



**HAL**  
open science

# Représentation de signaux robuste aux bruits -Application à la détection et l'identification des signaux d'alarme

Fatimetou El Jili

► **To cite this version:**

Fatimetou El Jili. Représentation de signaux robuste aux bruits -Application à la détection et l'identification des signaux d'alarme. Traitement du signal et de l'image [eess.SP]. Université de Reims Champagne-Ardenne, 2018. Français. NNT: . tel-02460271

**HAL Id: tel-02460271**

**<https://theses.hal.science/tel-02460271>**

Submitted on 29 Jan 2020

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

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

# THÈSE

Pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE**

*Discipline : AUTOMATIQUE, SIGNAL, PRODUCTIQUE, ROBOTIQUE*

*Spécialité : Traitement du Signal*

Présentée et soutenue publiquement par

**FATIMETOU EL JILI**

Le 17 décembre 2018

---

**Représentation de signaux robuste aux bruits - Application à la détection et l'identification des signaux d'alarme**

---

Thèse dirigée par **MAMADOU MBOUP**

## JURY

M. Frédéric MORAIN-NICOLIER,	Professeur,	Université de Reims Champagne Ardenne,	<b>Président</b>
Mme Régine LE BOUQUIN JEANNÈS,	Professeur,	Université de Rennes 1,	<b>Rapporteur</b>
Mme Meriem JAIDANE,	Professeur,	École Nationale d'Ingénieurs de Tunis ,	<b>Rapporteur</b>
M. Alban GOUPIL,	Maître de Conférences,	Université de Reims Champagne Ardenne,	<b>Examineur</b>
M. Mamadou MBOUP,	Professeur,	Université de Reims Champagne Ardenne,	<b>Directeur de thèse</b>
M. Gaël MAHÉ,	Maître de Conférences,	Université Paris Descartes,	<b>Co-encadrant</b>



## A MES PARENTS

qui m'ont toujours éclairé mon chemin et qui m'ont encouragé et soutenu durant toutes mes études.

## A MES SOEURS ET FRÈRE

qui n'ont jamais cessé de m'encourager durant ma thèse, et qui m'ont toujours apporté la joie et le bonheur.

# Remerciements

Je tiens à exprimer ma profonde gratitude à Monsieur Mamadou MBOUP et Monsieur Gaël MAHÉ mes encadrants de thèse qui ont été toujours attentifs et disponibles malgré leurs nombreuses charges et responsabilités. Leurs compétences et leur rigueur scientifique m'ont beaucoup appris, travailler sous leur direction était une expérience humaine et professionnelle très enrichissante.

Je remercie l'ensemble des membres de mon jury : Messieurs Frédéric MORAIN-NICOLIER et Alban GOUPIL et Mesdames Régine LE BOUQUIN JEANNÈS et Meriem JAIDANE.

Je souhaite également remercier Monsieur Bernard RIERA le directeur du laboratoire CReSTIC et Monsieur Petar M.DJURIC. Je tiens également à remercier mon père Mohamed el Bechir EL JILI qui m'a toujours soutenu.

# Table des matières

<b>Remerciements</b>	<b>3</b>
<b>Introduction</b>	<b>15</b>
<b>I Concepts généraux et état de l'art</b>	<b>17</b>
I.1 Transformées temps-fréquence . . . . .	17
I.1.1 La transformée de Fourier . . . . .	17
I.1.2 La MDCT . . . . .	18
I.2 La quantification vectorielle . . . . .	19
I.3 Le Matching Pursuit . . . . .	20
I.4 Le codage correcteur d'erreurs . . . . .	22
I.4.1 Les codes BCH . . . . .	23
I.4.2 Les Turbo codes . . . . .	26
I.5 Détection et identification . . . . .	29
I.5.1 Classifieurs Probabilistiques . . . . .	29
I.5.2 Détection de pitch . . . . .	34
I.5.3 Le réseau artificiel de neurones . . . . .	36
<b>II Détection et identification d'alarme des véhicules prioritaires par étude du sonogramme</b>	<b>41</b>
II.1 Recherche de la signature d'une alarme . . . . .	42
II.1.1 Recherche de la signature fréquentielle . . . . .	43
II.1.2 Recherche de la signature temporelle . . . . .	46
II.1.3 Identification de la bande de fréquence la moins bruitée . . . . .	48
II.1.4 Algorithme de détection et d'identification conjointes d'alarmes . . . . .	50
II.2 Application à la détection et l'identification d'un signal d'alarme d'une voiture de pompiers . . . . .	53
II.2.1 Signal peu bruité . . . . .	53
II.2.2 Signal bruité . . . . .	57
II.3 étude de performance (Courbe COR) . . . . .	61

II.4	Conclusion . . . . .	65
<b>III</b>	<b>Quantification correctrice d'erreurs</b>	<b>67</b>
III.1	Quantification fondée sur le codage correcteur d'erreurs . . . . .	67
III.1.1	Principe . . . . .	68
III.1.2	Application du principe de la modulation codée en bloc à la quantification . . . . .	70
III.1.3	Comparaison avec la quantification vectorielle . . . . .	71
III.2	Choix des codeurs . . . . .	71
III.2.1	Probabilité d'erreur binaire avant décodage . . . . .	71
III.2.2	Probabilité d'erreur binaire après décodage . . . . .	78
III.2.3	Choix des codeurs . . . . .	80
III.3	Application à la réduction du bruit . . . . .	83
III.3.1	Application aux signaux à distribution uniforme . . . . .	83
III.3.2	Application aux signaux d'alarmes . . . . .	84
III.3.3	Distorsion introduite par la quantification . . . . .	86
III.4	Conclusion . . . . .	87
<b>IV</b>	<b>Matching Pursuit pour la quantification</b>	<b>91</b>
IV.1	Algorithme du matching pursuit modifié pour la quantification . . . . .	93
IV.1.1	Minimisation du résidu à chaque itération . . . . .	95
IV.1.2	Minimisation de l'erreur finale de quantification : approche empirique . . . . .	97
IV.1.3	Minimisation de l'erreur finale de quantification : approche théorique	100
IV.2	Comparaison des codes BCH avec des turbo-codes . . . . .	106
IV.3	Choix des dictionnaires de mots de code . . . . .	108
IV.3.1	Problématique . . . . .	108
IV.3.2	Application aux signaux à distribution uniforme . . . . .	111
IV.4	Comparaison avec la méthode du chapitre III . . . . .	112
IV.5	Application aux signaux d'alarme . . . . .	116
IV.5.1	Quelques caractéristiques des signaux d'alarme et du bruit de trafic	117
IV.5.2	Probabilité d'erreurs introduites par le bruit de trafic . . . . .	119
IV.5.3	Application de l'algorithme du Matching Pursuit modifié pour la quantification des signaux d'alarme . . . . .	120
IV.6	Conclusion . . . . .	125
<b>V</b>	<b>Détection et identification des signaux</b>	<b>127</b>
V.1	Modèle de détection et identification des signaux dans le domaine temporel	128

V.2	Modèle de détection et identification des signaux dans le domaine fréquentiel	130
V.3	Processus de détection et d'identification . . . . .	133
V.4	Application à la détection et l'identification des signaux d'alarme . . . . .	138
V.4.1	Détection et identification des signaux d'alarme . . . . .	139
V.4.2	Analyse de la détection et de l'identification dans le domaine temporel : Signal quantifié avant transmission . . . . .	140
V.4.3	Analyse de la détection et de l'identification dans le domaine fréquentiel : signal quantifié avant transmission . . . . .	142
V.4.4	Analyse de la détection et de l'identification dans le domaine temporel : signal non quantifié avant transmission . . . . .	145
V.4.5	Analyse de la détection et de l'identification dans le domaine fréquentiel : signal non quantifié avant transmission . . . . .	147
V.4.6	Analyse de la détection et de l'identification des signaux : comparaison avec la méthode de corrélation . . . . .	150
V.5	Détection et identification des signaux d'alarme quantifiés par recherche de signature . . . . .	153
V.6	Conclusion . . . . .	155

**Conclusion** **157**

# Table des figures

I.1	Turbo Codes : codeur . . . . .	27
I.2	Turbo Codes : décodeur . . . . .	28
I.3	Le réseau artificiel de neurones : perceptron multicouche . . . . .	36
II.1	A gauche, représentation dans le plan temps-fréquence du sonagramme d'un signal d'alarme, à droite la projection sur le plan fréquence-puissance du sonagramme de l'alarme . . . . .	42
II.2	À gauche sonagramme d'un signal d'alarme. À droite zoom sur le sonagramme . . . . .	43
II.3	Signature fréquentielle : à gauche $\Phi(f, t)$ , à droite sa dérivée, $t=0$ s . . .	44
II.4	Signature temporelle : $p(t)$ et sa dérivée . . . . .	47
II.5	Sonagramme d'un signal d'alarme de sapeurs-pompiers, d'une durée de 78 s, de fréquences fondamentales $f_1=435$ Hz, $f_2=488$ Hz, de durées respectives $T_1 = 1.1$ s et $T_2 = 1.2$ s . . . . .	53
II.6	À gauche la projection $\Phi(f, t)$ . À droite la dérivée de $\Phi(f, t)$ , $t=0$ s. Les lignes verticales de couleur rouge indiquent les positions de $f_1=435$ Hz et de ses harmoniques et ceux de couleur verte indiquent les positions de $f_2=488$ Hz et de ses harmoniques . . . . .	54
II.7	En haut : à droite $p(t)$ , à gauche la dérivée de $p(t)$ . En bas : zoom sur $p(t)$ et sa dérivée, signal de sapeurs-pompiers . . . . .	55
II.8	Estimation de $T_1$ et $T_2$ dans le temps . . . . .	56
II.9	Détection de l'algorithme . . . . .	56
II.10	Sonagramme du signal enregistré, signal très bruité . . . . .	58
II.11	$\Phi(f, t)$ et sa dérivée pour $t = 0$ s et $\Delta$ égal à la taille du signal . . . . .	58
II.12	$p(t)$ d'un signal de sapeurs-pompier enregistré, signal très bruité . . . . .	59
II.13	Estimation de $T_1$ et $T_2$ dans le temps . . . . .	60
II.14	Détection de l'algorithme . . . . .	60
II.15	Probabilité de détection vs probabilité de fausse détection . . . . .	62



II.16	Probabilité de détection vs probabilité de fausse détection, $\epsilon'=[0.1 \text{ s}; 0.2 \text{ s}; 0.3 \text{ s}; 0.35 \text{ s}; 0.4 \text{ s}; 0.45 \text{ s}]$ , epsilon ici est une erreur relative $\epsilon=\frac{ \epsilon }{f_1}$ s'il s'agit de l'estimation de $f_1$ et $\epsilon=\frac{ \epsilon }{f_2}$ s'il s'agit de l'estimation de $f_2$ . . . . .	63
II.17	Probabilité de détection vs probabilité de fausse détection : détection d'un signal de sapeurs pompier en présence d'un autre signal d'alarme. . . . .	64
III.1	Principe de la modulation codée en bloc. . . . .	70
III.2	Probabilité d'erreur théorique $P_e^i$ en fonction de $i$ , pour plusieurs SNR, pour un signal à distribution uniforme bruité par un bruit blanc gaussien, représenté sur $L + 1=16$ bits. . . . .	78
III.3	Nombre d'erreurs après décodage correcteur d'erreurs vs nombre d'erreurs avant décodage correcteur d'erreur, pour des codes BCH de longueur $n = 31$ . . . . .	79
III.4	Probabilité d'erreur avant et après décodage correcteur d'erreurs pour plusieurs poids de bit vs capacités de correction des codeurs, pour un SNR=30 dB. Sous l'hypothèse que les signaux suivent une loi de distribution uniforme et sont bruités par un bruit blanc Gaussien, signaux représentés sur $L + 1=16$ bits. . . . .	81
III.5	Capacités de correction minimales $t_i$ en fonction du poids de bit pour plusieurs valeurs du SNR du canal de transmission et pour des mots de code de longueur $n = 31$ . Sous l'hypothèse que les signaux suivent une loi de distribution uniforme et sont bruités par un bruit blanc Gaussien, signaux représentés sur 16 bits. . . . .	82
III.6	Chaîne de transmission proposée. . . . .	83
III.7	Rapport signal à bruit (SNR) après décodage, en fonction du SNR avant décodage. Résultats sur $10^6$ simulations effectuées sur des signaux de taille $n = 31$ générés aléatoirement suivant une loi de distribution uniforme sur l'intervalle entier $[-2^L + 1 + 1, 2^L - 1]$ . . . . .	85
III.8	$\text{SNR}_{dB}$ après décodage, en fonction du $\text{SNR}_{dB}$ avant décodage. Résultats des simulations sur un signal d'alarme de sapeurs-pompiers. . . . .	86
III.9	À gauche histogramme du signal temporel de sapeurs-pompiers, à droite histogramme de la MDCT du signal. . . . .	87
III.10	$\text{SNR}_{dB}$ de l'erreur du codage, en fonction du $\text{SNR}_{dB}$ du canal de transmission. Résultats des simulations sur un signal d'alarme de sapeurs-pompiers. . . . .	88

III.11	Spectrogramme d'un signal d'alarme de sapeurs-pompiers : Signal original à gauche et signal quantifié à droite (quantification dans le domaine temporel). . . . .	89
III.12	Spectrogramme du signal original à gauche et signal quantifié à droite (quantification dans le domaine fréquentiel) . . . . .	89
IV.1	La norme du résidu à chaque itération de l'algorithme ; l'erreur de quantification est égale au résidu à la dernière itération. Simulations effectuées sur un nombre $N = 10^5$ de trames de longueur $n$ et de distribution uniforme dans $A^n$ . . . . .	95
IV.2	La norme $E^\lambda$ de l'erreur de quantification en fonction de $\lambda$ . . . . .	98
IV.3	La norme $E_l$ du résidu par itération en fonction de $\lambda$ , $0 \leq l \leq L - 1$ . . . . .	99
IV.4	Diagramme d'état schématisant la formation des erreurs au cours de l'algorithme du matching pursuit. . . . .	103
IV.5	Résidu du codage et du décodage par itération : $f_l$ vs $J_{1,l}$ , pour un bruit gaussien et un SNR=20 dB. . . . .	105
IV.6	Probabilités d'erreur après décodage : décodage utilisant $f_l$ vs décodage utilisant $J_{1,l}$ , signaux à distribution uniforme, bruit gaussien. . . . .	106
IV.7	Résidu du codage du Matching pursuit : dictionnaire de Turbo Code $\zeta(n = 72, k = 20, R = k/(3k + 12))$ et dictionnaire de codes BCH $\zeta(n = 31, k = 11)$ . . . . .	107
IV.8	Résidu du décodage du Matching pursuit : dictionnaire de Turbo Code $\zeta(n = 72, k = 20, R = k/(3k + 12))$ et dictionnaire de codes BCH $\zeta(n = 31, k = 11)$ . . . . .	108
IV.9	Probabilité d'erreur du décodage par le Matching pursuit, dictionnaire de Turbo codes $\zeta(n = 72, k = 20, R = k/(3k + 12))$ et dictionnaire de codes BCH $\zeta(n = 31, k = 11)$ . . . . .	109
IV.10	Moyenne du nombre d'erreurs après décodage vs nombre d'erreurs avant décodage : à gauche décodage par l'algorithme du matching pursuit modifié pour la quantification utilisant des dictionnaires de codes BCH, à droite décodage correcteur d'erreur par les codes BCH de capacité de correction $t = 2, 3, 5, 7$ . . . . .	110
IV.11	Moyenne du nombre d'erreurs après décodage par poids de bit : à gauche décodage du matching pursuit vs décodage correcteur d'erreur, à droite décodage du matching pursuit utilisant plusieurs codes BCH de dimension $k = 6, 11, 16$ . . . . .	110

IV.12	Probabilité d'erreur après décodage par poids de bit : simulations effectuées sur un signal de distribution uniforme et pour un SNR=10 dB, $i$ indice de poids de bit. . . . .	112
IV.13	Capacités de correction nécessaire pour corriger les erreurs introduites par un bruit blanc gaussien sur une trame de longueur $n=31$ d'un signal de distribution uniforme. . . . .	113
IV.14	Résidu du codage du Matching poursuit et du codage classique : dictionnaire de codes BCH et Codeurs BCH, simulations effectuées pour un rapport signal à bruit de 20 dB . . . . .	114
IV.15	Résidu du codage et décodage de la quantification fondée sur le Matching poursuit et du codage correcteur d'erreur : dictionnaire de codes BCH et Codeurs BCH, expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit gaussien, pour un rapport signal bruit de 30 dB . . . . .	115
IV.16	Erreur relative du codage (à gauche) et SNR du codage (à droite) par le Matching poursuit et par le codage correcteur d'erreurs. Expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit blanc gaussien. . . . .	116
IV.17	Erreur relative du décodage (à gauche) et SNR du décodage (à droite) par le Matching poursuit et par le décodage correcteur d'erreurs. Expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit blanc gaussien. . . . .	117
IV.18	Spectre de la MDCT d'un signal d'alarme de sapeurs-pompiers . . . . .	118
IV.19	Spectre de la MDCT du bruit de trafic . . . . .	118
IV.20	Probabilité d'erreur par poids de bit : à gauche probabilité d'erreurs temporelles, à droite probabilité d'erreurs fréquentielles, simulations effectuées sur un signal de sapeurs-pompiers. . . . .	120
IV.21	Probabilité d'erreur après décodage par poids de bit : probabilité d'erreurs temporelles, simulations effectuées sur un signal de sapeurs-pompiers et pour un SNR=10 dB, $i$ indice de poids de bit. . . . .	121
IV.22	Probabilité d'erreur après décodage par poids de bit : probabilité d'erreurs fréquentielles, simulations effectuées sur un signal de sapeurs-pompiers et pour un SNR=10 dB, $i$ indice de poids de bit. . . . .	122
IV.23	Capacités de correction nécessaire pour corriger les erreurs du bruit introduites sur une trame de longueur $n=31$ du signal de sapeurs-pompiers. . . . .	123

IV.24	Résidu du décodage du Matching pursuit et du codage classique : dictionnaire de codes BCH, expérience effectuée sur un signal d'ambulance, pour un rapport signal à bruit de 30 dB. . . . .	123
IV.25	SNR et erreur relative de quantification utilisant le codage de l'algorithme du Matching pursuit modifié et du codage classique : dictionnaire de codes BCH, expérience effectuée sur un signal d'ambulance. . . . .	124
IV.26	SNR et erreur relative du décodage par le Matching pursuit et du décodage par le décodage correcteur d'erreur : dictionnaire de codes BCH. expérience effectuée sur un signal d'ambulance. . . . .	124
IV.27	SNR global et erreur relative globale du décodage par le Matching pursuit et du décodage par le décodage correcteur d'erreur : dictionnaire de codes BCH. expérience effectuée sur un signal d'ambulance. . . . .	125
V.1	Chaîne de transmission : cas des signaux quantifiés avant transmission (domaine temporel). . . . .	129
V.2	Chaîne de transmission : cas des signaux transmis sans quantification (domaine temporel). . . . .	129
V.3	Chaîne de transmission : cas des signaux quantifiés avant transmission (domaine fréquentiel). . . . .	132
V.4	Chaîne de transmission : cas des signaux transmis sans quantification (domaine fréquentiel). . . . .	132
V.5	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=-1 dB, signal quantifié avant transmission, domaine temporel. . . .	140
V.6	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=5 dB, signal quantifié avant transmission, domaine temporel. . . .	141
V.7	Courbes COR paramétrées par $\epsilon$ , $\eta = 2$ , pour plusieurs SNR, signal quantifié avant transmission, domaine temporel. . . . .	141
V.8	Courbes COR paramétrées par $\epsilon$ , $\eta = 4$ , pour plusieurs SNR, signal quantifié avant transmission, domaine temporel. . . . .	142
V.9	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=-1 dB, signal quantifié avant transmission, domaine fréquentiel. . . .	143
V.10	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=5 dB, signal quantifié avant transmission, domaine fréquentiel. . . .	143
V.11	Courbes COR paramétrées par $\epsilon$ , $\eta = 2$ , pour plusieurs SNR, signal quantifié avant transmission, domaine fréquentiel. . . . .	144
V.12	Courbes COR paramétrées par $\epsilon$ , $\eta = 4$ , pour plusieurs SNR, signal quantifié avant transmission, domaine fréquentiel. . . . .	144

V.13	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=5 dB, signal non quantifié avant transmission, domaine temporel. . . . .	145
V.14	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=10 dB, signal non quantifié avant transmission, domaine temporel. . . . .	145
V.15	Courbes COR paramétrées par $\epsilon$ , $\eta = 2$ , pour plusieurs SNR, signal non quantifié avant transmission, domaine temporel. . . . .	146
V.16	Courbes COR paramétrées par $\epsilon$ , $\eta = 4$ , pour plusieurs SNR, signal non quantifié avant transmission, domaine temporel. . . . .	146
V.17	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=5 dB, signal non quantifié avant transmission, domaine fréquentiel. . . . .	148
V.18	Courbes COR paramétrées par $\epsilon$ , pour plusieurs valeurs de $\eta$ , pour un SNR=10 dB, signal non quantifié avant transmission, domaine fréquentiel. . . . .	148
V.19	Courbes COR paramétrées par $\epsilon$ , $\eta = 2$ , pour plusieurs valeurs du SNR, signal non quantifié avant transmission, domaine fréquentiel. . . . .	149
V.20	Courbes COR paramétrées par $\epsilon$ , $\eta = 4$ , pour plusieurs valeurs du SNR, signal non quantifié avant transmission, domaine fréquentiel. . . . .	149
V.21	Courbes COR paramétrées par $\epsilon$ pour les algorithmes proposés et par $s_d$ pour la corrélation, pour un SNR=-1 dB, $\eta = 2$ , signal quantifié avant transmission, domaine fréquentiel. . . . .	151
V.22	Courbes COR paramétrées par $\epsilon$ pour les algorithmes proposés et par $s_d$ pour la corrélation, pour un SNR=-1 dB, $\eta = 4$ , signal quantifié avant transmission, domaine fréquentiel. . . . .	151
V.23	Courbes COR paramétrées par $\epsilon$ pour les algorithmes proposés et par $s_d$ pour la corrélation, pour un SNR=5 dB, $\eta = 2$ , signal quantifié avant transmission, domaine temporel. . . . .	152
V.24	Courbes COR paramétrées par $\epsilon$ pour les algorithmes proposés et par $s_d$ pour la corrélation, pour un SNR=5 dB, $\eta = 4$ , signal quantifié avant transmission, domaine temporel. . . . .	152
V.25	Courbes COR paramétrées par $\epsilon$ , signal d'alarme de sapeurs-pompiers non quantifié. . . . .	154
V.26	Courbes COR paramétrées par $\epsilon$ , signal d'alarme de sapeurs-pompiers quantifié avant et après transmission (chaîne V.1). . . . .	155
V.27	Courbes COR paramétrées par $\epsilon$ , signal d'alarme de sapeurs-pompiers quantifié après transmission (chaîne V.2). . . . .	156
V.28	Courbes COR paramétrées par $\epsilon$ , signal d'alarme de sapeurs-pompiers, $\mu=3$ . . . . .	156

---

## **Représentation de signaux robuste aux bruits - Application à la détection et l'identification des signaux d'alarme**

---

Ces travaux ont pour application la détection l'identification des signaux audio et particulièrement les signaux d'alarmes de voitures prioritaires. Dans un premier temps, nous proposons une méthode de détection des signaux d'alarme dans un environnement bruité, fondée sur des techniques d'analyse temps-fréquence des signaux. Cette méthode permet de détecter et d'identifier des signaux d'alarmes noyés dans du bruit, y compris pour des rapports signal à bruit négatifs. Puis nous proposons une quantification des signaux robuste aux bruits de transmission. Il s'agit de remplacer chaque niveau de bit d'un vecteur d'échantillons temporels ou fréquentiels par un mot binaire de même longueur fourni par un codeur correcteur d'erreur. Dans une première approche, chaque niveau de bits est quantifié indépendamment des autres selon le critère de minimisation de la distance de Hamming. Dans une seconde approche, pour réduire l'erreur de quantification à robustesse égale, les différents niveaux de bits sont quantifiés successivement selon un algorithme de type matching pursuit. Cette quantification donne aux signaux une forme spécifique permettant par la suite de les reconnaître facilement parmi d'autres signaux. Nous proposons donc enfin deux méthodes de détection et d'identification des signaux fondées sur la quantification robuste, opérant dans le domaine temporel ou dans le domaine fréquentiel, par minimisation de la distance entre les signaux reçus restreints à leurs bits de poids fort et les signaux de référence. Ces méthodes permettent de détecter et d'identifier les signaux dans des environnements à rapport signal à bruit très faible et ceci grâce à la quantification. Par ailleurs, la première méthode, fondée sur la signature temps-fréquence, s'avère plus performante avec les signaux quantifiés.

