F. Picariello, I. Tudosa, and S. Rapuano, "Assessment of analog-to-information converters based on non-uniform sampling for spectrum analysis," in *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2018, pp. 1–6.
- [68] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic Decomposition by Basis Pursuit," *Society for Industrial and Applied Mathematics*, vol. 43, no. 1, p. 31, 2001. [Online]. Available : <https://web.stanford.edu/group/SOL/papers/BasisPursuit-SIGEST.pdf>
- [69] E. van den Berg and M. P. Friedlander, "Probing the Pareto Frontier for Basis Pursuit Solutions," *SIAM Journal on Scientific Computing*, vol. 31, no. 2, pp. 890–912, Jan. 2009. [Online]. Available : <http://epubs.siam.org/doi/10.1137/080714488>
- [70] E. Candes, J. Romberg, and T. Tao, "Stable Signal Recovery from Incomplete and Inaccurate Measurements," *arXiv :math/0503066*, Dec. 2005, comment : 15 pages, 11 figures. [Online]. Available : <http://arxiv.org/abs/math/0503066>
- [71] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993. [Online]. Available : <https://www.di.ens.fr/~mallat/papiers/MallatPursuit93.pdf>
- [72] P. J. Huber, "Projection Pursuit," *The Annals of Statistics*, vol. 13, no. 2, pp. 435–475, 1985. [Online]. Available : <https://www.jstor.org/stable/2241175>
- [73] S. Qaisar, R. M. Bilal, W. Iqbal, M. Naureen, and S. Lee, "Compressive sensing : From theory to applications, a survey," *Journal of Communications and Networks*, vol. 15, no. 5, pp. 443–456, Oct. 2013.
- [74] D. Needell and R. Vershynin, "Uniform Uncertainty Principle and Signal Recovery via Regularized Orthogonal Matching Pursuit," *Foundations of Computational Mathematics*, vol. 9, no. 3, pp. 317–334, Jun. 2009. [Online]. Available : <https://doi.org/10.1007/s10208-008-9031-3>
- [75] D. Needell and J. A. Tropp, "CoSaMP : Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, May 2009. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1063520308000638>
- [76] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin, "Algorithmic linear dimension reduction in the  $l_1$  norm for sparse vectors," *arXiv :cs/0608079*, Aug. 2006. [Online]. Available : <http://arxiv.org/abs/cs/0608079>
- [77] W. Dai and O. Milenkovic, "Subspace Pursuit for Compressive Sensing Signal Reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.
- [78] Y. Gargouri, H. Petit, P. Loumeau, B. Cecconi, and P. Desgreys, "Analog-to-information converter design for low-power acquisition of astrophysical signals," in *2017 15th IEEE*

- International New Circuits and Systems Conference (NEWCAS)*, Jun. 2017, pp. 113–116.
- [79] F. Pareschi, P. Albertini, G. Frattini, M. Mangia, R. Rovatti, and G. Setti, “Hardware-algorithms co-design and implementation of an analog-to-information converter for bio-signals based on compressed sensing,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 10, no. 1, pp. 149–162, 2016.
- [80] H. Bhamra, J. Lynch, M. Ward, and P. Irazoqui, “A Noise-Power-Area Optimized Bio-sensing Front End for Wireless Body Sensor Nodes and Medical Implantable Devices,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2917–2928, Oct. 2017.
- [81] T. Saba, “Recent advancement in cancer detection using machine learning : Systematic survey of decades, comparisons and challenges,” *Journal of Infection and Public Health*, vol. 13, no. 9, pp. 1274–1289, Sep. 2020. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1876034120305633>
- [82] V. Wasule and P. Sonar, “Classification of brain MRI using SVM and KNN classifier,” in *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, May 2017, pp. 218–223.
- [83] S. Mekruksavanich and A. Jitpattanakul, “Exercise Activity Recognition with Surface Electromyography Sensor using Machine Learning Approach,” in *2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT NCON)*, Mar. 2020, pp. 75–78.
- [84] S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio, “A smartphone-based fall detection system,” *Pervasive and Mobile Computing*, vol. 8, no. 6, pp. 883–899, Dec. 2012. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1574119212000983>
- [85] S.-K. Lin, Y.-S. Lin, C.-Y. Lin, and H. Chiueh, “A smart headband for epileptic seizure detection,” in *2017 IEEE Healthcare Innovations and Point of Care Technologies (HI-POCT)*, Nov. 2017, pp. 221–224.
- [86] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986. [Online]. Available : <http://link.springer.com/10.1007/BF00116251>
- [87] T. M. Mitchell, *Machine Learning*, ser. McGraw-Hill Series in Computer Science. New York : McGraw-Hill, 1997.
- [88] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001. [Online]. Available : <https://doi.org/10.1023/A:1010933404324>
- [89] A. Cutler, D. R. Cutler, and J. R. Stevens, “Random Forests,” in *Ensemble Machine Learning : Methods and Applications*, C. Zhang and Y. Ma, Eds.