---

Mots-clés : signaux audio, alarmes, détection, identification, quantification robuste, analyse temps-fréquence, quantification vectorielle, codage de canal, matching pursuit.

---

## **Signals representation robust to noise - Application to the detection and identification of alarm signals**

---

This work targets the detection and identification of audio signals and in particular alarm signals from priority cars. First, we propose a method for detecting alarm signals in a noisy environment, based on time-frequency signal analysis. This method makes it possible to detect and identify alarm signals embedded in noise, even with negative signal-to-noise ratios. Then we propose a signal quantization robust against transmission noise. This involves replacing each bit level of a vector of time or frequency samples with a binary word of the same length provided by an error-correcting encoder. In a first approach, each bit level is quantized independently of the others according to the Hamming distance minimization criterion. In a second approach, to reduce the quantization error at equal robustness, the different bit levels are quantized successively by a matching pursuit algorithm. This quantization gives the signals a specific shape that allows them to be easily recognized among other signals. Finally, we propose two methods for detecting and identifying signals based on robust quantization, operating in the time domain or in the frequency domain, by minimizing the distance between the received signals restricted to their high-weight bits and the reference signals. These methods make it possible to detect and identify signals in environments with very low signal-to-noise ratios, thanks to quantization. In addition, the first method, based on the time-frequency signature, is more efficient with quantized signals.

---

Key-word : audio signals, alarm, detection, identification, robust quantization, time-frequency analysis, vectorial quantization, channel coding, matching pursuit.

---

**Discipline : AUTOMATIQUE, SIGNAL, PRODUCTIQUE, ROBOTIQUE**

---

**Spécialité : Traitement du Signal**

Université de Reims Champagne-Ardenne

CRESTIC - EA 3804

Moulin de la Housse, 51897 Reims





# Introduction

Ce travail s'inscrit dans le projet ICityForAll. L'objectif du projet est de développer des méthodes de traitement du signal pour des systèmes embarqués destinés à rendre la cité plus accessible aux personnes souffrant de la presby-acousie, notamment les personnes âgées.

Dans les zones urbaines, les personnes atteintes de déficience auditive ont, en effet, beaucoup de mal à reconnaître rapidement et surtout à localiser dans l'espace les signaux d'alarme tels que sirènes de police, d'ambulance, de voiture de pompiers. Ces difficultés affectent leur indépendance et représentent une menace pour leur sécurité et pour la sécurité routière, plus généralement. Il s'agit d'étudier et de proposer des méthodes robustes et conjointes de détection et d'identification de signaux audio environnementaux, plus précisément les signaux d'alarmes des voitures prioritaires.

L'objectif est de proposer des techniques qui détectent et identifient ces signaux parmi d'autres signaux environnementaux, bruit de trafic, klaxons, musique, parole etc.... Ces signaux sont périodiques contrairement aux signaux environnementaux et disposent d'une forme spécifique qui les différencie des autres signaux. Ainsi dans un premier temps on s'est inspiré de la forme de ces signaux afin de proposer une méthode de détection de ces signaux.

L'environnement de transmission de ces signaux est en général trop bruité (rues). Contrairement aux signaux transmis via les canaux de transmissions habituels, ces si-



gnaux ne sont pas protégés contre le bruit de transmission. Le niveau élevé du bruit déforme le signal et rend la forme de ces signaux indiscernable du bruit.

Nous nous sommes inspirés de la quantification [27], [3], [33], [41] et du codage correcteur d'erreurs [50] afin de donner à ces signaux une nouvelle forme spécifique robuste aux bruits de transmission et facile à identifier. Après une présentation des techniques utilisées et de l'état de l'art dans le chapitre I, nous proposons dans le chapitre II des méthodes de détection d'alarme de véhicules de secours (voitures prioritaires comme le SAMU, voiture de police, ambulance etc ...) par l'analyse du spectrogramme (analyse temps-fréquence).

La détection et l'identification des signaux est confrontée à des niveaux du bruit ambiant très élevés de l'environnement dans lequel les signaux sont transmis. Ainsi un grand besoin de rendre ces signaux robustes au bruit s'impose.

Pour rendre les algorithmes de détection efficaces, nous proposons une méthode de quantification robuste des signaux basée sur le codage correcteur d'erreur dans le chapitre III. Une amélioration de cette quantification est proposée dans le chapitre IV. Ainsi les signaux quantifiés seront robustes au bruit ambiant, de telle sorte qu'on soit capable d'identifier et de détecter ces signaux dans des environnements de rapports signal à bruit très faibles. Des algorithmes de détection et d'identification des signaux fondés sur cette quantification sont proposés dans le chapitre V.

# Chapitre I

## Concepts généraux et état de l'art

Ce chapitre rassemble les travaux antérieurs qui sont à la base des contributions proposées au cours de cette thèse.

### I.1 Transformées temps-fréquence

#### I.1.1 La transformée de Fourier

**Définition I.1.1.** : Soit  $x(t)$  un signal temporel. Sa **transformée de Fourier (TF)** est une fonction complexe de la variable réelle  $f$  définie par :

$$X(f) = TF(x(t)) = \langle x(t), e^{-j2\pi ft} \rangle = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt.$$

**Définition I.1.2.** : On appelle **transformée de Fourier inverse** la fonction inverse de la transformée de Fourier  $X(f)$  donnée par :

$$x(t) = TF^{-1}(X(f)) = \langle X(f), e^{j2\pi ft} \rangle = \int_{-\infty}^{+\infty} X(f)e^{j2\pi ft} df.$$

**Définition I.1.3.** : Soit  $x(n)$  un signal discret, pour  $n \in [0, N - 1]$ , la **transformée de Fourier discrète (TFD)** de ce signal est donnée par :

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi \frac{nk}{N}},$$

avec  $k$  l'indice fréquentiel,  $k \in [0, N - 1]$ .

**Définition I.1.4.** : La transformée de Fourier discrète (TFD) inverse est donnée par :

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{j2\pi \frac{nk}{N}}.$$

### I.1.2 La MDCT

**Définition I.1.5.** : (Modified Discrete Cosine Transform) Soit  $x(n)$  une portion d'un signal discret,  $n = 0, 1, \dots, N - 1$ , la MDCT est une transformée temps-fréquence [42], [43], initialement conçue pour la suppression du phénomène de repliement causé par l'interpolation. La MDCT du signal  $x$  est donnée par :

$$X(k) = \frac{2}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) w(n) \cos \left( \frac{2\pi}{N} \left( n + \frac{N}{4} + \frac{1}{2} \right) \left( k + \frac{1}{2} \right) \right).$$

$0 \leq n \leq N - 1$ , l'indice temporel et  $0 \leq k \leq \frac{N}{2} - 1$ , l'indice fréquentiel.  $w(n)$  est une fenêtre symétrique satisfaisant la condition de la reconstitution parfaite :  $w^2(n) + w^2(n + \frac{N}{2}) = 1$ . La fenêtre  $w(n)$  utilisée dans ce manuscrit est la fenêtre de Kaiser Bessel [26] donnée par :

$$w(n) = \frac{I_0 \left( \beta \sqrt{1 - \left( \frac{n - N/2}{N/2} \right)^2} \right)}{I_0(\beta)}.$$

Le paramètre  $\beta$  permet de réaliser un compromis entre la largeur du lobe principal et l'amplitude des lobes secondaires.  $I_0$  est la fonction de Bessel de première espèce donnée par :

$$I_0(x) = \sum_{m=0}^{\infty} \frac{(-1)^m (x/2)^{2m}}{m! \Gamma(m + 1)},$$

avec  $\Gamma(m) = (m - 1)!$ .

**Définition I.1.6.** : La transformée inverse de la MDCT est donnée par :

$$x(n) = \frac{2}{\sqrt{N}} \sum_{k=0}^{\frac{N}{2}-1} X(k) \cos \left( \frac{2\pi}{N} \left( n + \frac{N}{4} + \frac{1}{2} \right) \left( k + \frac{1}{2} \right) \right).$$

## I.2 La quantification vectorielle

L'idée de la quantification vectorielle est le résultat des travaux de Claude Shannon sur le codage de source et la compression [51], [50] qui avait montré que les performances de la quantification peuvent s'améliorer en appliquant le système de quantification sur des vecteurs au lieu des scalaires. En 1956, Stuart P.Lloyd a proposé une optimisation de la quantification scalaire [41] qui a été à la base de l'amélioration de la quantification vectorielle et du développement de certains algorithmes de quantification vectorielle comme celui de LBG (Linde, Buzo et Gray) [27] proposé en 1980 par Joseph Linde, André Buzo et Robert Gray.

La quantification vectorielle est une extension de la quantification scalaire aux espaces multidimensionnels qui est un opérateur sur les vecteurs au lieu des scalaires. La quantification scalaire initialement prend en entrée un signal analogique discretisé, alors que la quantification vectorielle prend en général comme entrée des signaux qui sont déjà numériques, la sortie étant une version compressée du signal original.

La quantification vectorielle peut être considérée comme une forme de reconnaissance de motif, où un motif d'entrée sera approché par un motif appartenant à un ensemble prédéfini.

La quantification vectorielle de dimension  $k$  et de longueur  $N$  est une projection d'un vecteur appartenant à un espace euclidien de dimension  $k$  dans un ensemble  $\mathbb{D}$  de cardinal  $N$  appelé dictionnaire de mots de code :

$$Q : \mathbb{R}^k \longrightarrow \mathbb{D} \subset \mathbb{R}^k$$

$\mathbb{D}$  est donné par  $\mathbb{D} = \{ y_1 ; y_2 ; \dots ; y_i ; \dots ; y_N \}$ ,  $N = \text{card } \mathbb{D}$ . La résolution est donnée par  $r = (\log_2 N)/k$ . La quantification est associée à une partition de  $\mathbb{R}^k$  en  $N$  cellules  $\mathbb{R}_i$ ,  $i \in \mathcal{I} = \{1 ; 2 ; 3 ; \dots ; N\}$ , chaque  $\mathbb{R}_i$  est associé à un élément  $y_i$  de  $\mathbb{D}$  :

$$\mathbb{R}_i = \{x \in \mathbb{R}^k : Q(x) = y_i\} = Q^{-1}(y_i), \quad (\text{I.1})$$

Cette partition et les représentants  $y_i$  des classes  $\mathbb{R}_i$  sont construits de telle sorte que l'erreur quadratique moyenne de quantification,  $\|x - Q(x)\|^2$ , soit minimale.

Dans cette thèse nous nous sommes inspirés de l'idée de quantification afin de proposer une quantification correctrice d'erreur utilisant des dictionnaires de mots de codes (générés par des codeurs correcteur d'erreur), qui servira dans la suite à la détection et l'identification des signaux.

### I.3 Le Matching Pursuit

Le Matching Pursuit [17], [12] est un algorithme itératif qui permet de réaliser une décomposition parcimonieuse d'un signal à l'aide d'un dictionnaire  $D$ . Le signal est écrit comme une combinaison linéaire d'atomes  $g_{\gamma_n}(t)$  appartenant à un sous-ensemble de  $D$ .

$$x(t) = \sum_{n=0}^{+\infty} a_n(g_{\gamma_n}(t)).$$

Les coefficients  $a_n$  dépendent des atomes  $g_{\gamma_n}(t)$  choisis. Ces coefficients portent des informations sur la structure du signal  $x(t)$ . Ce sont ces coefficients que l'algorithme doit déterminer en plus des indices  $\gamma_n$  des atomes  $g_{\gamma_n}(t)$ ,  $\gamma_n \in \Gamma$ , où  $\Gamma$  désigne l'ensemble des indices des atomes constituant le dictionnaire  $D$ .

A chaque itération, l'algorithme doit choisir un atome  $(g_\gamma)_{\gamma \in \Gamma}$  qui correspond le mieux (au sens d'une corrélation maximale), à un facteur près, au résidu du signal (signal moins la combinaison des atomes choisis aux itérations précédentes), où le résidu à la première itération correspond au signal lui-même. Ce facteur définit le coefficient associé à l'atome choisi.

Le Matching Pursuit initialement proposé par Mallat et Zhang [17] utilise le produit scalaire comme corrélation.  $R^n$ , le résidu du signal  $x$  à la  $n$ -ième itération, est donné par :

$$R^n = x - \sum_{i=0}^{n-1} a_i(g_{\gamma_n}(t)),$$

avec  $R^0 = x$ . Le coefficient  $a_n$  est donné par :

$$a_n = \langle R^n, g_{\gamma_n} \rangle$$

tel que

$$|\langle R^n, g_{\gamma_n} \rangle| = \text{Sup} |\langle R^n, g_{\gamma} \rangle|_{\gamma \in \Gamma}.$$

On obtient alors le résidu d'ordre  $n + 1$  :

$$R^{n+1} = R^n - a_n g_{\gamma_n}.$$

Après  $m$  itérations le signal  $x$  s'écrit sous la forme suivante :

$$x = \sum_{n=0}^m \langle R^n, g_{\gamma_n} \rangle g_{\gamma_n} + R^{m+1} = \sum_{n=0}^m a_n g_{\gamma_n} + R^{m+1}.$$

Dans la plupart des cas l'algorithme s'arrête à l'itération  $M$ , telle que la norme du résidu  $\|R^{M+1}\|_2^2$  soit inférieure à une certaine erreur  $\epsilon$  désirée. Le nombre d'itérations nécessaire pour atteindre cette erreur dépend du taux de décroissance de la norme du résidu. Ainsi le signal  $x$  est approché par :

$$x \approx \sum_{n=0}^M a_n g_{\gamma_n}.$$

## I.4 Le codage correcteur d'erreurs

Le codage correcteur d'erreurs est une branche des mathématiques liée la théorie de l'information découverte par Claude Shannon en 1948 dans son papier [50]. Shannon a montré que chaque canal de communication possède une capacité  $c$ , mesuré en bits par seconde, et que tant que le taux de transmission  $R$  est inférieur à  $c$ , il est possible de concevoir un système de communication sans erreurs utilisant le codage correcteur d'erreurs.

La contribution de Shannon était de prouver l'existence de ces codes, mais il n'a pas dit comment trouver ces codes.

Après la publication de [50], les chercheurs se sont lancés dans la recherche de ces codes produisant une faible probabilité d'erreurs que Shannon a prédit. Le progrès dans la recherche de ces codes n'a pas abouti à des résultats importants dans les années 50 où uniquement des codes de faible capacité de correction ont été trouvés.

Dans les années 60 le domaine a été divisé en deux parties entre les algébristes qui se sont focalisés sur la classe des codes appelée codes en blocs et les probabilistes qui se sont concentrés sur la compréhension du codage et du décodage comme processus aléatoire. Parmi les codes en blocs, on peut citer les codes cycliques [38], les codes BCH [6], les codes Reed-solomon [55].

Les probabilistes ont découvert la seconde classe des codes appelée les codes convolutifs, initialement découverte par P.Elias, mais le premier à formuler ces codes est Haggelbarger [18], ensuite Peterson a présenté une version raffinée de ces codes dans [37]. Fourney a donné une structure algébrique de ces codes dans son papier [15]. Ainsi les probabilistes ont développé des algorithmes de décodage probabiliste efficaces.

Dans les années 70 les chercheurs des deux domaines probabilistes et algébristes se

sont réunis et ont fourni plusieurs algorithmes efficaces de décodage. En 1993 Claude Berrou, Alain Glavieux et Punya Thitimajshima ont inventé les turbo codes à partir des codes convolutifs [2], ces codeurs approchent la limite de Shannon. Cette découverte a ouvert de nombreuses pistes de recherche dans le domaine du codage correcteur d'erreur. Et par conséquent la découverte des codes LDPC (Low density Parity-check), initialement découverts par Gallager en 1962 [16], mais qui n'ont pas été utilisés jusqu'à 1996 ; motivés par l'apparition des turbo codes, Mackay et al [31] redécouvrent les codes de Gallager.

### I.4.1 Les codes BCH

Les codes BCH[6], [23] sont des codes de la classe des codes cycliques, l'appellation BCH provient des noms initiales de leurs inventeurs Bose, Chaudhuri et Hocquenghem.

Soit  $n = q^m - 1$  un entier premier avec  $q$  et  $\beta$  une racine primitive  $n^{ieme}$  de l'unité,  $\beta \in \mathbb{F}_{q^m}$ .  $\beta$  est d'ordre  $n$  ( $\beta^n = 1$ ), les éléments  $\beta, \beta^2, \dots, \beta^{n-1}$  sont distincts et forment l'ensemble des racines de  $X^n - 1$  dans  $\mathbb{F}_{q^m}$ ,  $X^n - 1 = \prod_{i=0}^{n-1} (X - \beta^i)$ .

$X^n - 1$  peut être factorisé en produit de polynômes irréductibles dans  $\mathbb{F}_q[X]$ . Soit  $\Sigma_j$  une partie de  $\{0, 1, \dots, n-1\}$  stable par multiplication par 2 modulo  $n$  définie par  $\Sigma_j = \{j, qj, q^2j, \dots, q^{r-1}j\}$ , tel que  $j = 2^r j \pmod n$ .  $\Sigma_j$  est appelée classe cyclotomique engendrée par  $j$ .

A chaque classe cyclotomique est associé un polynôme irréductible dans  $\mathbb{F}_q[X]$ . Ce polynôme est le polynôme minimal de  $\beta^j$  donné par :

$$P_{\beta^j}(X) = \prod_{i \in \Sigma_j} (X - \beta^i).$$

Ainsi,  $X^n - 1$  peut être factorisé en produit de polynômes irréductibles dans  $\mathbb{F}_q[X]$  :

$$X^n - 1 = \prod_{i=0}^{n-1} (X - \beta^i) = \prod_{j \in J} P_{\beta^j}(X),$$



où  $J$  désigne l'ensemble des indices des classes cyclotomiques.

Soit  $P_{\beta^k}(X)$  le polynôme minimal de  $\beta^k$ ,  $1 \leq k \leq n-1$ . En application du théorème I.4.1 ci-après, Bose, Chaudhuri et Hocquenghem ont proposé d'utiliser comme polynôme générateur d'un code, le polynôme  $g(X)$  donné par :

$$g(X) = \text{ppcm}(P_{\beta^k}(X), P_{\beta^{k+1}}(X), \dots, P_{\beta^{k+s+1}}(X)).$$

**Théorème I.4.1.** *Soit  $C$  un code cyclique de longueur  $n$  et de polynôme générateur  $g \in \mathbb{F}_q[X]$  avec  $n$  et  $q$  premiers entre eux ( $\text{pgcd}(n, q) = 1$ ) et soit  $\beta$  une racine  $n^{\text{ième}}$  primitive de l'unité. S'il existe deux entiers  $l$  et  $s$  tels que  $g(\beta^l) = g(\beta^{l+1}) = \dots = g(\beta^{l+s}) = 0$ , alors la distance minimale  $\delta(C)$  de  $C$  vérifie  $\delta(C) \geq s+1$ . La capacité de correction de ce codeur est  $t = \lfloor \frac{\delta(C)-1}{2} \rfloor$ , le code  $C$  est donc au moins  $\lfloor s/2 \rfloor$  correcteur ( $\lfloor \cdot \rfloor$  désigne la partie entière).*

**Définition I.4.1.** (Code BCH) Un code BCH de distance construite  $d' = s+1$ , est un code cyclique de polynôme générateur le produit des polynômes minimaux des racines  $\beta^l, \beta^{l+1}, \dots, \beta^{l+s}$ ; le polynôme  $g(X)$  de degré  $n-k+1 = s+1$ ,  $k$  la dimension du code et  $n$  sa longueur. La distance minimale  $\delta$  du code est supérieure ou égale à  $d'$ ,  $\delta \geq d'$ .

#### I.4.1.1 Décodage des codes BCH

Les codes BCH disposent de plusieurs algorithmes de décodage dont on peut citer : l'algorithme de Berlekamp-Massey [23], [32], [5], l'algorithme du décodage euclidien de Sugiyama et l'algorithme du décodage de Peterson, Gorenstein et Zierler [39]. Tous ces algorithmes sont fondés sur l'idée du polynôme localisateur d'erreurs dont les racines permettent de localiser les erreurs. Dans cette partie nous présentons l'algorithme de décodage de Peterson, Gorenstein et Zierler.

Supposons que  $y$  soit le vecteur binaire reçu et  $Q(X)$  le polynôme associé à ce vecteur, alors  $Q(X)$  peut s'écrire :

$$Q(X) = q(X)g(X) + e(X), \tag{I.2}$$

avec  $g(X)$  le polynôme générateur du code BCH,  $q(X)$  un polynôme de degré  $k - 1$ , ( $k$  la dimension du code) et  $e(X)$  le reste de la division euclidienne de  $Q(X)$  par  $g(X)$  de degré inférieur au degré de  $g(X)$ . Notons  $\alpha^i$ ,  $i = 1 \dots s$  ( $s = 2t$ ,  $t$  la capacité de correction) les racines de  $g(X)$ ,  $\alpha = \beta^l$ , alors les syndromes sont donnés par :

$$\begin{aligned} S_1 &= Q(\alpha) \\ S_2 &= Q(\alpha^2) \\ &\cdot \\ &\cdot \\ S_i &= Q(\alpha^i) = e(\alpha^i) \\ &\cdot \\ &\cdot \\ S_{2t} &= Q(\alpha^{2t}) = e(\alpha^{2t}) \end{aligned}$$

Posons  $S$  la matrice formée par les syndromes donnée par :

$$S = \begin{bmatrix} S_1 & \dots & S_t \\ S_2 & \dots & S_{t+1} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ S_t & \dots & S_{2t-1} \end{bmatrix}$$

Alors le polynôme localisateur d'erreur est donné par :

$$L(X) = \varsigma_\nu X^\nu + \dots + \varsigma_1 X + 1, \quad (\text{I.3})$$

avec  $\nu$  le rang de la matrice  $S$  et  $\varsigma_\nu, \varsigma_{\nu-1}, \dots, \varsigma_1$  les solutions du système d'équation :

$$S = \begin{bmatrix} S_1 & \dots & S_\nu \\ S_2 & \dots & S_{\nu+1} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ S_t & \dots & S_{2\nu-1} \end{bmatrix} \times \begin{bmatrix} \varsigma_\nu \\ \varsigma_{\nu-1} \\ \cdot \\ \cdot \\ \varsigma_1 \end{bmatrix} = \begin{bmatrix} S_{t+1} \\ S_{t+2} \\ \cdot \\ \cdot \\ S_{2t} \end{bmatrix}$$

Soit  $\alpha^{-i_1}, \dots, \alpha^{-i_\nu}$  les racines du polynôme localisateur d'erreur  $L(X)$  donné par l'équation (I.3), alors les erreurs sont localisées aux positions  $-i_1, \dots, -i_\nu$ . Pour corriger ces erreurs chaque bit est remplacé par son opposé (les bits 0 sont remplacés par 1 et vice-versa).