- Boston, MA : Springer US, 2012, pp. 157–175. [Online]. Available : [https://doi.org/10.1007/978-1-4419-9326-7\\_5](https://doi.org/10.1007/978-1-4419-9326-7_5)
- [90] I. T. Brain and F. Rosenblatt, “The Perceptron : A Probabilistic Model for Information Storage and Organization.”
- [91] S. B. Kotsiantis, “Supervised Machine Learning : A Review of Classification Techniques,” *Informatica*, vol. 31, pp. 249–268, 2007.
- [92] B. Schölkopf, A. J. Smola, and F. Bach, *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.
- [93] D. W. H. Jr, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. John Wiley & Sons, Apr. 2013.
- [94] M. Verhelst and A. Bahai, “Where Analog Meets Digital : Analog to Information Conversion and Beyond,” *IEEE Solid-State Circuits Magazine*, vol. 7, no. 3, pp. 67–80, 2015.
- [95] X. Liu, E. Gönültaş, and C. Studer, “Analog-to-Feature (A2F) Conversion for Audio-Event Classification,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, Sep. 2018, pp. 2275–2279.
- [96] MesserWoland, “English : Diagram of the human heart,” Nov. 2006. [Online]. Available : [https://commons.wikimedia.org/wiki/File:Diagram\\_of\\_the\\_human\\_heart\\_\(multilingual\).svg](https://commons.wikimedia.org/wiki/File:Diagram_of_the_human_heart_(multilingual).svg)
- [97] E. J. d. S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti, “ECG-based heartbeat classification for arrhythmia detection : A survey,” *Computer Methods and Programs in Biomedicine*, vol. 127, pp. 144–164, Apr. 2016. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0169260715003314>
- [98] R. G. Kumar and Y. Kumaraswamy, “Investigation and classification of ecg beat using input output additional weighted feed forward neural network,” in *2013 International Conference on Signal Processing, Image Processing & Pattern Recognition*. IEEE, 2013, pp. 200–205.
- [99] G. D. Clifford, F. Azuaje, P. McSharry *et al.*, *Advanced methods and tools for ECG data analysis*. Artech house Boston, 2006.
- [100] C.-C. Lin and C.-M. Yang, “Heartbeat classification using normalized rr intervals and morphological features,” *Mathematical Problems in Engineering*, vol. 2014, 2014.
- [101] G. Doquire, G. De Lannoy, D. François, and M. Verleysen, “Feature selection for interpatient supervised heart beat classification,” *Computational intelligence and neuroscience*, vol. 2011, 2011.
- [102] P. De Chazal, M. O’Dwyer, and R. B. Reilly, “Automatic classification of heartbeats using ecg morphology and heartbeat interval features,” *IEEE transactions on biomedical engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.