## I.4.2 Les Turbo codes

En 1993 Claude Berrou, Alain Glavieux et Punya Thitimajshima ont inventé les turbo codes [2]. Les turbo codes correspondent à une concaténation en parallèle de deux codeurs convolutifs [18], [37], [15], en général pris identiques. Toutefois le décodage des turbo codes utilise une concaténation des décodeurs en série. Le décodage en série est utilisé parce qu'il est plus performant que le décodage utilisant des décodeurs concaténés en parallèle. En effet le décodage en série a la particularité de partager l'information entre les décodages ce qui améliore les performances du système contrairement au décodage en parallèle où chaque décodeur fait son décodage indépendamment de l'autre. Le codeur est schématisé par la figure I.1.

Les turbo codes sont constitués de deux codes convolutifs  $C_1$  et  $C_2$  de type spécifique ; codes convolutifs récurrents et systématiques, RSC[54]. En plus de ces codeurs, une permutation  $\pi$  est mise à l'entrée du codeur  $C_2$ . Un multiplexeur et un tronçonneur sont placés à la sortie des codeurs permettant ainsi de combiner (et en tronçant s'il s'agit d'un code tronqué) les sorties des deux codeurs pour obtenir le rendement souhaité.

Plusieurs paramètres jouent un rôle important sur la performance des turbo codes comme : le nombre d'itérations du décodage, les polynômes générateurs des codes convolutifs et le nombre d'états du registre.

La récursivité des codes convolutifs augmente le poids de Hamming des mots de codes par rapport aux codes non récurrents. Un codeur de poids de Hamming faible produit un faible pouvoir de correction. Si l'un des codeurs a une sortie de faible poids de Hamming, la permutation de la séquence binaire à l'entrée du codeur  $C_2$  produit à la sortie de ce codeur un mot de code de poids de Hamming élevé pour garantir une amélioration du pouvoir de correction. Les permutations pseudo-aléatoires ont été vues comme les meilleures permutations. Ces permutations assurent des performances supérieures à celles assurées par les autres types de permutations (permutation en blocs, permutation diagonale etc...).

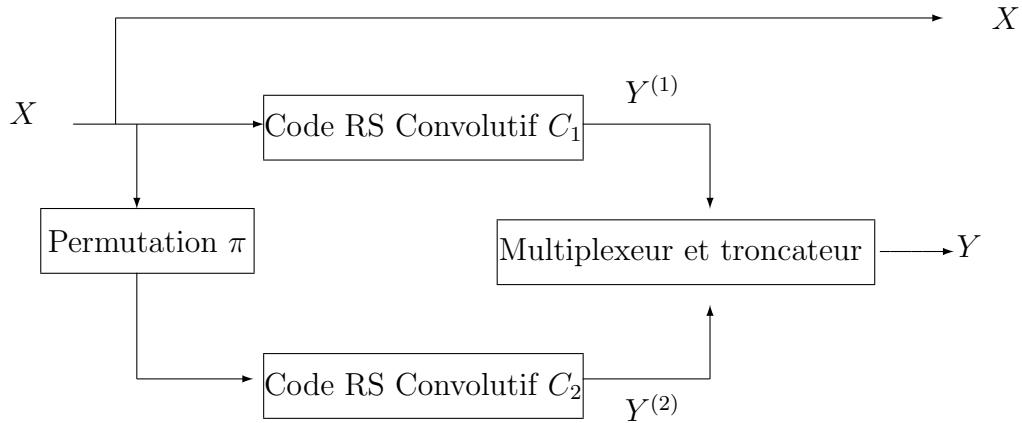


FIGURE I.1 – Turbo Codes : codeur

Les performances des turbo codes sont d'autant meilleurs que le rapport signal à bruit est faible.

#### I.4.2.1 Décodage des Turbo codes

Le décodage des turbo codes utilise les algorithmes comme l'algorithme du MAP (maximum a posteriori), l'algorithme du Log MAP ou l'algorithme du Max-Log-MAP. Le décodage des turbo codes utilise un décodage itératif et une information logique correspondante aux valeurs a priori et a posteriori des bits d'information et des bits de redondances. Le décodage s'arrête après un certain nombre d'itérations à partir du moment où le décodage converge. L'ensemble des opérations effectuées par le décodage des turbo codes sont représentées dans la figure I.2.

Soit  $X = (x_1 x_2 \dots x_k)$  la séquence des bits d'information à l'entrée du codeur, soit  $Y^{(1)} = (y_1^{(1)} y_2^{(1)} \dots y_k^{(1)})$  la séquence de sortie du codeur  $C_1$  et soit  $Y^{(2)} = (y_1^{(2)} y_2^{(2)} \dots y_k^{(2)})$  la sortie du codeur  $C_2$ .

Pour un turbo code de rendement  $R = 1/3$ , le mot de code à la sortie du codeur est de la forme  $Y = (x_1 y_1^{(1)} y_1^{(2)}, x_2 y_2^{(1)} y_2^{(2)}, \dots, x_k y_k^{(1)} y_k^{(2)})$ . Soit  $r = (r_1^{(0)} r_1^{(1)} r_1^{(2)}, r_2^{(0)} r_2^{(1)} r_2^{(2)}, \dots, r_k^{(0)} r_k^{(1)} r_k^{(2)})$ ,

la séquence reçue par le décodeur après transmission, alors cette séquence est divisée en trois séquences  $r^{(0)} = (r_1^{(0)} r_2^{(0)} \dots r_k^{(0)})$ ,  $r^{(1)} = (r_1^{(1)} r_2^{(1)} \dots r_k^{(1)})$  et  $r^{(2)} = (r_1^{(2)} r_2^{(2)} \dots r_k^{(2)})$  correspondant respectivement aux bits d'information erronés, aux bits erronés de redondances de la sortie du codeur  $C_1$  et aux bits erronés de redondances de la sortie du codeur  $C_2$ . Les séquences  $r^{(0)}$  et  $r^{(1)}$  sont données au décodeur  $D_1$  et les séquences  $r^{(0)}$  et  $r^{(1)}$  sont données au décodeur  $D_2$ .

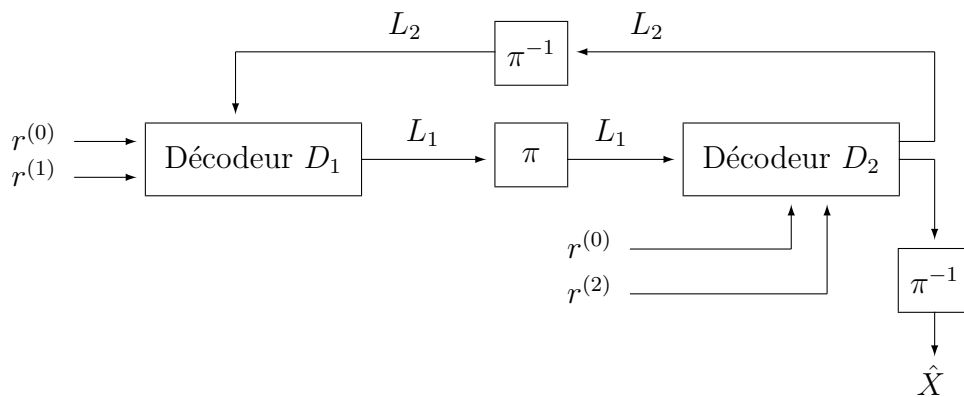


FIGURE I.2 – Turbo Codes : décodeur

A la première itération, le décodeur  $D_1$  reçoit les séquences  $r^{(0)}$  et  $r^{(1)}$  et une probabilité a priori de bits d'information (en général égale à 0 à la première itération, ceci provient du fait que les bits à la sortie du canal de transmission sont considérés équiprobables). Ainsi le décodeur produit des probabilités a posteriori des bits d'information sachant la séquence  $r_1 = (r^{(0)} r^{(1)})$ , ensuite il calcule des probabilités extrinsèques  $L_1$  ( $L_1$  est une séquence de probabilités extrinsèque dont chacune correspond à celle d'un bit d'information) qui seront transmises au décodeur  $D_2$  après permutation et qui seront utilisées comme probabilités a priori des bits d'information à l'entrée du décodeur  $D_2$ . De même le décodeur  $D_2$  reçoit la séquence  $r_2 = (r^{(0)} r^{(2)})$  puis calcule des probabilités extrinsèques  $L_2$  ( $L_2$  est une séquence de probabilités extrinsèque dont chacune correspond à celle d'un bit d'information) à partir des probabilités a posteriori des bits d'information sachant la séquence  $r_2$  et les probabilités a priori des bits d'information  $L_1$ . Après inversion de la permutation, ces probabilités sont fournies au décodeur  $D_1$  comme probabilités a priori des bits d'information. La séquence  $\hat{X}$  à la sortie du décodeur correspond

à une estimation de la séquence d'information  $X$ , si le décodage converge la séquence  $\hat{X}$  sera égale à la séquence  $X$ . Pour plus de détails sur le décodage des turbo code voir [2].

## I.5 Detection et identification

La majorité des travaux réalisés sur la détection et l'identification des signaux s'est servie des classifieurs statistiques comme le modèle de Markov caché (Hidden Markov Model) [13], le modèle de mélange de gaussiennes (Gaussian Mixture Model) [47]. D'autres méthodes pour la détection et la reconnaissance des signaux utilisent les réseaux artificiels de neurones (Artificial Neural Network) et la prédiction linéaire. D'autres ont utilisé l'autocorrection et la détection de pitch pour la détection des signaux ou la détection de périodicité des signaux périodiques comme les signaux d'alarmes.

### I.5.1 Classifieurs Probabilistiques

La majorité des classifieurs probabilistiques calculent la probabilité conditionnelle  $P(\mathcal{C}_i | x)$  pour déterminer la classe la plus probable sachant l'observation  $x$ ,  $1 \leq i \leq N$ , avec  $N$  le nombre de classes  $\mathcal{C}_i$ .

$$P(\mathcal{C}_i | x) = \frac{P(x | \mathcal{C}_i)P(\mathcal{C}_i)}{P(x)} \quad (\text{I.4})$$

L'estimation de  $P(\mathcal{C}_i)$  est en général facile (en général est connue) à l'encontre de l'estimation de  $P(x | \mathcal{C}_i)$ .

Le classifieur ainsi décide la classe vérifiant :

$$\begin{aligned} \hat{\mathcal{C}} &= \arg \max_{1 \leq i \leq N} P(\mathcal{C}_i | x) \\ &= \arg \max_{1 \leq i \leq N} \frac{P(x | \mathcal{C}_i)P(\mathcal{C}_i)}{P(x)} \\ &= \arg \max_{1 \leq i \leq N} P(x | \mathcal{C}_i)P(\mathcal{C}_i). \end{aligned} \quad (\text{I.5})$$

Un modèle de classifieur est composé principalement de deux étapes :

- L'extraction des attributs les plus significatifs des données de telle sorte que les classes des données peuvent être identifiées.
- Un classifieur dont le rôle est de désigner une classe à chaque donnée représentée par ses attributs.

### I.5.1.1 Modèle de mélange de gaussiennes pour la classification

Le modèle de mélange de gaussiennes (GMM : Gaussian Mixture Model) [48], [47] est une densité de probabilité paramétrique représentée comme somme de gaussiennes donnée par l'équation suivante :

$$p(x | \lambda) = \sum_{i=1}^K w_i p(x | \mu_i, \Sigma_i) \quad (\text{I.6})$$

où  $x$  est un vecteur de dimension  $d$ ,  $w_i$  est le poids du mélange, vérifiant  $\sum_{i=1}^K w_i = 1$  et  $p(x | \mu_i, \Sigma_i)$  la densité de probabilité de loi normale multidimensionnelle. Ce modèle est paramétré par  $\lambda = \{w_i, \mu_i, \Sigma_i\}$ , l'ensemble des poids  $w_i$ , des moyennes  $\mu_i$  et des matrices de covariance  $\Sigma_i$  des différentes gaussiennes,  $1 \leq i \leq K$ .

Les paramètres  $\{w_i, \mu_i, \Sigma_i\}$  du modèle sont estimés à partir de l'apprentissage des données utilisant l'algorithme itératif de maximisation de l'espérance [11] (EM : Expectation-Maximization algorithm) et du maximum de vraisemblance (Maximum likelihood).

Pour la classification chaque classe  $\mathcal{C}_j$  est représentée par un modèle de mélange de gaussiennes (GMM) de paramètre  $\lambda_j$ ,  $1 \leq j \leq N$ ,  $N$  le nombre de classes. Pour l'apprentissage du modèle de chaque classe  $\mathcal{C}_j$ , le paramètre  $\lambda_j$  de chaque modèle est déterminé à partir des données de la classe correspondante à ce modèle.

Pour l'extraction de l'information (sur la classe des données observées), le modèle de mélange de gaussiennes fait les hypothèses suivantes :

- Les données proviennent d'un nombre  $N$  de classes connues.
- La probabilité  $P(\mathcal{C}_j)$  est connue.
- Les paramètres  $\lambda_j = [w_{ij}, \mu_{ij}, \Sigma_{ij}]$  de chaque classe sont connus.

Considérons le vecteur  $X = [x_1, x_2, \dots, x_m]$  de  $m$  observations de  $d$  attributs. En supposant que les observations sont indépendantes et identiquement distribuées, alors la probabilité que l'ensemble d'observation  $X$  soit produit par la classe  $\mathcal{C}_j$  est donnée par :

$$P(X | \mathcal{C}_j) = \prod_{k=1}^m p(x_k | \mathcal{C}_j), \quad (\text{I.7})$$

où la probabilité  $p(x_k | \mathcal{C}_j)$  est supposée un mélange de  $K$  multidimensionnelles gaussiennes de la forme de l'équation (I.6) donnée par :

$$\begin{aligned} p(x_k | \mathcal{C}_j) &= p(x_k | \lambda_j) = \sum_{i=1}^K P(x_k | \mu_{ij}, \Sigma_{ij}) P(\mu_{ij}, \Sigma_{ij}) \\ &= \sum_{i=1}^K P(x_k | \mu_{ij}, \Sigma_{ij}) w_{ij}. \end{aligned} \quad (\text{I.8})$$

Pour des classes équiprobables,  $P(\mathcal{C}_j) = \frac{1}{N}$ ,  $0 \leq j \leq N$ , le classifieur choisit la classe vérifiant l'équation :

$$\begin{aligned} \hat{\mathcal{C}} &= \arg \max_{1 \leq j \leq N} P(X | \mathcal{C}_j) P(\mathcal{C}_j) = \arg \max_{1 \leq j \leq N} P(X | \mathcal{C}_j) \\ &= \arg \max_{1 \leq j \leq N} \prod_{k=1}^m p(x_k | \mathcal{C}_j) \\ &= \arg \max_{1 \leq j \leq N} \sum_{k=1}^m \log p(x_k | \mathcal{C}_j). \end{aligned} \quad (\text{I.9})$$

### I.5.1.2 Modèle de Markov caché pour la classification

Un processus de Markov est une séquence d'événements aléatoires  $S_i$  (processus de Markov à temps discret) appelés états qui suivent la propriété de Markov permettant de modéliser l'évolution du processus dans le temps,  $1 \leq i \leq N$ . Soit  $q_n$  l'état du processus



à l'instant  $n$  alors on a :

$$\begin{aligned} P(q_{n+1} = S_j \mid q_n = S_i, q_{n-1} = S_l, \dots, q_0 = S_k) \\ = P(q_{n+1} = S_j \mid q_n = S_i) \end{aligned} \quad (\text{I.10})$$

Le processus subit un changement dans le temps d'un état à un autre, suivant une probabilité appelée probabilité de transition. A l'instant  $n$  le passage du processus de l'état  $q_n$  à l'état  $q_{n+1}$  dépend uniquement du présent. L'expression de la probabilité de transition est donnée par :

$$a_{ij} = P(q_{n+1} = S_i \mid q_n = S_j) \quad (\text{I.11})$$

Un modèle caché de Markov représente un processus stochastique comme un processus de Markov dont les états ne sont pas observables directement mais qui à chaque état associe une probabilité à partir des observations qui ne sont pas forcément Markoviennes [45], [13], [25]. Un modèle caché de Markov est caractérisé par :

- Le nombre  $N$  des états cachés du modèle, on note  $S = [S_1, \dots, S_N]$ , l'ensemble des états du processus.
- Le nombre  $M$  des différents symboles d'observations possibles, la taille de l'alphabet.
- La probabilité de transition  $a_{ij}$  donnée par l'équation I.11 et la matrice de transition  $A$  associée donnée par  $A = \{a_{ij}\}$
- La probabilité que le symbole  $o_k$  d'observation soit émis par l'état  $j$  à l'instant  $n$ , soit  $b_j(k)$  cette probabilité et  $B$  la matrice associée,  $B = \{b_j(k)\}$ . Cette probabilité est appelée probabilité d'émission et est donnée par :

$$b_j(k) = P(o_k \mid q_n = S_j) \quad (\text{I.12})$$

- La distribution  $\pi = \{\pi_i\}$  des états initiaux est donnée par :

$$\pi_i = P(q_1 = S_i) \quad (\text{I.13})$$

Les éléments  $A$ ,  $B$  et  $\pi$  sont appelés les paramètres du modèle en général combinés dans une variable  $\lambda = \{A, B, \pi\}$  pour des raisons pratiques.

Pour le problème de classification fondée sur le modèle caché de Markov, un modèle de paramètre  $\lambda_l$  est créé pour chaque classe  $\mathcal{C}_l$  du classifieur,  $1 \leq l \leq L$ ,  $L$  le nombre de classes. L'apprentissage de chacun de ces modèles est effectué à partir des données de la classe correspondante où les paramètres du modèle sont ajustés pour maximiser la probabilité des séquences observées appartenant à cette classe sachant ces paramètres.

Pour une séquence d'observation  $O = o_1, \dots, o_T$  et la séquence  $Q = q_1, \dots, q_T$  d'états inconnus associée à cette observation, le classifieur calcule la probabilité  $P(O | \lambda_l)$  que la séquence d'observation  $O$  soit produite par le modèle de la classe  $\mathcal{C}_l$  de paramètre  $\lambda_l$ ,  $1 \leq l \leq M$ . Cette probabilité est donnée par :

$$P(O | \lambda_l) = \sum_Q P(O | Q, \lambda_l)P(Q | \lambda_l), \quad (\text{I.14})$$

Le classifieur décide la classe  $\hat{\mathcal{C}}$  de modèle de paramètre  $\hat{\lambda}$  ayant la plus grande probabilité de produire l'observation  $O$ ,  $\hat{\lambda}$  est donnée par :

$$\hat{\lambda} = \arg \max_{1 \leq l \leq L} P(O | \lambda_l) \quad (\text{I.15})$$

Les auteurs des papiers [29], [1] ont proposé une technique de détection et de reconnaissance des signaux acoustiques fondée sur ces classificateurs (Modèles de Markov caché et Gaussian Mixture Models), où des tests ont été effectués sur une base de données contenant six classes des signaux éventuellement compressés. Dans le contexte de [29] (surveillance), le niveau du bruit ambiant n'est pas très élevé par rapport au bruit de trafic.

La plupart des classifieurs statistiques donnent des résultats statistiquement acceptables. Cependant les performances de ces classifieurs se dégradent quand ils sont appliqués aux cas réels, où les signaux sont bruités.

Les auteurs du papier [49] ont utilisé des techniques de réduction du bruit fondée sur l'estimation de la densité spectrale du bruit et sur la suppression du bruit, proposée dans [46], [14], afin d'améliorer leur méthode de classification des signaux fondée sur le modèle de Markov caché.

Des techniques de détection et de reconnaissance des signaux impulsifs adaptées aux changements environnementaux ont été proposées dans [30]. La détection de ces signaux dans ce papier est fondée sur un filtre médian. La reconnaissance de ces signaux est effectuée à l'aide du modèle de Markov caché et du modèle de mélange de gaussiennes. En présence du bruit certains attributs des signaux sont masqués, ainsi, pour résoudre ce problème, des techniques de réduction du bruit fondées sur la soustraction spectrale ont été proposées pour améliorer les performances des systèmes de reconnaissance fondés sur les modèles statistiques (HMM et GMM), voir [53].

## I.5.2 Détection de pitch

La détection de pitch ou fréquence fondamentale a été particulièrement utilisée pour étudier des signaux périodiques ou quasipériodiques comme les signaux de musique, de la parole ou des signaux d'alarme. Il existe plusieurs méthodes de détection de pitch dont certaines sont effectuées dans le domaine fréquentiel et d'autres dans le domaine temporel ou bien les deux ensemble. Un nombre minimum de périodes du signal est nécessaire pour une bonne estimation du pitch. L'une des plus anciennes méthodes utilisées pour la détection de pitch est la fonction d'autocorrélation donnée par l'équation (I.16) (fonction d'autocorrélation pour des signaux discrets).

$$R_{xx}(k) = \sum_{n=k}^{N+k-1} x(n)x(n-k) \quad (\text{I.16})$$

Le maximum de la fonction d'autocorrélation est obtenu quand le retard  $k$  est égal à zéro. La localisation  $\mathcal{L}$  du prochain maximum donne une estimation de la période  $T = \mathcal{L}T_s$  du signal, ainsi la fréquence fondamentale  $f_0$  est donnée par :

$$f_0 = \frac{1}{T}, \quad (\text{I.17})$$

avec  $T_s$  la période d'échantillonnage.

D'autres méthodes comme l'algorithme On-The-Fly [10] utilisant la fonction différence à l'équation (I.18) ont été utilisées pour l'estimation du pitch où  $f_0$  doit être localisée au minimum global de la fonction différence. Il a été montré que cet algorithme est plus efficace que la méthode de corrélation.

$$d(k) = \sum_{n=k}^{N+k-1} (x(n) - x(n-k))^2 \quad (\text{I.18})$$

Si  $\tau$  est le retard minimisant la fonction différence  $d(k)$ , alors  $\tau T_s$  est égal à la période du pitch et la fréquence fondamentale  $f_0$  est donnée par :

$$f_0 = \frac{1}{\tau T_s}, \quad (\text{I.19})$$

Dans le papier [34] les auteurs ont proposé un algorithme de détection des signaux périodiques en présence du bruit, plus précisément, les signaux produits par des sirènes et alarmes de voitures prioritaires. Cet algorithme est fondé sur la détection des pitches.

D'autres papiers comme [28] et [7] ont utilisé ces techniques pour la détection des signaux. Dans le papier [28] deux méthodes de détection des signaux d'alarmes dans quatre types de bruit de fond ont été proposées ; le bruit de trafic, de musique, des parcs et des cafétérias. La première méthode utilise la technique d'auto-corrélation des signaux et la deuxième est fondée sur le changement du niveau sonore. Les résultats ont montré que la combinaison de ces deux méthodes peut améliorer la performance des systèmes de détection des signaux d'alarme pour les personnes qui ont des difficultés d'audition. Les auteurs du papier indiquent un taux de 80% de détection pour un taux de 25 % de fausse détection.

Le papier [7] a aussi utilisé la fonction d'auto-corrélation pour développer un appareil (pour l'oreille) permettant la détection des signaux d'alarme. D'autres techniques de traitement des signaux ont été ajoutées à cette technique : un filtre passe-bande pour sélectionner les fréquences fondamentales des signaux d'alarmes, un détecteur de l'enveloppe spectrale et un algorithme évaluateur de périodicité. Le système proposé n'est pas efficace pour l'identification des signaux d'alarmes de deux tons et des signaux de sirènes.

### I.5.3 Le réseau artificiel de neurones

Inspiré du réseau de neurones biologique, le réseau artificiel de neurones a été conçu pour résoudre une variété de problèmes : reconnaissance de motif, prédiction etc [22].

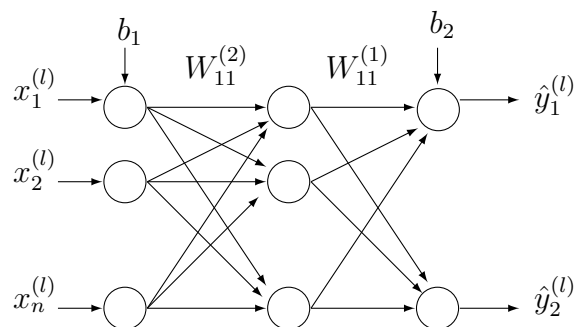


FIGURE I.3 – Le réseau artificiel de neurones : perceptron multicouche

Le réseau de neurones peut être vu comme un graphe orienté avec des poids dans lequel les noeuds correspondent aux neurones artificiels et les arcs orientés correspondent aux connexions entre les neurones de sortie et les neurones d'entrée, voir la figure I.3. Selon son architecture le réseau artificiel de neurones peut être divisé en deux groupes :

- le réseau Feed forward dans lequel chaque graphe n'a pas de boucle
- le réseau Feedback ou réseau récurant de neurones dans lequel une boucle se produit à cause du feedback, voir [22] pour plus de détail.

Dans cette section nous parlons du perceptron multicouche qui est l'une des familles du réseau feedforward. Dans le multicouche perceptron, les neurones sont organisés en couches et les liens entre les couches sont unidirectionnels.

Appliqué au problème de classification, le perceptron à une couche est limité car uniquement des limites linéaires peuvent être créées entre les classes (régions des attributs). Cette limite peut être évitée en utilisant plusieurs couches d'où le terme perceptron multicouche (Multilayers perceptron MLP). En pratique, trois couches sont suffisantes (une couche d'entrée, une couche cachée et une couche de sortie).

La sortie de la dernière couche peut être considérée comme la sortie du réseau (pour un perceptron à trois couches). Dans le multicouche perceptron à trois couches le nombre de neurones à la sortie de la seconde couche correspond en général au nombre de classes considérées. Le nombre de neurones de la deuxième couche dépend en général du problème à résoudre.

Soit  $\{(x^{(1)}, y_1), \dots, (x^{(l)}, y^{(l)}), \dots, (x^{(M)}, y^{(M)})\}$  l'ensemble des données d'apprentissage où  $x^{(l)} = (x_1^{(l)}, \dots, x_i^{(l)}, \dots, x_n^{(l)})$  est un vecteur d'attributs de dimension  $n$  correspondant à l'entrée du réseau et  $y^{(l)}$  le label (cible). La sortie de chaque couche  $k$  du réseau est donnée par :

$$z_j^{(k+1)} = f\left(\sum_i^n w_{ij}^{(k)} z_i^{(k)} + b_k\right),$$

pour  $1 \leq k \leq 3$  (dans le cas d'un multicouche perceptron à trois couches),  $k$  l'indice des couches du perceptron avec  $z^{(1)}$  l'entrée du réseau ( $x^{(l)}$ ) et  $b_k$  appelé biais.  $b_k$  et  $w_{ij}^{(k)}$  sont appelés les paramètres du réseau. La fonction  $f$  est appelée fonction de validation en général  $f$  est une fonction sigmoïde.

Pour l'apprentissage du réseau artificiel de neurones, il faut déterminer les paramètres du réseau  $b_k$  et  $w_{ij}^{(k)}$  qui correspondent le plus aux données d'apprentissage, ainsi pour

cette fin le réseau artificiel de neurones utilise l'algorithme du backpropagation.

Soit  $\hat{y}_j^{(l)}$  la  $j$ -ème sortie du réseau correspondant à l'estimation de  $y_j^{(l)}$  pour une entrée  $x^{(l)}$  du réseau, l'algorithme du backpropagation détermine pour chaque couche  $k$  les poids  $w_{ij}^{(k)}$  correspondant au poids de connexion entre le neurone  $i$  de la couche  $k$  et le neurone  $j$  de la couche  $k + 1$  et le biais  $b_k$  de telle sorte que la fonction coût  $\mathcal{E}(x^{(l)})$  ( $\mathcal{E}$  est en général la somme des erreurs quadratiques ou l'erreur quadratique moyenne) soit minimale :

$$\mathcal{E}(x^{(l)}) = \sum_{l=1}^M e_l = \sum_{l=1}^M \sum_{j=1}^m (\hat{y}_j^{(l)} - y_j^{(l)})^2$$

Cet algorithme utilise la descente du gradient pour trouver le minimum de la fonction  $\mathcal{E}$ , les performances du réseau de neurones sont améliorées à l'aide d'une approche itérative permettant de mettre à jour les poids  $w_{ij}^{(k)}$  à chaque itération, jusqu'à l'obtention des poids minimisant l'erreur  $\mathcal{E}$ .

Pour la classification avec un multicouche perceptron à trois couches, la  $j$ -ème sortie de la deuxième couche pour une entrée  $x^{(l)}$  est donnée par :

$$y_j^{(l)} = f\left(\sum_i^n w_{ij}^{(2)} z^{(2)i} + b_2\right),$$

avec  $1 \leq j \leq N$ ,  $N$  le nombre de neurones de la dernière couche. Avec la dernière couche est ajouté un seuil de décision au dessus duquel une classe est étiquetée 1 et en dessous duquel la classe est étiquetée 0. Ainsi si la classe  $j$  est étiquetée 1 ( $y_j^{(l)}$ , la  $j$ -ème sortie de la deuxième couche est supérieure au seuil de décision), alors  $x^{(l)}$  appartient à la classe  $\mathcal{C}_j$ . Un faible seuil augmente le taux de fausse détection et un grand seuil diminue le taux de détection.

Cette technique de classification a été utilisée par les auteurs du papier [4] afin de concevoir un système de détection automatique des signaux d'alarmes. Les coefficients cepstraux des fréquences de Mel ( MFCCs : Mel-frequency Cepstral Coefficients ) [35],

[44] ont été utilisés comme attributs pour effectuer la classification. Les auteurs du papier déclarent un taux de 99% d'exactitude pour des rapports signal à bruit au dessus de 0 dB.





# Chapitre II

## Détection et identification d'alarme des véhicules prioritaires par étude du sonagramme

Dans ce chapitre nous proposons une méthode de détection et d'identification des signaux d'alarmes de voitures prioritaires, fondée sur les propriétés fréquentielles de ces signaux.

Les signaux d'alarmes des voitures prioritaires sont constitués d'une transmission alternée de deux signaux harmoniques, voir la figure II.1. Les propriétés spectro-temporelles de ces signaux sont exploitées pour faire la détection. Ces propriétés sont aussi utilisées pour l'identification de ces signaux. La représentation au niveau de la figure II.1 est valable pour la plupart des signaux d'alarmes des voitures prioritaires : sapeurs-pompiers, SAMU, gendarmerie et police. Une alarme est caractérisée par les paramètres  $T_1$  et  $T_2$ , représentant les durées des deux tonales, et  $f_1$  et  $f_2$ , les fréquences fondamentales correspondantes.

La figure II.1 représente la projection sur le plan temps-fréquence du sonagramme d'un signal d'alarme, cette projection représente la signature de ce signal. Cette signature se répète à toutes les harmoniques  $kf_1$ ,  $kf_2$ ,  $k=1,2,3\dots$  du signal.

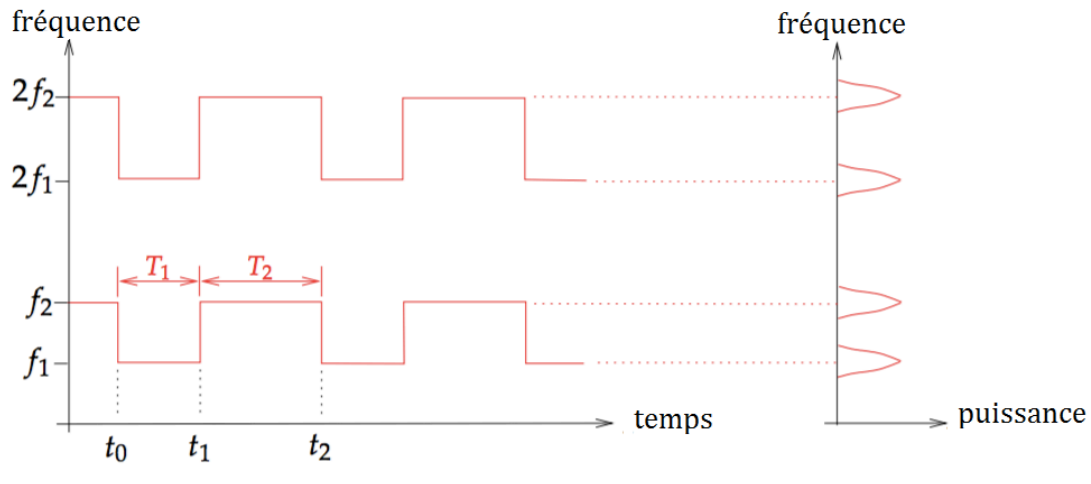


FIGURE II.1 – A gauche, représentation dans le plan temps-fréquence du sonagramme d’un signal d’alarme, à droite la projection sur le plan fréquence-puissance du sonagramme de l’alarme

Ainsi nous allons procéder à la détection et l’identification de ces signaux en exploitant les propriétés de cette représentation temps-fréquence.

## II.1 Recherche de la signature d’une alarme

Une alarme est présente lorsque l’on retrouve sa signature temps-fréquence. Autrement dit, une alarme sera détectée si le sonagramme du signal enregistré admet une distribution d’énergie maximale au niveau des harmoniques des deux fréquences fondamentales  $f_1$  et  $f_2$  et s’il présente une structure comme celle de la figure II.1.

La figure II.1 représente des courbes identiques dont chacune alterne d’une harmonique  $kf_1$  à l’harmonique  $kf_2$  de l’alarme,  $k=1,2,3\dots$  (courbes de gauche de la figure II.1). En effectuant une agrégation sur l’axe des fréquences de la figure II.1, on obtient une courbe. Cette courbe représentera la signature temporelle du signal d’alarme. De même si l’on effectue une projection sur le plan fréquence-puissance du sonagramme, nous obtiendrons une structure comme celle de la courbe de droite de la figure II.1. Cette courbe présente des maxima (une énergie maximale) au niveau des harmoniques des fréquences fondamen-

tales de l'alarme. La présence de ces signatures dans un signal permettra de détecter la présence d'un signal d'alarme.

La recherche d'un signal d'alarme est composée de deux étapes ; une recherche de sa signature fréquentielle et une recherche de sa signature temporelle.

### II.1.1 Recherche de la signature fréquentielle

Le signal est découpé en blocs de longueur  $N=2048$ , pondérés par une fenêtre de Hamming. Le calcul de la transformée de Fourier à court-terme de chaque bloc produit le sonagramme : image indexée par le temps et la fréquence, où chaque pixel de coordonnées  $(t, f)$  indique la puissance du signal à l'instant  $t$  et à la fréquence  $f$ . Le sonagramme de la figure II.2 est celui d'un signal d'alarme d'une voiture de sapeurs-pompiers.

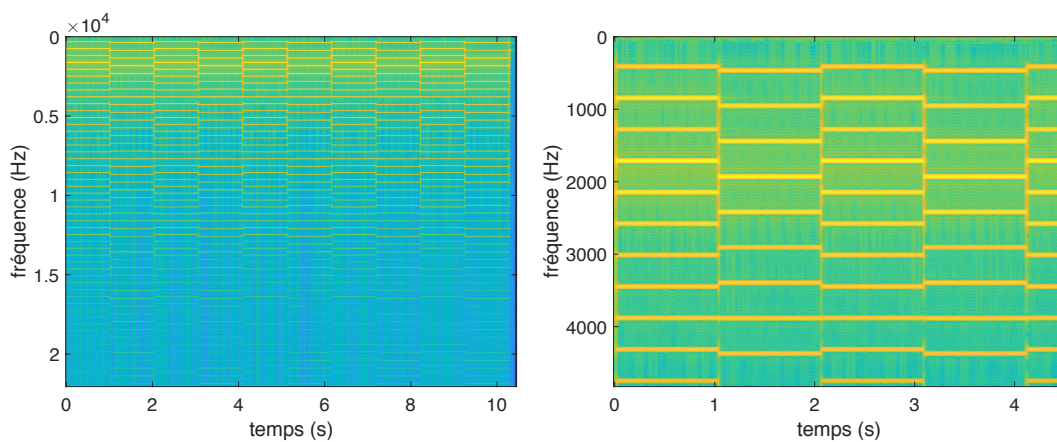


FIGURE II.2 – À gauche sonagramme d'un signal d'alarme. À droite zoom sur le sonagramme

La structure de la signature fréquentielle du signal d'alarme est caractérisée par la présence d'une énergie maximale au niveau des harmoniques des fréquences fondamentales du signal, voir la courbe de droite de la figure II.1. Soit  $\Delta$  la durée d'un intervalle de temps du signal. Pour chaque fréquence  $f$  fixée, on définit par  $\Phi(f, t)$ , le module du sonagramme

à cette fréquence  $f$ , intégré depuis un passé de longueur  $\Delta$  jusqu'au temps  $t$ . On obtient ainsi une fonction temps-fréquence donnée par :

$$\Phi(f, t) = \sum_{\tau=t-\Delta}^t |X(f, \tau)|. \quad (\text{II.1})$$

où  $X(f, t)$  représente la transformée de Fourier à court-terme du signal. Si une alarme est présente dans l'intervalle de temps  $[t_0 - \Delta, t_0]$  alors le tracé de  $\Phi(f, t_0)$  doit être de la forme de la courbe de gauche donnée par la figure II.3. L'intervalle d'observation  $\Delta$  doit au moins contenir une période du signal ( $T_1 + T_2$ ). En effet, si ce n'est pas le cas, l'une des deux fréquences fondamentales ne pourra pas être observée. Ainsi, dans toute la suite,  $\Delta$  sera choisi tel que  $\Delta > T_1 + T_2$ . Dans ces conditions,  $\Phi(f, t_0)$  doit admettre des pics au niveau des fréquences  $kf_1$  et  $kf_2$ ,  $k = 1, 2, \dots$ , correspondant aux harmoniques des deux fréquences fondamentales de l'alarme.

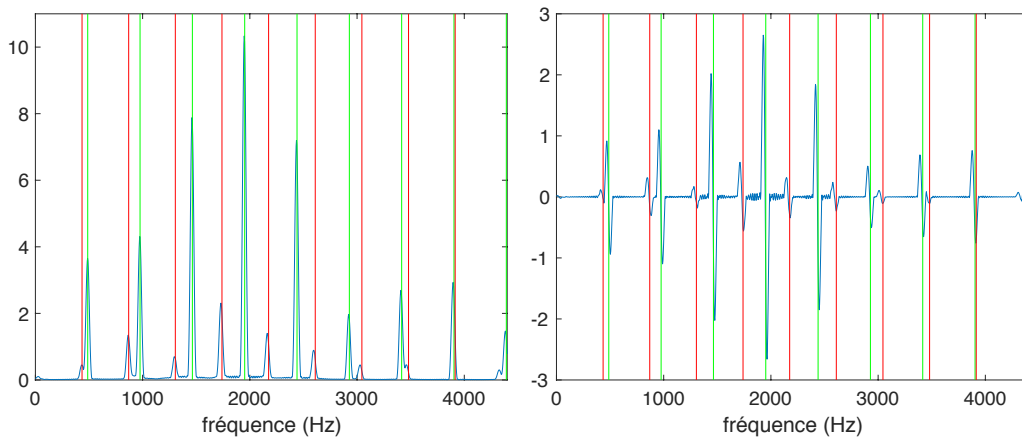


FIGURE II.3 – Signature fréquentielle : à gauche  $\Phi(f, t)$ , à droite sa dérivée,  $t=0$  s

Pour déterminer les maxima de  $\Phi(f, t)$  pour un  $t$  donné, nous utilisons la dérivée du premier ordre de  $\Phi(f, t)$  par rapport à la fréquence  $f$ . Si pour une fréquence donnée la dérivée de  $\Phi(f, t)$  passe brutalement d'une valeur positive ou nulle à une valeur négative, alors  $\Phi(f, t)$  admet un maximum en cette fréquence. Le tracé de la dérivée de  $\Phi(f, t)$  est donné par la courbe de droite de la figure II.3. La courbe de gauche de la figure II.3 représente  $\Phi(f, t)$  d'un signal de sapeurs-pompiers, cette courbe présente des maxima qui

se trouvent au niveau des harmoniques  $h_{1k} = kf_1$ ,  $h_{2k} = kf_2$ ,  $k = 1, 2, 3, \dots$ , de l'alarme. Les lignes verticales de couleur rouge de la figure II.3 sont positionnées au niveau des harmoniques de  $f_1$  et celles de couleur verte sont positionnées au niveau des harmoniques de  $f_2$ .

Supposons maintenant qu'on dispose d'un signal et qu'on veuille savoir si ce signal contient la signature fréquentielle d'une alarme de fréquences fondamentales  $f_1$  et  $f_2$ . Dans un premier temps la projection  $\Phi(f, t)$  de ce signal sera calculée, puis une recherche des maxima de cette fonction sera effectuée.

Parmi ces maxima on cherche des fréquences  $F$ , multiples de  $f_1$  et/ou  $f_2$  à une erreur près :

$$F = \begin{cases} \lambda f_1 + \varepsilon \\ \text{et/ou} \\ \lambda' f_2 + \varepsilon \end{cases} \quad (\text{II.2})$$

où  $\varepsilon \in \mathbb{R}$ , une tolérance donnée et  $(\lambda, \lambda') \in \mathbb{N}^2$ .

Le déplacement du véhicule prioritaire produit un déplacement des fréquences du signal d'alarme. Ce déplacement correspond à une fréquence  $f_D$  donnée par l'équation (II.3).

$$f_D = \frac{(\vec{u} \vec{v}_r - \vec{u} \vec{v}_s) f_s}{c - \vec{u} \vec{v}_s} \quad (\text{II.3})$$

où  $\vec{v}_s$  et  $f_s$  désignent respectivement le vecteur vitesse de la source et la fréquence de la source,  $\vec{v}_r$  désigne le vecteur vitesse du récepteur,  $\vec{u}$  correspond au vecteur du point source au point récepteur normalisé par la distance entre la source et le récepteur et  $c$  la vitesse de propagation.

Cet effet est connu sous le nom d'effet Doppler [8]. La grande majorité des papiers qui traitent le problème de détection des signaux d'alarme des voitures prioritaires ne prennent pas cet effet en considération.

La tolérance  $\varepsilon$  qu'on introduit au niveau de l'équation (II.2) permet de trouver les maxima au niveau des harmoniques de l'alarme même si ces derniers ont été décalés par l'effet Doppler. Ainsi la présence des maxima au niveau des harmoniques de l'alarme décalées d'une fréquence  $f_D$  indiquera la présence de la signature fréquentielle de l'alarme pour  $f_D < \varepsilon$ .

Si l'on trouve au moins des maxima au niveau de deux harmoniques  $\lambda_1 f_1$  et  $\lambda_2 f_1$  de  $f_1$  et au niveau de deux harmoniques  $\lambda'_1 f_2$  et  $\lambda'_2 f_2$  de  $f_2$  tel que  $\lambda_1 = \lambda'_1$  et  $\lambda_2 = \lambda'_2$ , alors on détecte la signature de l'alarme.

### II.1.2 Recherche de la signature temporelle

Dans cette partie nous exploitons la structure temps-fréquence du sonagramme afin de déterminer la signature temporelle d'un signal d'alarme, voir la figure II.1. Cette figure représente des courbes identiques qui se répètent au niveau des harmoniques des deux fréquences fondamentales du signal. Ces courbes présentent des discontinuités au niveau des instants  $t_k$ .