- [103] G. B. Moody and R. G. Mark, "MIT-BIH Arrhythmia Database," 1992. [Online]. Available : <https://physionet.org/content/mitdb/>
- [104] J. Morlet, G. Arens, E. Fourgeau, and D. Glard, "Wave propagation and sampling theory—Part I : Complex signal and scattering in multilayered media," *GEOPHYSICS*, vol. 47, no. 2, pp. 203–221, Feb. 1982.
- [105] S. Mallat, *Une exploration des signaux en ondelettes*. Editions Ecole Polytechnique, 2000.
- [106] W. Heisenberg, "Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik," *Zeitschrift für Physik*, vol. 43, no. 3, pp. 172–198, Mar. 1927. [Online]. Available : <https://doi.org/10.1007/BF01397280>
- [107] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," *Mathematische Annalen*, vol. 69, no. 3, pp. 331–371, Sep. 1910. [Online]. Available : <https://doi.org/10.1007/BF01456326>
- [108] J. Tang, S. Alelyani, and H. Liu, "Feature Selection for Classification : A Review," *Data classification : algorithms and applications*, p. 33, 2014.
- [109] M. Robnik-Sikonja and I. Kononenko, "Theoretical and Empirical Analysis of ReliefF and RReliefF," *Machine Learning Journal*, 2003. [Online]. Available : <http://lkm.fri.uni-lj.si/rmarko/papers/robnik03-mlj.pdf>
- [110] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler, "Comparative study of techniques for large-scale feature selection," in *Machine Intelligence and Pattern Recognition*, ser. Pattern Recognition in Practice IV, E. S. Gelsema and L. S. Kanal, Eds. North-Holland, Jan. 1994, vol. 16, pp. 403–413. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/B9780444818928500407>
- [111] J. Yang and V. Honavar, "Feature Subset Selection Using a Genetic Algorithm," in *Feature Extraction, Construction and Selection : A Data Mining Perspective*, 1998, p. 15.
- [112] X. Zhang and Y. Lian, "A 300-mV 220-nW Event-Driven ADC With Real-Time QRS Detection for Wearable ECG Sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 6, pp. 834–843, Dec. 2014.
- [113] N. Ravanshad, H. Rezaee-Dehsorkh, R. Lotfi, and Y. Lian, "A Level-Crossing Based QRS-Detection Algorithm for Wearable ECG Sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 1, pp. 183–192, Jan. 2014.
- [114] F. Morshedlou, N. Ravanshad, and H. Rezaee-Dehsorkh, "An ultra-low power analog QRS-detection circuit for ambulatory ECG monitoring," *AEU - International Journal of Electronics and Communications*, vol. 129, p. 153551, Feb. 2021. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1434841120327552>
- [115] K. Benmouiza and A. Cheknane, "Forecasting hourly global solar radiation using hybrid k-means and nonlinear autoregressive neural network models," *Energy*

- Conversion and Management*, vol. 75, pp. 561–569, Nov. 2013. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0196890413003749>
- [116] J. Pérez-Bailón, B. Calvo, N. Medrano, and J. Ramírez-Angulo, “A 1.8 V Gm-C Highly Tunable Low Pass Filter for Sensing Applications,” in *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2020, pp. 1–5. [Online]. Available : <https://ieeexplore.ieee.org/document/9128692/>
- [117] J. Pérez-Bailón, A. Márquez, B. Calvo, and N. Medrano, “A 0.18 $\mu$ m CMOS Widely Tunable Low Pass Filter with sub-Hz Cutoff Frequencies,” in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2018, pp. 1–4. [Online]. Available : <https://ieeexplore.ieee.org/document/8351166>
- [118] E. Rodriguez-Villegas, A. J. Casson, and P. Corbishley, “A Subhertz Nanopower Low-Pass Filter,” *IEEE Transactions on Circuits and Systems II : Express Briefs*, vol. 58, no. 6, pp. 351–355, Jun. 2011. [Online]. Available : [ieeexplore.ieee.org/document/5930351/](https://ieeexplore.ieee.org/document/5930351/)
- [119] Q. Wu, M. Pedram, and X. Wu, “Clock-gating and its application to low power design of sequential circuits,” *IEEE Transactions on Circuits and Systems I : Fundamental Theory and Applications*, vol. 47, no. 3, pp. 415–420, Mar. 2000.
- [120] T. Chindhu S. and N. Shanmugasundaram, “Clock Gating Techniques : An Overview,” in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, Mar. 2018, pp. 217–221.
- [121] N. B. Gaikwad, V. Tiwari, A. Keskar, and N. C. Shivaprakash, “Efficient FPGA Implementation of Multilayer Perceptron for Real-Time Human Activity Classification,” *IEEE Access*, vol. 7, pp. 26 696–26 706, 2019.
- [122] “Artix-7 FPGA Family.” [Online]. Available : <https://www.xilinx.com/products/silicon-devices/fpga/artix-7.html>
- [123] A. Bahr, M. Schneider, M. A. Francis, H. M. Lehmann, I. Barg, A.-S. Buschhoff, P. Wulff, T. Strunskus, and F. Faupel, “Epileptic Seizure Detection on an Ultra-Low-Power Embedded RISC-V Processor Using a Convolutional Neural Network,” *Biosensors*, vol. 11, no. 7, p. 203, Jul. 2021. [Online]. Available : <https://www.mdpi.com/2079-6374/11/7/203>
- [124] B. Moons, D. Bankman, L. Yang, B. Murmann, and M. Verhelst, “BinarEye : An always-on energy-accuracy-scalable binary CNN processor with all memory on chip in 28nm CMOS,” in *2018 IEEE Custom Integrated Circuits Conference (CICC)*, Apr. 2018, pp. 1–4.
- [125] O. Krestinskaya, K. N. Salama, and A. P. James, “Analog Backpropagation Learning Circuits for Memristive Crossbar Neural Networks,” in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2018, pp. 1–5.

- [126] D. Soudry, D. Di Castro, A. Gal, A. Kolodny, and S. Kvatinsky, "Memristor-Based Multilayer Neural Networks With Online Gradient Descent Training," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2408–2421, Oct. 2015.
- [127] O. Krestinskaya, K. N. Salama, and A. P. James, "Learning in Memristive Neural Network Architectures Using Analog Backpropagation Circuits," *IEEE Transactions on Circuits and Systems I : Regular Papers*, vol. 66, no. 2, pp. 719–732, Feb. 2019.
- [128] L. Chua, "Memristor-The missing circuit element," *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, Sep. 1971.
- [129] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, May 2008.
- [130] A. H. Shoeb, "Application of machine learning to epileptic seizure onset detection and treatment," Thesis, Massachusetts Institute of Technology, 2009. [Online]. Available : <https://dspace.mit.edu/handle/1721.1/54669>
- [131] "Electrode locations of International 10-20 system for EEG (electroencephalography) recording," May 2010. [Online]. Available : [https://commons.wikimedia.org/wiki/File:21\\_electrodes\\_of\\_International\\_10-20\\_system\\_for\\_EEG.svg?uselang=fr](https://commons.wikimedia.org/wiki/File:21_electrodes_of_International_10-20_system_for_EEG.svg?uselang=fr)
- [132] A. Shoeb, B. Bourgeois, S. T. Treves, S. C. Schachter, and J. Guttag, "Impact of Patient-Specificity on Seizure Onset Detection Performance," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2007, pp. 4110–4114.
- [133] I. Tomek, "Two Modifications of CNN," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-6, no. 11, pp. 769–772, Nov. 1976.
- [134] M. Zeng, B. Zou, F. Wei, X. Liu, and L. Wang, "Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data," in *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*, May 2016, pp. 225–228.
- [135] P. Jeatrakul, K. W. Wong, and C. C. Fung, "Classification of Imbalanced Data by Combining the Complementary Neural Network and SMOTE Algorithm," in *Neural Information Processing. Models and Applications*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, 2010, vol. 17th International Conference, ICONIP 2010, Sydney, Australia, November 22-25, 2010, Proceedings, Part II, pp. 152–159. [Online]. Available : <https://link.springer.com/content/pdf/10.1007%2F978-3-642-17534-3.pdf>
- [136] P. Kraipeerapun and C. C. Fung, "Binary classification using ensemble neural networks and interval neutrosophic sets," *Neurocomputing*, vol. 72, no. 13, pp. 2845–2856, Aug. 2009. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0925231209000940>

- [137] P. Jeatrakul, K. W. Wong, and C. C. Fung, “Data cleaning for classification using misclassification analysis,” *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 14, no. 3, pp. 297–302, 2010. [Online]. Available : <https://doi.org/10.20965/jaciii.2010.p0297>
- [138] J. Olivo, S. Carrara, and G. De Micheli, “Energy Harvesting and Remote Powering for Implantable Biosensors,” *IEEE Sensors Journal*, vol. 11, no. 7, pp. 1573–1586, Jul. 2011.
- [139] V. Leonov, T. Torfs, P. Fiorini, and C. Van Hoof, “Thermoelectric Converters of Human Warmth for Self-Powered Wireless Sensor Nodes,” *IEEE Sensors Journal*, vol. 7, no. 5, pp. 650–657, May 2007.
- [140] J. Katic, S. Rodriguez, and A. Rusu, “A Dual-Output Thermoelectric Energy Harvesting Interface With 86.6% Peak Efficiency at 30  $\mu$ W and Total Control Power of 160 nW,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 8, pp. 1928–1937, Aug. 2016.
- [141] —, “A High-Efficiency Energy Harvesting Interface for Implanted Biofuel Cell and Thermal Harvesters,” *IEEE Transactions on Power Electronics*, vol. 33, no. 5, pp. 4125–4134, May 2018.