En agrégeant ces courbes sur la dimension fréquentielle on met davantage en évidence les discontinuités tout en lissant les parties continues, soit  $p(t)$  cette aggrégation,  $p(t)$  est donnée par l'équation (II.4).

$$p(t) = \sum_f |X(f, t)|. \quad (\text{II.4})$$

La courbe gauche de la figure II.4 représente  $p(t)$ , d'un signal d'alarme de sapeurs-pompiers. En dérivant cette nouvelle courbe on aura des maxima aux instants  $t_k$  correspondant aux discontinuités ; la présence de ces maxima au niveau des instants  $t_k$  permet de retrouver les caractéristiques temporelles  $T_1$  et  $T_2$  du signal d'alarme, voir la courbe droite de la figure II.4. Avant d'effectuer la dérivée, il convient de lisser la courbe au préalable pour éviter de confondre les fluctuations dues aux bruits, avec les sauts. Les

instants de discontinuités  $t_k$  sont liés aux durées  $T_1$  et  $T_2$  des deux tons de l'alarme par les relations suivantes :

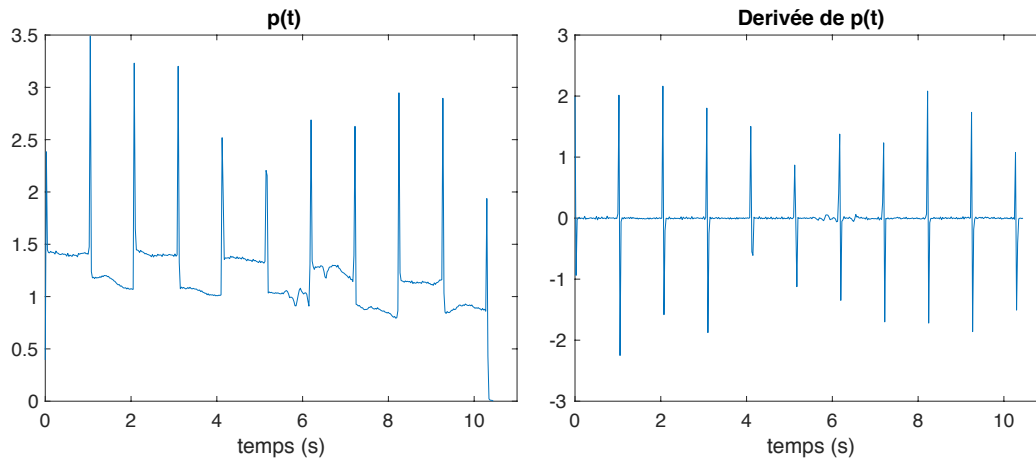


FIGURE II.4 – Signature temporelle :  $p(t)$  et sa dérivée

$$t_1 = t_{2k+1} = t_0 + T_1, \quad k = 0;$$

$$t_2 = t_{2k} = t_1 + T_2 = t_0 + T_1 + T_2, \quad k = 1;$$

$$t_3 = t_{2k+1} = t_2 + T_1 = t_0 + (T_1 + T_2) + T_1, \quad k = 1;$$

$$t_4 = t_{2k} = t_3 + T_2 = t_0 + 2(T_1 + T_2), \quad k = 2;$$

$$\begin{cases} t_{2k} &= t_0 + k(T_1 + T_2), \\ t_{2k+1} &= t_0 + k(T_1 + T_2) + T_1, \end{cases} \quad (\text{II.5})$$

avec  $t_0=0$  s, l'instant du commencement du signal.

$$T_1 = t_1 - t_0 = t_3 - t_2 = \dots = t_{2k+1} - t_{2k}, \quad k \geq 0,$$

$$T_2 = t_2 - t_1 = t_4 - t_3 = \dots = t_{2k} - t_{2k-1}, \quad k \geq 1.$$



$$\begin{cases} T_1 &= \frac{1}{N} \sum_{k=0}^N t_{2k+1} - t_{2k}, \\ T_2 &= \frac{1}{N} \sum_{k=1}^N t_{2k} - t_{2k-1}, \end{cases} \quad (\text{II.6})$$

avec  $N$ , le nombre des instants  $t_{2k+1}$  et des instants  $t_{2k}$ .

Supposons qu'on soit dans la situation où on ne sait pas s'il s'agit d'un signal d'alarme ou non, alors dans cette situation les instants correspondant aux maxima de la dérivée de  $p(t)$  de ce signal seront considérés les instants  $t_k$ . Si les instants détectés permettent d'estimer les paramètres  $T_1$  et  $T_2$  de l'alarme cherchée, alors on peut dire que la signature temporelle de l'alarme cherchée est présente.

Soit  $\hat{T}_1$  et  $\hat{T}_2$  les estimations respectives de  $T_1$  et de  $T_2$ . Posons  $L$  le nombre des instants correspondant aux maxima de la dérivée de  $p(t)$ , alors le nombre des instants  $t_{2k}$  et le nombre de  $t_{2k+1}$  sont équivalents et correspondent à la moitié du nombre  $L$ . Soit  $C = \lceil \frac{L}{2} \rceil$  ce nombre ( $\lceil \cdot \rceil$  la partie entière), alors on a les relations suivantes :

$$\begin{cases} t_{2k} &= t_0 + k(\hat{T}_1 + \hat{T}_2), \\ t_{2k+1} &= t_0 + k(\hat{T}_1 + \hat{T}_2) + \hat{T}_1 \end{cases} \quad (\text{II.7})$$

$$\begin{cases} \hat{T}_1 &= \frac{1}{C} \sum_{k=0}^C (t_{2k+1} - t_{2k}) = T_1 + \varepsilon' \\ \hat{T}_2 &= \frac{1}{C} \sum_{k=1}^C (t_{2k} - t_{2k-1}) = T_2 + \varepsilon' \end{cases} \quad (\text{II.8})$$

où  $\varepsilon'$  est une erreur correspondant à une précision tolérable.

Si la valeur de  $\varepsilon'$  est très petite, alors  $p(t)$  du signal est de la forme de la courbe de gauche de la figure II.4, ainsi on détecte la signature temporelle d'une l'alarme.

### II.1.3 Identification de la bande de fréquence la moins bruitée

Dans cette partie nous avons optimisé et rendu plus robuste le travail réalisé à la section précédente sur la recherche de la signature temporelle de l'alarme (estimation des paramètres  $T_1$  et  $T_2$  du signal).

Auparavant tout le sonagramme a été pris en considération y compris les zones où il n'y a que du bruit, ce qui rend la précision de l'estimation des paramètres  $T_1$  et  $T_2$  moins bonne.

L'estimation des paramètres  $T_1$  et  $T_2$  est basée sur la recherche des positions des maxima de  $p(t)$ , où  $p(t)$  est calculée sur l'intervalle de temps  $\Delta$ . Comme les valeurs estimés de  $T_1$  et  $T_2$  correspondent à la moyenne de la différence entre deux instants de ruptures de  $p(t)$ , voir l'équation (II.8), alors il est convenable de prendre cet intervalle le plus grand possible pour faire une bonne estimation de ces paramètres.

Comme la méthode utilisée pour la détection des maxima de  $p(t)$  repose sur la dérivée par rapport à  $t$ , alors le bruit perturbe la recherche de ces maxima. Pour rendre l'estimation de ces paramètres plus précise, il convient de traiter le signal uniquement dans la bande de fréquences du sonagramme contenant le moins de bruit.

Pour cela on a utilisé la mesure de la platitude spectrale du signal [24] (spectral flatness measure,  $sfm_{dB}$ ) qui permet de calculer le coefficient de tonalité  $\alpha$  du signal.

$$\alpha = \min \left( \frac{sfm_{dB}}{SFM_{dB}}, 1 \right),$$

où  $SFM_{dB} = -60$  dB représente la platitude d'un signal tonal pur.  $sfm_{dB}$  est donnée par :

$$sfm_{dB} = 10 \log_{10} \frac{mG}{mA},$$

Les paramètres  $mG$  et  $mA$  représentent respectivement la moyenne géométrique et la moyenne arithmétique du spectre de puissance du signal. Ce coefficient est maximal dans les bandes de fréquences où le bruit est le plus faible, et est minimal dans les bandes de fréquences où le bruit domine.

L'axe fréquentiel est divisé en plusieurs bandes de fréquences, dont chacune contient au moins deux harmoniques du signal (des bandes de largeur 10 kHz). Ainsi la bande de

fréquences ayant la plus grande valeur du coefficient  $\alpha$  est retenue, et l'estimation des paramètres  $T_1$  et  $T_2$  sera effectuée dans cette bande de fréquences.

#### II.1.4 Algorithme de détection et d'identification conjointes d'alarmes

Dans cette partie nous proposons un algorithme de détection et d'identification des signaux d'alarme fondé sur l'étude effectuée dans les deux sections précédentes II.1.2 et II.1.1. La recherche de la signature temporelle et la recherche de la signature fréquentielle d'un signal d'alarme ont été combinées dans un seul algorithme 1 permettant de détecter la présence d'un signal d'alarme.

L'indicateur prend la valeur  $r=3$  si l'on détecte la présence des deux signatures, fréquentielle et temporelle du signal d'alarme cherché. Cependant si l'on détecte uniquement la présence de la signature temporelle du signal d'alarme, cet indicateur prend la valeur  $r = 1$ , et prend la valeur  $r = 2$  si l'on détecte uniquement la présence de la signature fréquentielle du signal d'alarme.

La détection de l'alarme repose sur la comparaison de  $r$  à un seuil de 1,2 ou 3. Dans le cas réel les signaux d'alarme sont bruités par le bruit de trafic, ce bruit peut rendre l'estimation des paramètres  $T_1$  et  $T_2$  très difficile; la dérivée d'un signal bruité contient des pics partout et donc il sera difficile de déterminer les instants de rupture de  $p(t)$ .

Par conséquent la signature temporelle ne sera pas toujours détectable. Ainsi en cas de détection de cette signature, cette signature servira pour renforcer la décision de l'algorithme. En cas d'absence de cette signature, la signature fréquentielle uniquement sera considérée comme indicateur de présence ou d'absence de l'alarme. Si en plus de la présence de la signature fréquentielle on détecte la présence de la signature temporelle, alors la probabilité de faire une fausse détection d'une alarme sera faible.

Comme les fréquences fondamentales des différents signaux d'alarmes des voitures

---

**Algorithme 1** : Recherche de signature d'une alarme

---

**Entrée** :

- $x(t)$  : le signal enregistré au microphone.
- $f_1, f_2$  : les deux tonales de l'alarme.
- $T_1, T_2$  : durées des deux tonales dans un cycle.

- 1 Initialiser  $r_t=0, r = 0, r_f=0$ ;
  - 2 Calculer  $X(f, t)$  le sonogramme du signal  $x(t)$ ;
  - 3 Calculer  $\Phi(f, t) = \sum_{\tau=t-\Delta}^t |X(f, \tau)|, \Delta > T_1 + T_2$ ;
  - 4 Chercher les maxima de  $\Phi(f, t)$ ;
  - 5 Extraire les fréquences correspondantes à ces maxima;
  - 6 Chercher parmi ces fréquences des fréquences  $F$ , multiples de  $f_1$  et/ou  $f_2$   
$$F = \begin{cases} \lambda f_1 + \varepsilon \\ \lambda' f_2 + \varepsilon \end{cases};$$
  - 7  $\Gamma_1 = \{\lambda \in \mathbb{N}, F = \lambda f_1 + \varepsilon\}$ ;
  - 8  $\Gamma_2 = \{\lambda' \in \mathbb{N}, F = \lambda' f_2 + \varepsilon\}$ ;
  - 9  $\mu = \text{card}(\Gamma_1 \cap \Gamma_2)$ ;
  - 10 **if**  $\mu \geq 2$  **then**
  - 11 |  $r_f=2$ ;
  - 12 **end**
  - 13 Chercher la bande de fréquence  $B$  contenant moins de bruit;
  - 14 Calculer  $p(t)$  dans cette bande;
  - 15  $p(t) = \sum_{f \in B} |X(f, t)|$ . Calculer la dérivée de  $p(t)$ ;
  - 16 Déterminer les maxima  $t_i$  de la dérivée de  $p(t)$ ;
  - 17 Calculer  $C = \lceil \frac{L}{2} \rceil$ ,  $L$  le nombre des maxima;
  - 18 Calculer  $\hat{T}_1$  et  $\hat{T}_2$  ;
  - 19 
$$\begin{cases} \hat{T}_1 = \frac{1}{C} \sum_{k=0}^C (t_{2k+1} - t_{2k}) \\ \hat{T}_2 = \frac{1}{C} \sum_{k=1}^C (t_{2k} - t_{2k-1}) \end{cases}$$
  - 20 **if**  $|\hat{T}_1 - T_1| \leq \epsilon'$  et  $|\hat{T}_2 - T_2| \leq \epsilon'$  **then**
  - 21 |  $r_t=1$ ;
  - 22 **end**
  - 23  $r = r_t + r_f$ ;
- Sortie** : Une réponse  $r$  indiquant la présence ou l'absence de la signature de l'alarme recherchée.
-

prioritaires sont proches les unes des autres, voir la table II.1, pour une identification de ces signaux fondée sur ces fréquences en présence de l'effet Doppler on risque de confondre les fréquences fondamentales des alarmes. Il en résultera des fausses identifications. Dans le cas de la détection de la signature temporelle en plus de la détection de la signature fréquentielle, la combinaison des paramètres  $f_1$ ,  $f_2$ ,  $T_1$  et  $T_2$  des signaux d'alarmes est différente d'une alarme à l'autre. Cette combinaison sera utilisée pour renforcer l'identification de ces signaux.

TABLE II.1 – Fréquences fondamentales d'alarmes de voitures prioritaires

<b>Alarme</b>	<b>Fréquences fondamentales</b>
Police de France	$f_1 = 435$ Hz $f_2 = 580$ Hz
Gendarmerie de France	$f_1 = 435$ Hz $f_2 = 732$ Hz
Sapeurs-pompiers de France	$f_1 = 435$ Hz $f_2 = 488$ Hz
UMH/SAMU de France	$f_1 = 435$ Hz $f_2 = 651$ Hz
Ambulance de France	$f_1 = 420$ Hz $f_2 = 516$ Hz
Police d'Italie	$f_1 = 466$ Hz $f_2 = 622$ Hz
Ambulance et sapeurs-pompiers de l'Italie	$f_1 = 392$ Hz $f_2 = 660$ Hz
Police, Ambulance et sapeurs-pompiers de l'Allemagne et de la suisse	$f_1 = 430$ Hz $f_2 = 570$ Hz

## II.2 Application à la détection et l'identification d'un signal d'alarme d'une voiture de pompiers

### II.2.1 Signal peu bruité

L'algorithme précédent a été appliqué à une scène audio contenant un signal d'alarme de voiture de pompiers peu bruité. Le sonagramme du signal enregistré (voir la figure II.5) montre une signature d'alarme comme celle de la figure II.1 (a) avec du bruit qui s'y ajoute dû au passage des voitures, aux klaxons, au bruit de trafic, etc. Les signaux d'alarme utilisés dans ce chapitre sont échantillonnés à une fréquence d'échantillonnage  $F_s = 44100$  Hz.

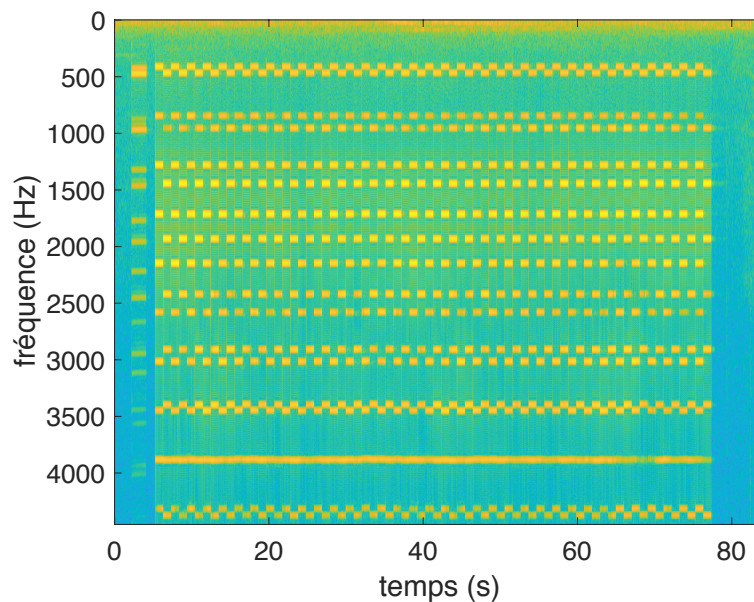


FIGURE II.5 – Sonagramme d'un signal d'alarme de sapeurs-pompiers, d'une durée de 78 s, de fréquences fondamentales  $f_1=435$  Hz,  $f_2=488$  Hz, de durées respectives  $T_1 = 1.1$  s et  $T_2 = 1.2$  s

Nous remarquons sur la figure II.5 qu'entre les instants 0 s et 7 s la distribution d'énergie ne correspond pas à celle d'un signal d'alarme. Cette zone ne contient pas d'information utile, de même pour l'intervalle de temps compris entre les instants 78 s et 80s, l'information entre ces deux instants n'est pas utile. Par contre entre les instants 7 s et

78 s nous observons la présence de la signature de l'alarme.

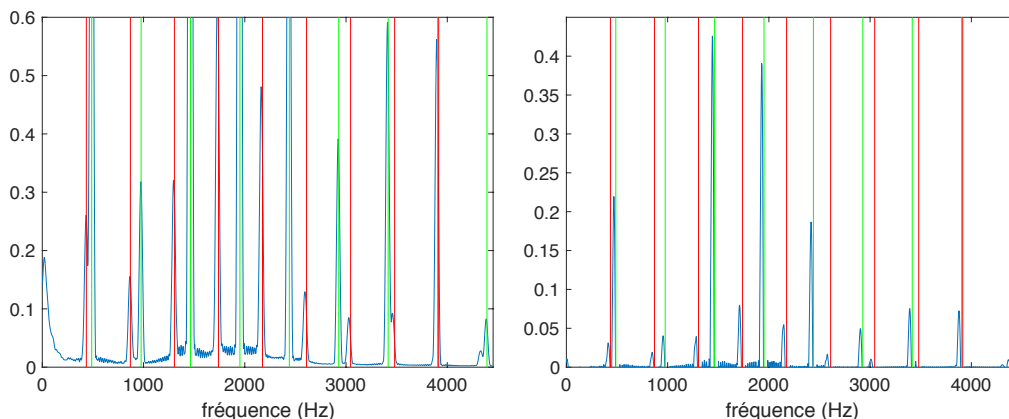


FIGURE II.6 – À gauche la projection  $\Phi(f, t)$ . À droite la dérivée de  $\Phi(f, t)$ ,  $t=0$  s. Les lignes verticales de couleur rouge indiquent les positions de  $f_1=435$  Hz et de ses harmoniques et ceux de couleur verte indiquent les positions de  $f_2=488$  Hz et de ses harmoniques

La figure II.6 représente le tracé de  $\Phi(f, t)$ . Les pics de  $\Phi(f, t)$  sont au niveau des harmoniques de  $\hat{f}_1=432$  Hz et  $\hat{f}_2=487$  Hz pour  $f_1=435$  Hz et  $f_2=488$  Hz ;  $\hat{f}_1$  et  $\hat{f}_2$  sont les estimés de  $f_1$  et  $f_2$ .

Le couple  $(f_1, f_2)$  de fréquences fondamentales d'alarme le plus proche de  $(\hat{f}_1, \hat{f}_2)$  est celui d'une alarme d'un véhicule de sapeurs-pompiers. La fonction  $\Phi(f, t)$  permet donc bien, ici, de détecter et d'identifier l'alarme.

La figure II.7 représente la fonction  $p(t)$  du signal et sa dérivée. La figure II.8 représente les instants  $t_{2k}$  et  $t_{2k+1}$  de rupture de  $p(t)$ , ces instants se trouvent à l'intersection des droites verticales rouges et vertes avec l'axe du temps. La moyenne des distances entre ces instants donnée par l'équation (II.8) permet de retrouver les valeurs estimées  $\hat{T}_1$  et  $\hat{T}_2$ , des paramètres  $T_1$  et  $T_2$ .

En moyenne  $\hat{T}_1 = 1.0488$  s et  $\hat{T}_2 = 1.1596$  s qui correspondent à une bonne estimation de  $T_1 = 1.1$  s et  $T_2 = 1.2$  s d'un véhicule de sapeurs-pompiers.

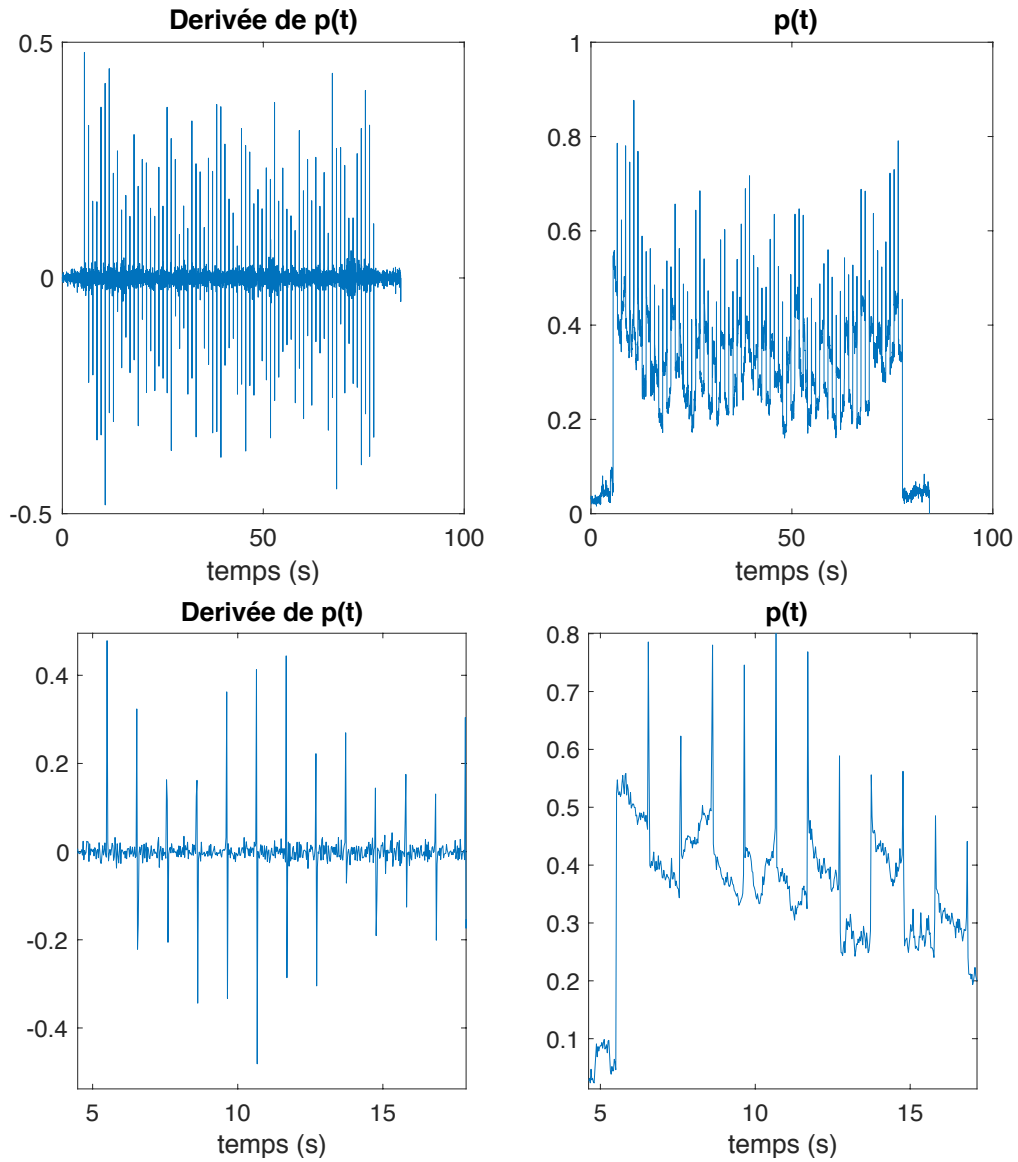


FIGURE II.7 – En haut : à droite  $p(t)$ , à gauche la dérivée de  $p(t)$ . En bas : zoom sur  $p(t)$  et sa dérivée, signal de sapeurs-pompiers

La figure II.9 représente les valeurs prises par l'indicateur  $r$  de l'algorithme 1 au cours du temps. Si la courbe prend la valeur 3, cela veut dire qu'on a détecté la présence de la signature fréquentielle et de la signature temporelle du signal. Si la courbe prend la valeur 2 cela veut dire qu'on détecte uniquement la signature fréquentielle du signal, sinon si la courbe prend la valeur 1, ceci veut dire qu'on a détecté uniquement la signature tempo-



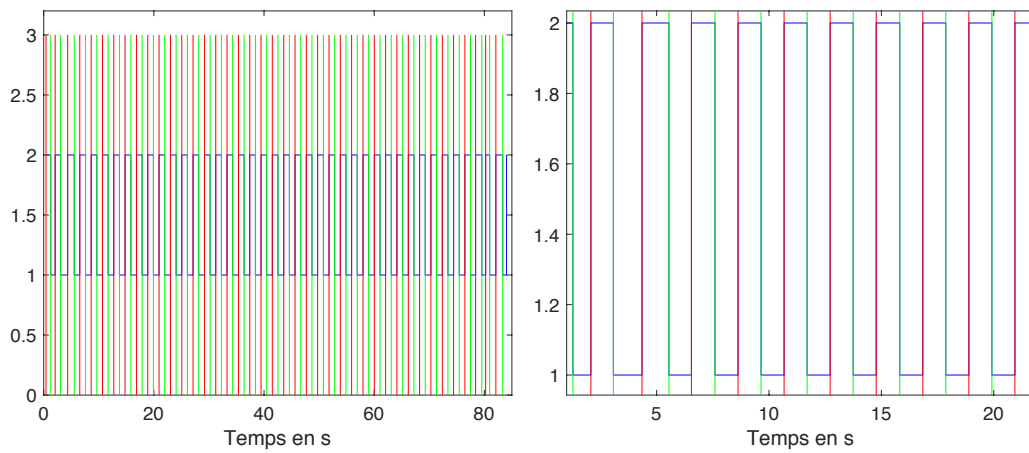


FIGURE II.8 – Estimation de  $T_1$  et  $T_2$  dans le temps

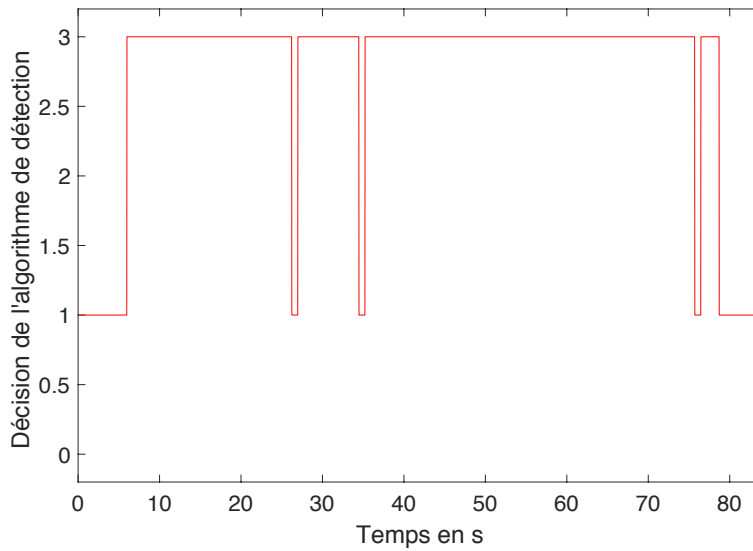


FIGURE II.9 – Détection de l'algorithme

relle de l'alarme.