**Titre :** Conception et intégration d'un convertisseur analogique-paramètre flexible pour les capteurs intelligents

**Mots clés :** Convertisseur analogique-paramètre, Sélection de paramètres, Apprentissage automatique, Échantillonnage non-uniforme en ondelettes, Capteurs intelligents.

**Résumé :**

Avec le fort développement de l'Internet des Objets (IoT), il devient nécessaire de converger vers de nouveaux capteurs dit intelligents. Ces capteurs doivent permettre d'analyser l'environnement extérieur, comprendre le contexte dans lequel ils sont utilisés et être conscient des besoins utilisateurs. Ils doivent cependant rester petits, fiables, bon marché et avoir une autonomie de plusieurs années. La conversion analogique-paramètre (Analog-to-Feature, A2F) est une nouvelle méthode d'acquisition pensée pour les appareils IoT, et semble être une solution adaptée pour de tels capteurs. Cette conversion consiste à extraire des paramètres directement sur le signal analogique. Une sélection pertinente des paramètres permet d'extraire uniquement l'information nécessaire à une tâche particulière. Le convertisseur proposé est basé sur la technique de l'échantillonnage non-uniforme en ondelettes (NUWS). L'architecture mélange le signal analogique avec des ondelettes paramétrables avant d'intégrer et

convertir le signal en données numériques. L'objectif de la thèse est de proposer une méthode pour concevoir un convertisseur A2F générique basé sur le NUWS. Il est ainsi nécessaire de définir les caractéristiques des ondelettes afin d'acquérir une large gamme de signaux basse fréquence (ECG, EMG, EEG, parole...). Cette étape nécessite l'utilisation d'algorithmes de sélection de paramètres et d'algorithmes d'apprentissage automatique pour sélectionner le meilleur ensemble d'ondelettes pour une application donnée et qui doit permettre de définir les spécifications du convertisseur. L'étape de sélection des paramètres doit tenir compte des contraintes de mise en œuvre pour optimiser au mieux la consommation d'énergie. Un algorithme de sélection de paramètres est proposé pour choisir des ondelettes pour une application donnée, afin de maximiser la précision de classification tout en diminuant la consommation d'énergie, grâce à un modèle de consommation réalisé dans une technologie CMOS 0.18  $\mu\text{m}$ .

**Title :** Design and integration of a flexible analog-to-feature converter for smart sensors

**Keywords :** Analog-to-feature converter, Feature selection, Machine Learning, Non-Uniform Wavelet Sampling, Smart sensors.

**Abstract :**

The Internet of Things (IoT) is currently experiencing huge developments. IoT includes lots of different devices such as Wireless Sensors Networks (WSN) or wearable electronics that rely on wireless communications. These networks need to understand the context in which they are used. This means that the system must know what is happening around it, i.e. sense the environment, and understands the needs of the user. This requires always-on sensing on many sensors while being small, cheap, reliable and having a lifetime of several years. Analog-to-Feature (A2F) conversion is a new acquisition method that was thought for IoT devices. The converter aims at extracting useful features directly on the analog signal. By carefully choosing a set of features, it is possible to acquire only the relevant information for a given task. The proposed converter is based on the Non-Uniform Wavelet Sampling

(NUWS) architecture. The architecture mixes the analog signal with tunable wavelets prior to integration and digital conversion. The aim of the thesis is to propose a method to design a generic A2F converter based on the NUWS. It includes the definition of the wavelet parameters in order to acquire a broad range of low frequency signals (ECG, EMG, EEG, speech ...). This step requires the use of feature selection algorithms and machine learning algorithms for selecting the best set of wavelets for a given application and should be used to define the specifications for the converter. The feature selection step must be aware of physical implementation constraints to optimize energy consumption as much as possible. A feature selection algorithm is proposed to choose wavelets for a given application, in order to maximize classification accuracy while decreasing power consumption, through a power model designed in 0.18  $\mu\text{m}$  CMOS technology.