Entre les instants 0 s et 8 s on détecte la présence de la signature temporelle sans détecter la signature fréquentielle. En observant le sonagramme du signal, on remarque qu'on ne dispose pas d'assez d'informations dans cet intervalle de temps, de même pour l'intervalle de temps compris entre les instants 76 s et 80 s.

Par contre entre les instants 8 s et 76 s on détecte presque partout la présence de la signature temporelle et de la signature fréquentielle, malgré le bruit qui s'ajoute au signal au niveau des basses fréquences et vers les fréquences 3900 Hz et 4000 Hz. La taille de  $\Delta$  est prise égale à la taille du signal ( $\Delta > T_1 + T_2$ ).

Comme les fréquences fondamentales des alarmes sont proches les unes des autres, alors le paramètre  $\epsilon$  introduit pour tenir compte de l'effet Doppler doit être de telle sorte qu'on ne confonde pas les fréquences fondamentales des alarmes. Dans un premier temps on a pris  $\epsilon$  tel que  $|\epsilon|=9$  Hz ; la différence entre les fréquences fondamentales des différents signaux d'alarme est supérieure à cette valeur. L'estimation des paramètres  $T_1$  et  $T_2$  étant difficile à cause de l'effet du bruit sur la dérivée, la valeur  $\epsilon'$  est prise grande dans un premier temps :  $|\epsilon'|=0.3$  s (relativement à  $T_1 = 1.1$  s et  $T_2 = 1.2$  s, cette erreur est grande).

Par la suite, les valeurs optimales de  $\epsilon'$  et  $\epsilon$  seront choisies à partir des courbes ROC en fonction des probabilités de fausse détection et détection.

## II.2.2 Signal bruité

Le signal est à présent bruité par le bruit de trafic et les klaxons des voitures. Pour cette scène audio, les conditions sont moins favorables pour effectuer la détection du signal d'alarme, comme le laisse présager le sonagramme de la figure II.10. Ce signal correspond au signal d'une alarme d'un véhicule de sapeurs-pompiers en déplacement.

La figure II.11 représente le tracé de  $\Phi(f, t)$  et sa dérivée pour  $t = 0$  s. Les pics sont décalés d'une petite valeur par rapport aux harmoniques des fréquences fondamentales ( $f_1, f_2$ ) d'une alarme de sapeurs-pompiers. Ce décalage est causé par le déplacement du véhicule produisant ainsi un déplacement des fréquences (effet Doppler [8]). La grande majorité des travaux sur la détection des signaux d'alarme des voitures prioritaires ne prennent pas cet effet en considération.

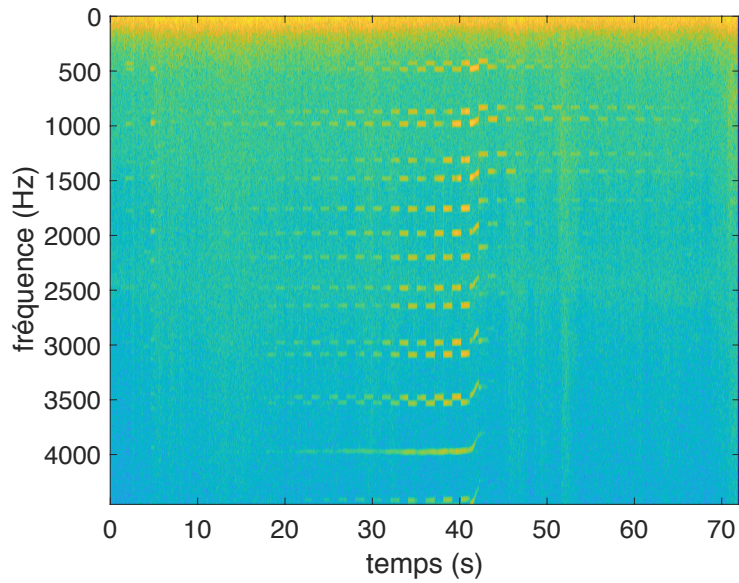


FIGURE II.10 – Sonagramme du signal enregistré, signal très bruité

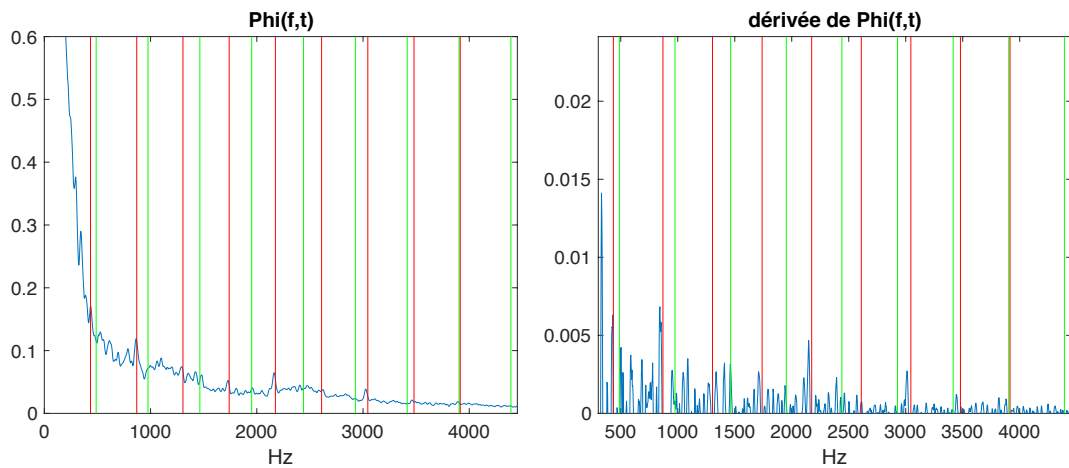


FIGURE II.11 –  $\Phi(f, t)$  et sa dérivée pour  $t = 0$  s et  $\Delta$  égal à la taille du signal

La tolérance introduite aux points 7 et 8 de l'algorithme permet de trouver les maxima au niveau des harmoniques de l'alarme même si les fréquences de celui-ci ont été décalées par l'effet Doppler.

La figure II.12 représente la fonction  $p(t)$  du signal bruité et sa dérivée. La figure II.13 représente les instants  $t_{2k}$  et  $t_{2k+1}$  de discontinuité de  $p(t)$  (droites verticales rouges et vertes). La moyenne de la distance entre ces instants donne les valeurs estimées de  $T_1$  et

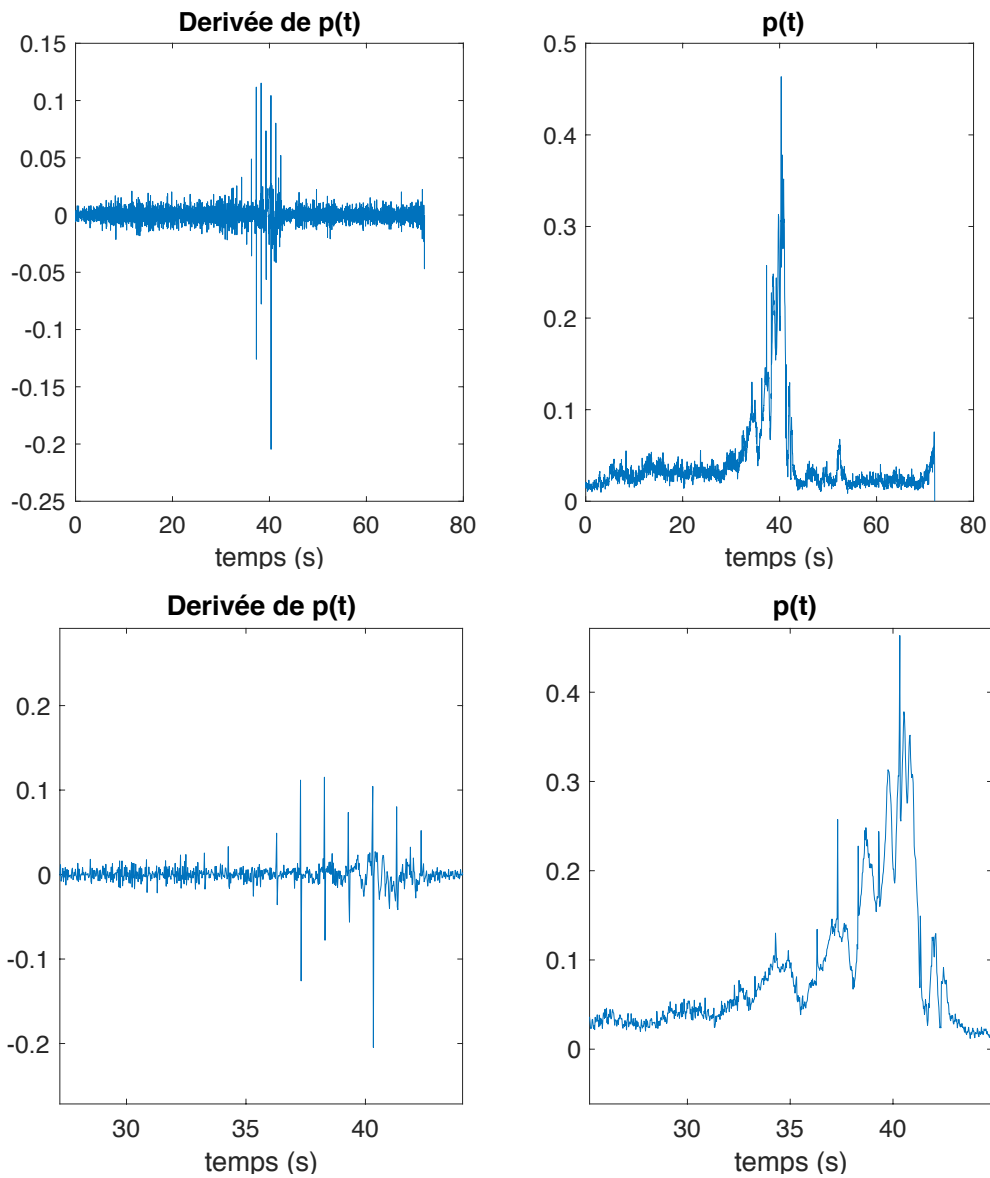


FIGURE II.12 –  $p(t)$  d'un signal de sapeurs-pompier enregistré, signal très bruité

$T_2$ , données par l'équation (II.8).

Dans cette situation, l'estimation des paramètres  $T_1$  et  $T_2$  n'est pas bonne, ceci provient du fait que le signal est très bruité. En moyenne  $\hat{T}_1 = 1.44$  s et  $\hat{T}_2 = 1.29$  s ( $T_1 = 1.1$  s,  $T_2 = 1.2$  s).

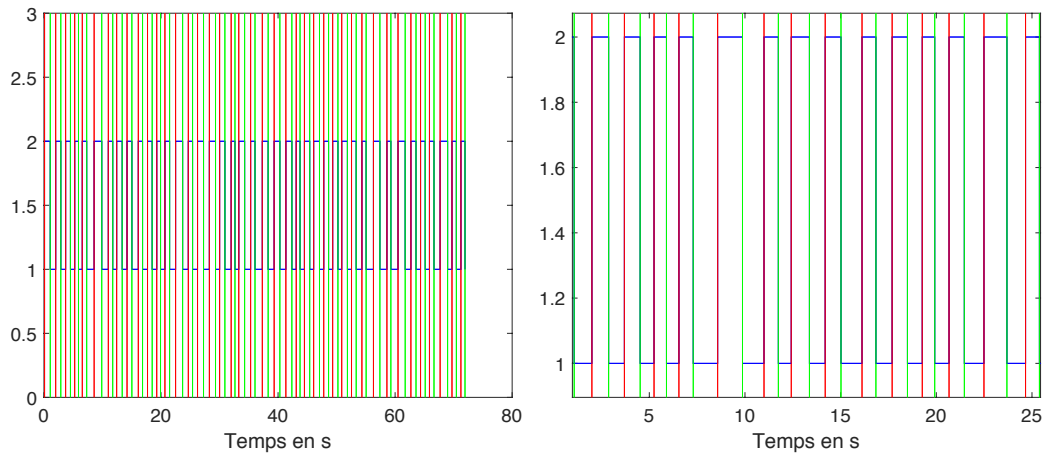


FIGURE II.13 – Estimation de  $T_1$  et  $T_2$  dans le temps

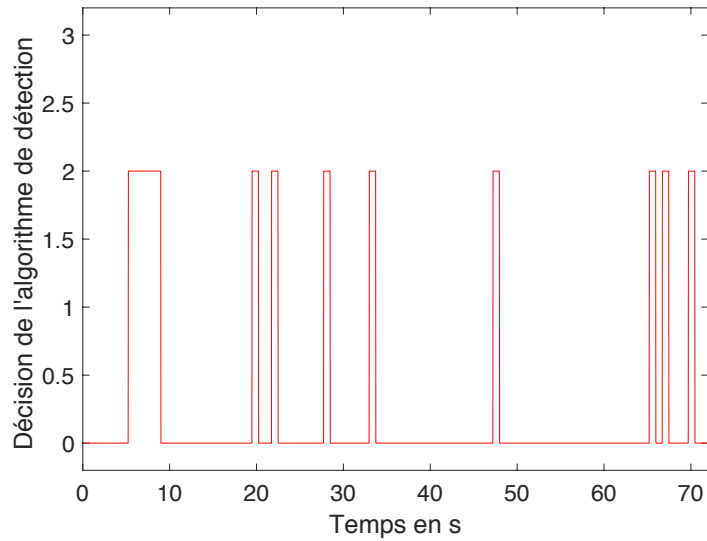


FIGURE II.14 – Détection de l'algorithme

La figure II.14 représente la décision de la détection. La courbe alterne entre 0 et 2, ce qui indique la détection occasionnelle de la signature fréquentielle du signal d'alarme.

La signature temporelle de l'alarme n'a pas été détectée, du fait d'une mauvaise estimation des paramètres  $T_1$  et  $T_2$ .

Dans la partie qui suit nous étudions les performances de l'algorithme proposé à l'aide

des courbes COR afin de déterminer les valeurs optimales des paramètres  $\epsilon$ ,  $\epsilon'$  et  $\mu$  permettant d'effectuer la détection.

## II.3 étude de performance (Courbe COR)

Dans cette partie nous étudions les performances de la méthode proposée. Plusieurs expériences ont été effectuées sur des signaux d'alarme bruités par un bruit de trafic enregistré dans l'une des rues de Paris. L'amplitude de ce bruit a été modifiée afin d'avoir des rapports signal à bruit compris entre  $-1$  dB et  $15$  dB. Les probabilités de fausse détection correspondent aux probabilités de détection dans le cas d'absence du signal, où uniquement le bruit de transmission est présent. Les probabilités de détection correspondent aux probabilités de détection en présence du signal.

Dans un premier temps, on considère une détection fondée uniquement sur la signature fréquentielle. Les courbes COR de la figure II.15 ont été calculées pour différentes valeurs de  $\mu$  (voir algorithme 1) et sont paramétrées par  $\epsilon$  (les valeurs testées  $\epsilon$  sont telles que :  $\frac{\epsilon}{f} \in \{0.05; 0.08; 0.1; 0.2; 0.25\}$ , pour chaque fréquence fondamentale  $f$ ).

Pour des rapports signal à bruit allant de  $5$  dB à  $15$  dB, l'algorithme de détection réalise de bonnes performances pour des valeurs de  $\mu$  égales à  $2$ ,  $3$  ou  $4$ . Par contre pour un rapport signal à bruit égal à  $-1$  dB, des valeurs de  $\mu$  supérieures ou égale à  $4$  sont préférables.

On considère une détection fondée sur la signature fréquentielle et la signature temporelle de l'alarme cherchée. Pour cela, on a choisi à partir de la figure II.15 pour chaque rapport signal à bruit le  $\mu$  optimal maximisant la probabilité de détection tout en minimisant la probabilité de fausse détection. Pour ces valeurs de  $\mu$ , des probabilités de

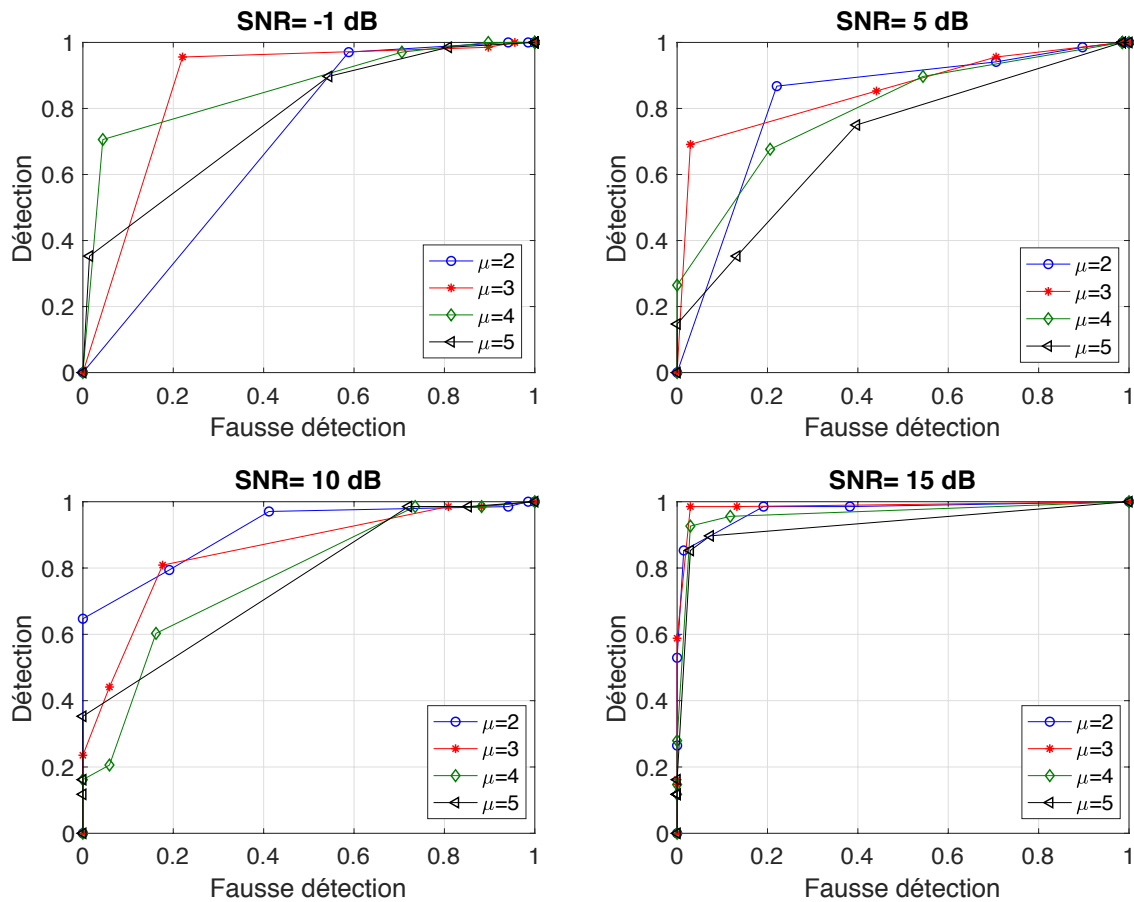


FIGURE II.15 – Probabilité de détection vs probabilité de fausse détection

détection et des probabilités de fausse détection ont été calculées pour chaque couple  $(\epsilon, \epsilon')$ , pour  $\epsilon$  fixé et  $\epsilon'$  variable. Les résultats sont représentés sur la figure II.16.

La détection de la signature temporelle est fondée sur l'estimation des paramètres  $T_2$  et  $T_1$ . Cette estimation est fondée sur la recherche des maxima de la dérivée de  $p(t)$ , or on a vu à la section II.2 que si le niveau du bruit est élevé l'estimation de ces paramètres devient très difficile à cause du bruit qui s'ajoute au signal, voir les figures II.13 et II.12. En effet, pour des rapports signal à bruit inférieurs à 15 dB, l'introduction de la contrainte de détection de la signature temporelle en plus de la signature fréquentielle dégrade les performances du détecteur (dans ce cas, une alarme est détectée uniquement quand les deux signature sont présentes, c'est à dire le paramètre de décision  $r = 3$ ).

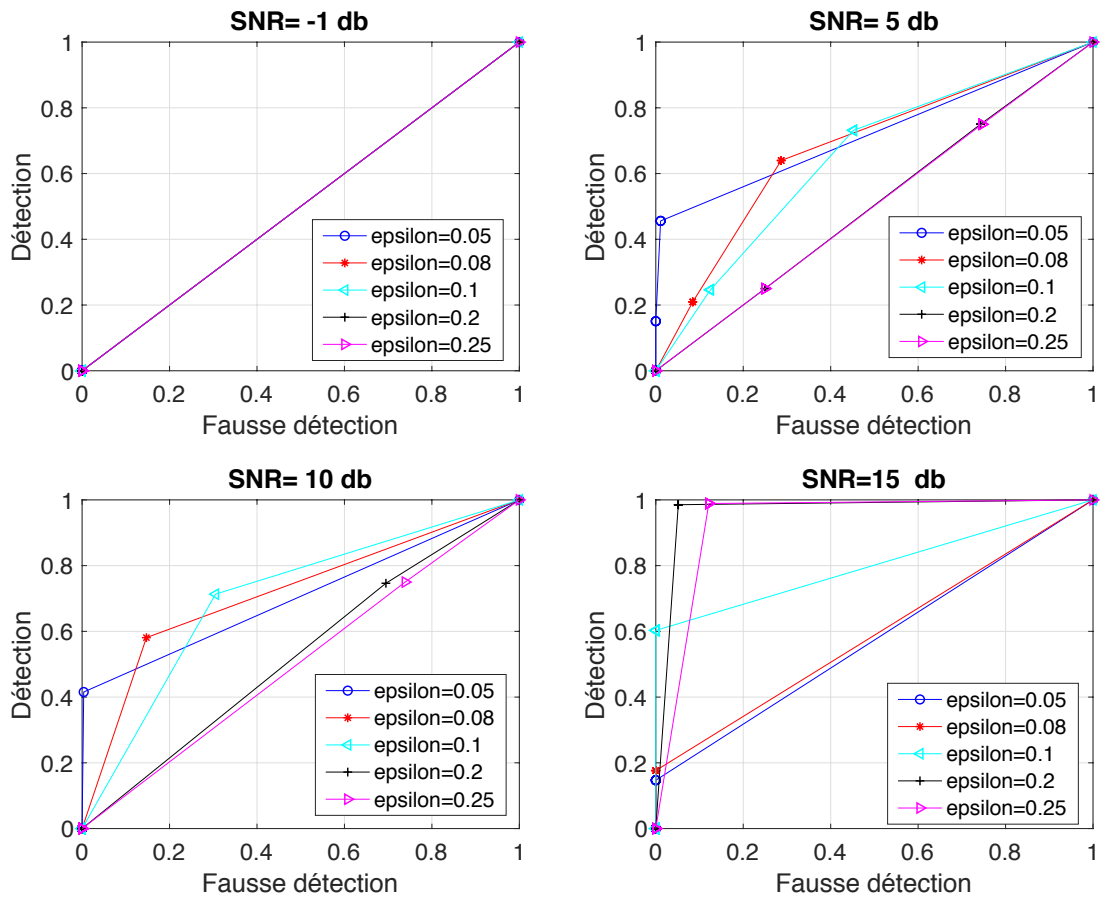


FIGURE II.16 – Probabilité de détection vs probabilité de fausse détection,  $\epsilon'=[0.1 \text{ s}; 0.2 \text{ s}; 0.3 \text{ s}; 0.35 \text{ s}; 0.4 \text{ s}; 0.45 \text{ s}]$ , epsilon ici est une erreur relative  $\epsilon=\frac{|\epsilon|}{f_1}$  s'il s'agit de l'estimation de  $f_1$  et  $\epsilon=\frac{|\epsilon|}{f_2}$  s'il s'agit de l'estimation de  $f_2$

Des expériences ont été effectuées dans le but d'étudier les performances de l'algorithme proposé dans le cas de la présence de deux signaux d'alarme en même temps. On cherche à détecter la présence d'un signal d'alarme d'une voiture de sapeurs pompiers en présence d'une alarme d'une voiture d'ambulance. La détection de l'alarme est fondée uniquement sur la détection de la signature fréquentielle de l'alarme. L'amplitude de l'alarme perturbatrice a été modifiée afin d'avoir plusieurs valeurs du rapport signal à bruit, comprises entre  $-1 \text{ dB}$  et  $15 \text{ dB}$ .



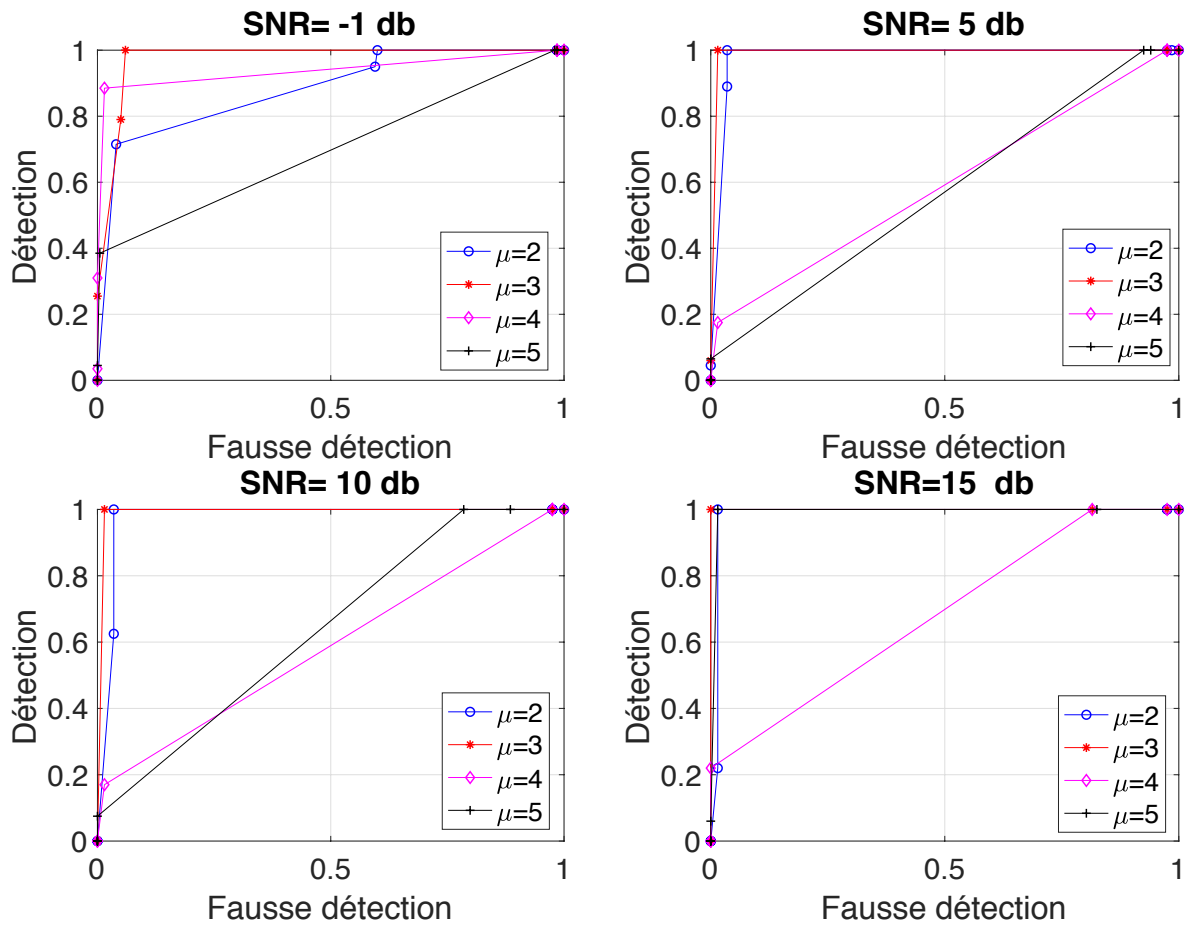


FIGURE II.17 – Probabilité de détection vs probabilité de fausse détection : détection d'un signal de sapeurs pompier en présence d'un autre signal d'alarme.

Les courbes COR de la figure II.17 montrent que l'algorithme assure une meilleure détection pour des valeurs de  $\mu$  égales à 2 et 3 pour des rapports signal à bruit allant de 5 dB à 10 dB, et  $\mu=2,3,5$  pour un rapport signal à bruit de 15 dB. Par contre pour un rapport signal à bruit de -1 dB,  $\mu$  doit être inférieur ou égal à 4 pour avoir de meilleures performances de l'algorithme.

## II.4 Conclusion

Nous avons proposé un algorithme de détection des signaux d'alarmes fondé sur la représentation temps-fréquence de ces signaux. Les signaux d'alarmes sont caractérisés par les paramètres  $T_1$ ,  $T_2$ ,  $f_1$  et  $f_2$ . Or ces paramètres diffèrent d'une alarme à l'autre avec de petits écarts. Avec cet algorithme on obtient des résultats satisfaisants dans le cadre de la détection de la présence d'un signal d'alarme par contre quand il s'agit d'identifier le signal détecté, cet algorithme ne sera pas capable d'identifier ce signal.

En effet, pour la détection de la signature fréquentielle de l'alarme on tolère une erreur  $\epsilon$  dans l'identification des maxima qui se trouvent au niveau des harmoniques des fréquences fondamentales  $f_1$  et  $f_2$ , qui permettra de tenir compte de l'effet Doppler. Comme les fréquences fondamentales des signaux d'alarme sont proches les unes des autres, cette erreur peut faire confondre les harmoniques d'une alarme avec celles d'une autre alarme. De même, pour l'estimation des paramètres  $T_1$  et  $T_2$ , l'erreur  $\epsilon'$  devient grande quand le niveau du bruit devient très élevé, or comme l'identification de ces signaux est fondée sur les paramètres  $T_1$ ,  $T_2$ ,  $f_1$  et  $f_2$ , alors les paramètres des alarmes peuvent être confondus et par conséquent la probabilité de fausse identification sera grande.



# Chapitre III

## Quantification correctrice d'erreurs

Nous proposons dans ce chapitre une méthode de représentation des signaux audio robuste au bruit. Cette méthode est une forme de quantification vectorielle. Elle repose sur le codage correcteur d'erreurs.

Les méthodes de quantification décrites au chapitre I visent à représenter une séquence du signal au moyen d'un dictionnaire de taille limitée construit de manière à minimiser l'erreur de quantification. Ici, si la minimisation de l'erreur de quantification reste un objectif lors de la quantification, la construction du dictionnaire vise prioritairement à maximiser la distance entre ses éléments de telle sorte que ceux-ci soient robustes au bruit.

Le chapitre V décrit une application de cette approche : on utilise la quantification pour donner aux signaux d'alarme une forme spécifique permettant de les détecter et de les identifier facilement en présence d'autres signaux ou en présence du bruit.

### III.1 Quantification fondée sur le codage correcteur d'erreurs

Chaque trame de longueur  $n$  du signal est représentée comme une combinaison linéaire de  $L$  vecteurs binaires de longueur  $n$ . La quantification consiste à transformer chaque

trame en une trame de même longueur formée par une combinaison linéaire de  $L$  mots de code. Deux possibilités de codage sont proposées, un codage dans le domaine temporel et un codage dans le domaine fréquentiel.

### III.1.1 Principe

Soit  $s = [s_1, s_2, \dots, s_n]$  une trame de longueur  $n$  d'échantillons temporels ou fréquentiels d'un signal  $x$ ,  $s \in A^n$  avec  $A = [-2^L + 1, 2^L - 1] \subset \mathbb{Z}$  et  $L + 1$  le nombre de bits sur lesquels seront représentés les coefficients du signal  $x$ .

	$s_1$	$s_2$	.	.	.	.	$s_n$	$s_{(n+1)} \dots$
$2^0$	$X_{01}$	$X_{02}$	.	.	.	.	$X_{0n}$	$X_0$
$2^1$	$X_{11}$	$X_{12}$	.	.	.	.	$X_{1n}$	$X_1$
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
$2^{L-1}$	.	.	.	.	.	.	.	$X_{L-1}$
signe	$X_{L1}$	$X_{L2}$	.	.	.	.	$X_{Ln}$	$X_L$

TABLE III.1 – Représentation binaire des signaux

Chaque trame du signal sera décomposée en combinaison linéaire de vecteurs binaires. Les coefficients de la combinaison linéaire sont des puissances de deux (correspondant au poids de bit). Chaque vecteur binaire de cette combinaison sera projeté dans un espace de mots de code d'un codeur correcteur d'erreurs de longueur  $n$ . Les codeurs utilisés dans ce chapitre sont des codes en bloc, en particulier les codes BCH [6].

La trame  $s$  s'exprime comme suit :

$$s = M_s X^t, \tag{III.1}$$

où

$$X = \sum_{i=0}^{L-1} 2^i X_i, \in A_+^n$$

avec  $X_i = [X_{i1}, X_{i2}, \dots, X_{in}] \in \mathbb{F}_2^n$ ,  $0 \leq i \leq L$  et  $M_s$  une matrice diagonale dont la diagonale

est le vecteur  $[-1^{X_{L1}}, -1^{X_{L2}}, \dots, -1^{X_{Ln}}]$ . Le vecteur  $X_L$  est formé par les bits de signe ; si la composante d'indice  $j$  de  $X_L$  est égale à zéro, alors le coefficient  $s_j$  de la trame  $s$  est positif, sinon le coefficient  $s_j$  est négatif,  $1 \leq j \leq n$ . La représentation des signaux est résumée par la table III.1.

La trame  $s$  est le produit d'une matrice diagonale  $M_s$  dont la diagonale correspond au vecteur de signe  $X_L$  et d'une trame  $X$ . La quantification proposée dans ce chapitre laissera invariants les bits de signes, de sorte que quantifier la trame  $s$  revient à quantifier la trame  $X$ .

Après quantification  $s$  devient  $\hat{s}$  définie par :

$$\hat{s} = M_s \hat{X}^t, \quad (\text{III.2})$$

où  $\hat{X}$  est la trame  $X$  quantifiée donnée par :

$$\hat{X} = \sum_{i=0}^{L-1} 2^i C_i,$$

où chaque  $C_i$ ,  $0 \leq i \leq L - 1$  est un mot de code d'un codeur en bloc  $\zeta_i(n, k_i, t_i)$  de longueur  $n$ , de dimension  $k_i$  et de capacité de correction  $t_i$ .

Le codage classique consiste à transformer  $k_i$  bits d'information en  $n$  bits codés. Ici, pour tout  $i$  de 0 à  $L - 1$ , la quantification consiste à transformer les  $n$  bits d'information de  $X_i$  en un mot de code  $C_i$  de  $n$  bits.

Le processus du codage ici est similaire à un processus de décodage correcteur d'erreurs, où chaque vecteur binaire  $X_i$  sera remplacé par le mot de code  $C_i$  le plus proche en termes de distance de Hamming.

### III.1.2 Application du principe de la modulation codée en bloc à la quantification

La modulation codée en bloc (MCB) a été introduite en 1997 par Imai et Hirikawa [21], elle permet d'optimiser conjointement la modulation et le codage du canal. Le principe de la modulation codée en bloc est indiqué à la figure III.1. Considérons une modulation  $M$ -aire avec  $M = 2^{L+1}$ . Pour transmettre une séquence binaire de longueur  $k_0 + k_1 + \dots + k_i + \dots + k_L$  bits, on code chaque bloc de  $k_i$  bits par un bloc de longueur  $n$  bits,  $n \geq k_i$  et  $0 \leq i \leq L$ .

La matrice binaire  $(L + 1) \times n$  résultante du codage est par la suite transmise comme  $n$  symboles  $M$ -aires. Le codage exploite le fait que les bits de poids fort sont moins vulnérables au bruit du canal de transmission. Si le poids de bit est élevé, le taux d'erreurs binaire sans codage diminue. Ainsi plusieurs codeurs de capacités de correction décroissantes sont utilisés. Les codeurs de capacités de correction élevées codent les bits de poids faibles tandis que les bits de poids élevés sont codés par des codeurs de faibles capacités de correction. Ainsi les  $k_i$  sont ordonnés de la manière suivante,  $k_0 \leq k_1 \leq \dots \leq k_L$ .

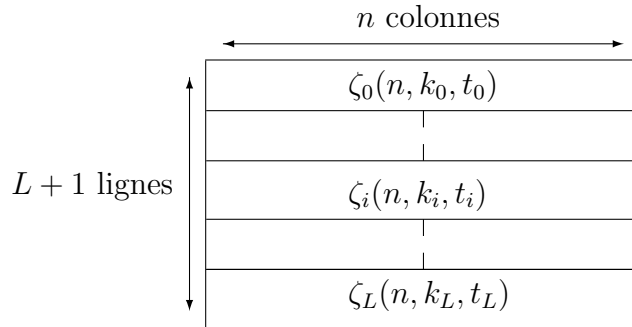


FIGURE III.1 – Principe de la modulation codée en bloc.

Soit  $(\zeta_i)_{0 \leq i \leq L}$  une famille de codeurs en bloc de longueur  $n$ , de dimensions  $k_i$  et de capacités de correction  $t_i$ ,  $0 \leq i \leq L$ . Alors une matrice de  $L + 1$  codes de cette famille peut être utilisée pour coder des trames de  $n$  coefficients temporels ou fréquentiels chacun, représenté sur  $L + 1$  bits. Nous utilisons le principe de la modulation codée en bloc décrit

ci-dessus pour effectuer la quantification des signaux, où le codeur  $\zeta_L$  laisse invariants les bits de poids  $L$  (les bits de poids le plus fort).

### III.1.3 Comparaison avec la quantification vectorielle

La quantification proposée est une forme de quantification vectorielle : au lieu de quantifier chaque échantillon séparément, on code des vecteurs d'échantillons. Notre méthode présente cependant les différences suivantes avec l'approche classique de la quantification vectorielle :

- Au lieu d'être une application de  $\mathbb{R}^n$  dans un sous-ensemble  $\mathbb{D}$  de  $\mathbb{R}^n$ , la quantification proposée consiste en  $L$  applications de  $\mathbb{F}_2^n$  vers des sous-espaces vectoriels  $\mathbb{D}_i$  de  $\mathbb{F}_2^n$  de dimension  $k_i$ , indépendantes les unes des autres.
- Les dictionnaires sont indépendants des données. Ils sont construits selon un critère différent : minimisation de l'erreur quadratique moyenne de quantification pour la quantification vectorielle classique, maximisation de la distance de Hamming minimale entre deux mots de code pour notre méthode. De même, alors que la quantification vectorielle classique associe un mot à un vecteur selon un critère de distance euclidienne, c'est la distance de Hamming qui est utilisée ici.

On pourrait se rapprocher de la quantification vectorielle classique par une optimisation conjointe des  $L$  quantifieurs, *i.e.* en considérant une application de  $\mathbb{F}_2^n \times \mathbb{F}_2^n \times \dots \times \mathbb{F}_2^n$  ( $L$  fois) vers  $\mathbb{D}_0 \times \mathbb{D}_1 \times \dots \times \mathbb{D}_{L-1}$ . La taille de l'espace de recherche rendrait cependant la complexité rédhibitoire.

## III.2 Choix des codeurs

### III.2.1 Probabilité d'erreur binaire avant décodage

On a vu à la section précédente que chaque niveau de bits est codé par un codeur de longueur  $n$  et de dimension  $k_i$ . Comme les bits de poids fort sont moins vulnérables aux erreurs, alors les codeurs seront choisis avec des capacités de correction décroissantes



dans l'ordre croissant des poids de bit en fonction de la probabilité d'erreurs binaires introduites par poids de bit.

Afin de choisir les codeurs par poids de bit, nous calculons pour chaque poids de bit  $i$ , une probabilité  $P_e^i$  d'erreur introduite par le bruit du canal de transmission en fonction du rapport signal à bruit du canal.

**Proposition III.2.1.** *On considère un signal numérique  $s$ , reçu à travers un canal à bruit additif. Alors la probabilité d'erreur binaire  $P_e^i$  introduite par le canal sur le bit  $i$  est donnée par :*

$$\begin{aligned}
\text{Pour } i = 0, P_e^i &= \frac{1}{2}P(|b_j| \geq 1) \\
\text{Pour } i = 1, P_e^i &= \frac{1}{2}P(|b_j| \geq 2) + \frac{1}{4}P(|s_j| \geq 1)P(|b_j| \geq 1)(1 - P(|b_j| \geq 2)) \\
\text{Pour } i \geq 2, P_e^i &= \frac{1}{2}P(|b_j| \geq 2^i) + \frac{1}{4}P(|s_j| \geq 2^{i-1})P(|b_j| \geq 2^{i-1})(1 - P(|b_j| \geq 2^i)) \\
&\quad + \frac{1}{4} \left[ \sum_{k=0}^{i-2} P(|s_j| \geq 2^k)P(|b_j| \geq 2^k) \prod_{l=k+1}^{i-1} A_{lj} \right] (1 - P(|b_j| \geq 2^i)),
\end{aligned} \tag{III.3}$$

où  $A_{lj}$  est donné par l'expression suivante :

$$A_{lj} = \frac{1}{2} [(P(|s_j| \geq 2^l) + P(|b_j| \geq 2^l) - P(|s_j| \geq 2^l)P(|b_j| \geq 2^l))].$$

**Preuve :**

Soit  $s_j$  un échantillon de signal et  $b_j$  un échantillon de bruit. Soit  $[s_{0j}, s_{1j}, \dots, s_{ij}, \dots, s_{Lj}]$  la représentation binaire de  $s_j$  et  $[b_{0j}, b_{1j}, \dots, b_{ij}, \dots, b_{Lj}]$  la représentation binaire de  $b_j$ . Soit  $r_{ij}$  la retenue (0 ou 1) issue de l'addition de  $s_{(i-1)j}$  et  $b_{(i-1)j}$  qui va s'appliquer à l'addition de  $s_{ij}$  et  $b_{ij}$  :

$$r_{ij} = s_{(i-1)j} + b_{(i-1)j} + r_{(i-1)j} + 1$$

L'addition du bruit transforme chaque bit  $s_{ij}$  en  $s_{ij} + b_{ij} + r_{ij}$  il y a donc une erreur sur le bit de poids  $i$  dès lors que l'effet cumulé du bruit binaire et de la retenue est non nul :  $b_{ij} + r_{ij} = 1$ . La probabilité d'erreur binaire sur le bit  $i$  est donc :

$$\begin{aligned}
 P_e^i &= P(b_{ij} = 1 \text{ et } r_{ij} = 0) + P(b_{ij} = 0 \text{ et } r_{ij} = 1) \\
 &= P(b_{ij} = 1)(1 - P(r_{ij} = 1)) + P(r_{ij} = 1)(1 - P(b_{ij} = 1)) \\
 &= P(b_{ij} = 1) + P(r_{ij} = 1)(1 - 2P(b_{ij} = 1)),
 \end{aligned} \tag{III.4}$$

avec  $P(r_{0j} = 1) = 0$  et  $P(r_{1j} = 1) = P(s_{0j} = 1)P(b_{0j} = 1)$ .

Pour  $i \geq 1$  on a :

$$P(s_{ij} = 1) = P(s_{ij} = 1 \mid |s_j| \geq 2^i)P(|s_j| \geq 2^i) \tag{III.5}$$

et

$$P(b_{ij} = 1) = P(b_{ij} = 1 \mid |b_j| \geq 2^i)P(|b_j| \geq 2^i) \tag{III.6}$$

Comme  $s_{ij}$  et  $b_{ij}$  prennent uniquement les valeurs 1 ou 0, alors du fait de la représentation binaire utilisée les bits 0 et 1 sont équiprobables, ainsi  $\forall s_j$  on a :

$$P(s_{ij} = 1 \mid |s_j| \geq 2^i) = P(s_{ij} = 0 \mid |s_j| \leq 2^i) = \frac{1}{2}.$$

et

$$P(b_{ij} = 1 \mid |b_j| \geq 2^i) = P(b_{ij} = 0 \mid |b_j| \leq 2^i) = \frac{1}{2}.$$

Ainsi, les probabilités  $P(s_{ij} = 1)$  et  $P(b_{ij} = 1)$  données respectivement par les équations (III.5) et (III.6) deviennent :

$$P(s_{ij} = 1) = \frac{1}{2}P(|s_j| \geq 2^i)$$

et

$$P(b_{ij} = 1) = \frac{1}{2}P(|b_j| \geq 2^i)$$

Ainsi on obtient :

$$\text{Pour } i = 0, P_e^0 = P(b_{0j} = 1) = \frac{1}{2}P(|b_j| \geq 1).$$

Pour  $i = 1$ ,

$$\begin{aligned} P_e^1 &= P(b_{1j} = 1) + P(s_{0j} = 1)P(b_{0j} = 1)(1 - 2P(b_{1j} = 1)) \\ &= \frac{1}{2}P(|b_j| \geq 2) + \frac{1}{4}P(|s_j| \geq 1)P(|b_j| \geq 1)(1 - P(|b_j| \geq 2)) \end{aligned} \quad (\text{III.7})$$

Pour  $i \geq 2$ ,  $P(r_{ij} = 1)$  est donnée par :

$$\begin{aligned} P(r_{ij} = 1) &= P(s_{(i-1)j} = 1, b_{(i-1)j} = 1 \text{ et } r_{(i-1)j} = 0) + P(r_{(i-1)j} = 1, b_{(i-1)j} = 1 \text{ et } s_{(i-1)j} = 0) \\ &\quad + P(s_{(i-1)j} = 1, r_{(i-1)j} = 1 \text{ et } b_{(i-1)j} = 0) \end{aligned}$$

$$\begin{aligned} P(r_{ij} = 1) &= P(r_{(i-1)j} = 1) [P(s_{(i-1)j} = 1) + P(b_{(i-1)j} = 1) - 2P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1)] \\ &\quad + P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1). \end{aligned}$$

(III.8)

Posons :

$$A_{(i-1)j} = P(s_{(i-1)j} = 1) + P(b_{(i-1)j} = 1) - 2P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1).$$

Alors,

$$P(r_{ij} = 1) = P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1) + A_{(i-1)j}P(r_{(i-1)j} = 1). \quad (\text{III.9})$$

En exploitant la relation de récurrence entre les  $r_{ij}$  l'équation (III.9) peut s'écrire sous la forme suivante en fonction de  $P(s_{(i-1)j} = 1)$  et  $P(b_{(i-1)j} = 1)$  uniquement :

$$P(r_{ij} = 1) = P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1) + \sum_{k=0}^{i-2} P(s_{kj} = 1)P(b_{kj} = 1) \prod_{l=k+1}^{i-1} A_{lj}. \quad (\text{III.10})$$

En remplaçant  $P(r_{ij} = 1)$ ,  $P(s_{ij} = 1)$ ,  $P(b_{ij} = 1)$  et  $A_{lj}$  par leur expression dans l'équation (III.4) on obtient :

$$P_e^i = P(b_{ij} = 1) + P(s_{(i-1)j} = 1)P(b_{(i-1)j} = 1) (1 - 2P(b_{ij} = 1)) + \left[ \sum_{k=0}^{i-2} P(s_{kj} = 1)P(b_{kj} = 1) \prod_{l=k+1}^{i-1} A_{lj} \right] (1 - 2P(b_{ij} = 1)) \quad (\text{III.11})$$

Ainsi, l'équation (III.11) devient :

$$P_e^i = \frac{1}{2}P(|b_j| \geq 2^i) + \frac{1}{4}P(|s_j| \geq 2^{i-1})P(|b_j| \geq 2^{i-1}) (1 - P(|b_j| \geq 2^i)) + \frac{1}{4} \left[ \sum_{k=0}^{i-2} P(|s_j| \geq 2^k)P(|b_j| \geq 2^k) \prod_{l=k+1}^{i-1} A_{lj} \right] (1 - P(|b_j| \geq 2^i)). \quad (\text{III.12})$$

$A_{lj}$  est donné par l'expression suivante :

$$A_{lj} = \frac{1}{2} [(P(|s_j| \geq 2^l) + P(|b_j| \geq 2^l) - P(|s_j| \geq 2^l)P(|b_j| \geq 2^l))].$$

**Fin preuve**

**Proposition III.2.2.** *On considère un signal numérique  $s$ , reçu à travers un canal additif à bruit blanc gaussien<sup>1</sup> de variance  $\sigma^2$ . Si le signal  $s$  suit une loi uniforme sur l'intervalle  $[-2^L + 1, 2^L - 1]$ , alors la probabilité d'erreur binaire  $P_e^i$  introduite par le canal sur le bit  $i$  est donnée par :*

---

1. On suppose que le nombre de bits  $L + 1$  pour représenter les signaux est suffisamment élevé de sorte que le bruit numérique puisse être décrit comme sa version réelle, par une variable gaussienne de variance  $\sigma^2$ .

$$\begin{aligned}
\text{Pour } i = 0, P_e^i &= \text{erfc}\left(\frac{1}{\sigma}\right) \\
\text{Pour } i = 1, P_e^i &= \text{erfc}\left(\frac{2}{\sigma}\right) + \frac{(2^L - 2)\text{erfc}\left(\frac{1}{\sigma}\right)}{2^L - 1} (1 - 2\text{erfc}\left(\frac{2}{\sigma}\right)) \\
\text{Pour } i \geq 2, P_e^i &= \text{erfc}\left(\frac{2^i}{\sigma}\right) + \frac{(2^L - 1 - 2^{i-1})\text{erfc}\left(\frac{2^{i-1}}{\sigma}\right)(1 - \text{erfc}\left(\frac{2^i}{\sigma}\right))}{(2^L - 1)} \\
&\quad + \left[ \sum_{k=0}^{i-2} (2^L - 1 - 2^k) \text{erfc}\left(\frac{2^k}{\sigma}\right) \prod_{l=k+1}^{i-1} \left( \frac{2^L - 1 - 2^l}{(2^L - 1)} + \text{erfc}\left(\frac{2^l}{\sigma}\right) - \frac{(2^L - 1 - 2^l)\text{erfc}\left(\frac{2^l}{\sigma}\right)}{(2^L - 1)} \right) \right] \\
&\quad \times \frac{\left(1 - \text{erfc}\left(\frac{2^i}{\sigma}\right)\right)}{(2^L - 1)}.
\end{aligned} \tag{III.13}$$

**Preuve :**

Supposons que  $b_j$  soit un échantillon d'un bruit blanc gaussien de variance  $\sigma^2$ , alors  $P(|b_j| \geq 2^i)$  est donnée par :

$$P(|b_j| \geq 2^i) = 2P(b_j \geq 2^i) = 2\text{erfc}\left(\frac{2^i}{\sigma}\right) \tag{III.14}$$

Et, supposons que  $s_j$  suit une loi de distribution uniforme entre  $[-2^L + 1, 2^L - 1]$ , alors  $P(|s_j| \geq 2^i)$  est donnée par :

$$P(|s_j| \geq 2^i) = 2P(s_j \geq 2^i) = \frac{2(2^L - 1 - 2^i)}{2^{L+1} - 2}. \tag{III.15}$$

En remplaçant  $P(|s_j| \geq 2^i)$  et  $P(|b_j| \geq 2^i)$  dans l'équation (III.12) par leur valeur on obtient :

$$\text{Pour } i = 0, P_e^0 = P(|b_j| \geq 1) = \text{erfc}\left(\frac{1}{\sigma}\right).$$

$$\text{Pour } i = 1, P_e^1 = \text{erfc}\left(\frac{2}{\sigma}\right) + \frac{(2^L - 2)\text{erfc}\left(\frac{1}{\sigma}\right)}{2^L - 1} (1 - 2\text{erfc}\left(\frac{2}{\sigma}\right)).$$

Pour  $i \geq 2$ ,

$$P_e^i = \operatorname{erfc}\left(\frac{2^i}{\sigma}\right) + \left[ (2^L - 1 - 2^{i-1})\operatorname{erfc}\left(\frac{2^{i-1}}{\sigma}\right) + \sum_{k=0}^{i-2} (2^L - 1 - 2^k)\operatorname{erfc}\left(\frac{2^k}{\sigma}\right) \prod_{l=k+1}^{i-1} A_{lj} \right] \frac{\left(1 - \operatorname{erfc}\left(\frac{2^i}{\sigma}\right)\right)}{(2^L - 1)} \quad (\text{III.16})$$

avec,

$$\begin{aligned} A_{lj} &= P(s_{lj} = 1) + P(b_{lj} = 1) - 2P(s_{lj} = 1)P(b_{lj} = 1). \\ &= \frac{2^L - 1 - 2^l}{(2^L - 1)} + \operatorname{erfc}\left(\frac{2^l}{\sigma}\right) - \frac{(2^L - 1 - 2^l)\operatorname{erfc}\left(\frac{2^l}{\sigma}\right)}{(2^L - 1)} \end{aligned}$$

En remplaçant  $A_{lj}$  par sa valeur on obtient :

$$\begin{aligned} P_e^i &= \operatorname{erfc}\left(\frac{2^i}{\sigma}\right) + \frac{(2^L - 1 - 2^{i-1})\operatorname{erfc}\left(\frac{2^{i-1}}{\sigma}\right)(1 - \operatorname{erfc}\left(\frac{2^i}{\sigma}\right))}{(2^L - 1)} \\ &+ \left[ \sum_{k=0}^{i-2} (2^L - 1 - 2^k)\operatorname{erfc}\left(\frac{2^k}{\sigma}\right) \prod_{l=k+1}^{i-1} \left( \frac{2^L - 1 - 2^l}{(2^L - 1)} + \operatorname{erfc}\left(\frac{2^l}{\sigma}\right) - \frac{(2^L - 1 - 2^l)\operatorname{erfc}\left(\frac{2^l}{\sigma}\right)}{(2^L - 1)} \right) \right] \\ &\times \frac{\left(1 - \operatorname{erfc}\left(\frac{2^i}{\sigma}\right)\right)}{(2^L - 1)}. \end{aligned} \quad (\text{III.17})$$

### Fin preuve

La variance du bruit  $\sigma^2$  est une fonction du rapport signal à bruit et de la puissance  $P_s^2$  du signal, donnée par :

$$\sigma^2 = P_s^2 10^{-\frac{SNR_{dB}}{10}}, \quad (\text{III.18})$$

La figure III.2 montre comment varie la probabilité  $P_e^i$  d'erreur par poids de bit en fonction de  $i$  pour différents rapports signal à bruit. À partir de ces courbes (ou des formules correspondantes), nous allons pouvoir, pour chaque rapport signal à bruit et chaque poids de bit, définir le codeur adéquat en fonction de la probabilité d'erreur.

Comme indiqué par la figure III.2 la probabilité d'erreur sur chaque bit dépend du rapport signal à bruit du canal de transmission et du poids de bit. Eu égard au faible taux d'erreur (suivant le rapport signal à bruit), certains poids de bit n'ont pas besoin

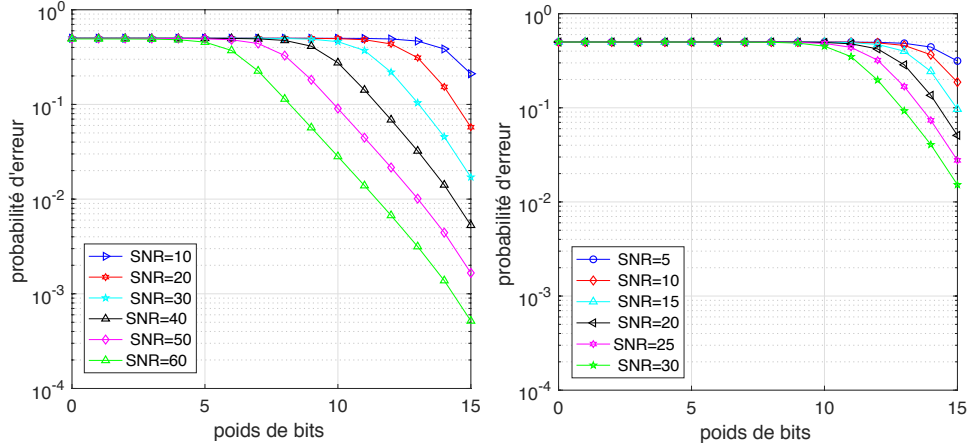


FIGURE III.2 – Probabilité d’erreur théorique  $P_e^i$  en fonction de  $i$ , pour plusieurs SNR, pour un signal à distribution uniforme bruité par un bruit blanc gaussien, représenté sur  $L + 1 = 16$  bits.

d’être codés tandis que d’autres nécessitent une grande capacité de correction. Pour ces derniers, si le pouvoir de correction du codeur utilisé n’est pas suffisant pour corriger les erreurs, le décodage a tendance à augmenter le taux d’erreur plutôt que de le diminuer : voir les courbes de la figure III.3. Cette figure représente le nombre moyen d’erreurs après décodage correcteur d’erreurs en fonction du nombre d’erreurs avant décodage, pour plusieurs codes BCH [6] de différentes capacités de correction. Pour chaque codeur de capacité de correction  $t$ , un nombre  $k$  d’erreurs a été ajouté à un mot de code de ce codeur. Ces erreurs ont été réparties aléatoirement sur ce mot de code ensuite le décodage est effectué sur le mot de code erroné,  $1 \leq k \leq n$ . Pour  $k \leq t$ , le nombre moyen d’erreurs après décodage est nul, pour  $k > t$  ce nombre est non nul, un nombre moyen d’erreurs après décodage a été ainsi calculé sur un nombre d’expériences de  $10^6$ .

### III.2.2 Probabilité d’erreur binaire après décodage

Pour effectuer le choix des codeurs par poids de bit nous nous sommes servis de la probabilité d’erreur après décodage correcteur d’erreurs. Cette probabilité est calculée pour chaque poids de bit  $i$  et pour chaque capacité de correction  $t_i$  en fonction des rapports signal à bruit. Elle sera notée  $P_{t_i}^{decod}$ .

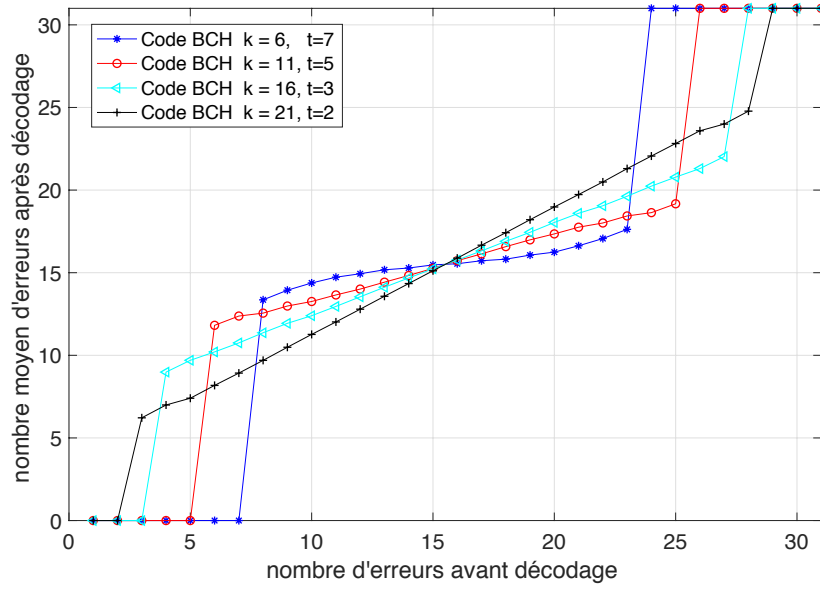


FIGURE III.3 – Nombre d’erreurs après décodage correcteur d’erreurs vs nombre d’erreurs avant décodage correcteur d’erreur, pour des codes BCH de longueur  $n = 31$ .

**Proposition III.2.3.** *La probabilité binaire d’erreur après décodage par poids de bit  $P_{t_i}^{decod}$  d’un signal numérique transmis à travers un canal bruité est une fonction de la probabilité binaire d’erreur avant décodage  $P_e^i$ , de la capacité de correction du codeur  $\zeta_i(n, k_i, t_i)$  et du nombre moyen d’erreurs après décodage :*

$$P_{t_i}^{decod} = \frac{1}{n} \sum_{j=t_i+1}^n N_i^{decod}(j, n) \binom{n}{j} (P_e^i)^j (1 - P_e^i)^{(n-j)}. \quad (III.19)$$

avec  $N_i^{decod}(j, n)$  le nombre d’erreurs moyen après décodage sur  $n$  bits de poids  $i$ , sachant qu’il y avait  $j$  erreurs sur ces  $n$  bits avant décodage.

**Preuve :**

Pour un bit de poids  $i$ , considérons les événements suivants  $A_i$  et  $B_{ij}$  donnés par :

$A_i = \{ \text{le bit de poids } i \text{ est erroné après décodage} \}$  et  $B_{ij} = \{ j \text{ bits de poids } i \text{ sont erronés sur } n \text{ bits avant décodage} \}$ .



Alors, la probabilité  $P_{t_i}^{decod}$  d'erreur après décodage par poids de bit  $i$  en fonction de la capacité de correction  $t_i$  est donnée par :

$$P_{t_i}^{decod} = P(A_i) = \sum_{j=0}^n P(A_i | B_{ij})P(B_{ij}).$$

Pour tout poids de bit  $i$ , la probabilité d'avoir  $j$  bits erronés sur  $n$  bits est donnée par :

$$P(B_{ij}) = \binom{n}{j} (P_e^i)^j (1 - P_e^i)^{(n-j)}. \quad (\text{III.20})$$

Et

$$P(A_i | B_{ij}) = \frac{N_i^{decod}(j, n)}{n},$$

On obtient

$$P_{t_i}^{decod} = \frac{1}{n} \sum_{j=0}^n N_i^{decod}(j, n)P(B_{ij}).$$

Pour  $j \leq t_i$ ,  $N_i^{decod}(j, n) = 0$ , ainsi  $P_{t_i}^{decod}$  sera donnée par :

$$P_{t_i}^{decod} = \frac{1}{n} \sum_{j=0}^{t_i} N_i^{decod}(j, n)P(B_{ij}) + \frac{1}{n} \sum_{j=t_i+1}^n N_i^{decod}(j, n)P(B_{ij}) = \frac{1}{n} \sum_{j=t_i+1}^n N_i^{decod}(j, n)P(B_{ij}).$$

$$P_{t_i}^{decod} = \frac{1}{n} \sum_{j=t_i+1}^n N_i^{decod}(j, n) \binom{n}{j} (P_e^i)^j (1 - P_e^i)^{(n-j)}. \quad (\text{III.21})$$

**Fin preuve**

### III.2.3 Choix des codeurs

Pour chaque poids de bit  $i$  et chaque rapport signal à bruit,

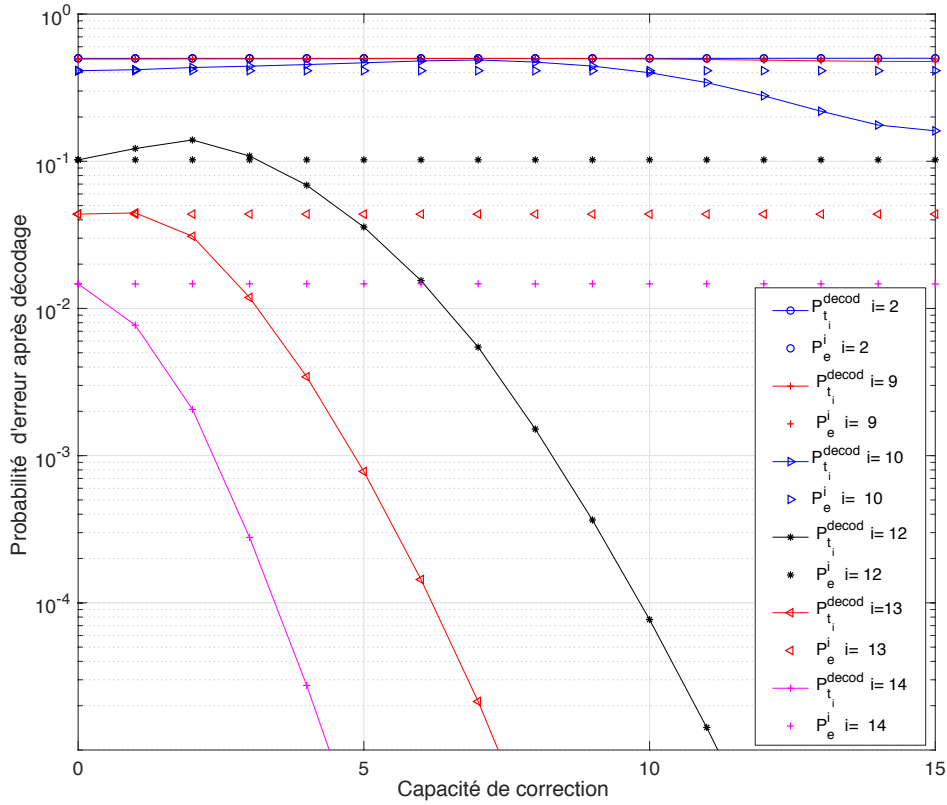


FIGURE III.4 – Probabilité d’erreur avant et après décodage correcteur d’erreurs pour plusieurs poids de bit vs capacités de correction des codeurs, pour un SNR=30 dB. Sous l’hypothèse que les signaux suivent une loi de distribution uniforme et sont bruités par un bruit blanc Gaussien, signaux représentés sur  $L + 1 = 16$  bits.

- la proposition III.2.2 et l’équation III.18 donnent la probabilité d’erreur avant décodage  $P_e^i$ .
- La proposition III.2.3 donne  $P_{t_i}^{decod}$  la probabilité d’erreur après décodage en fonction de la capacité de correction  $t_i$  du codeur.

On peut donc choisir  $t_i$  tel que  $P_{t_i}^{decod}$  soit faible et inférieur à  $P_e^i$ .

La figure III.4 illustre  $P_{t_i}^{decod}$  en fonction de la capacité de correction  $t_i$  pour plusieurs poids de bit, pour un rapport signal à bruit de 30 dB et pour  $n=31$ . À partir de ces résultats, on peut choisir pour chaque rapport signal à bruit et chaque poids de bit la capacité de correction qui assure la réduction de la probabilité d’erreur après décodage par rapport à la probabilité d’erreur avant décodage, conduisant à une augmentation du rapport signal à bruit après décodage.

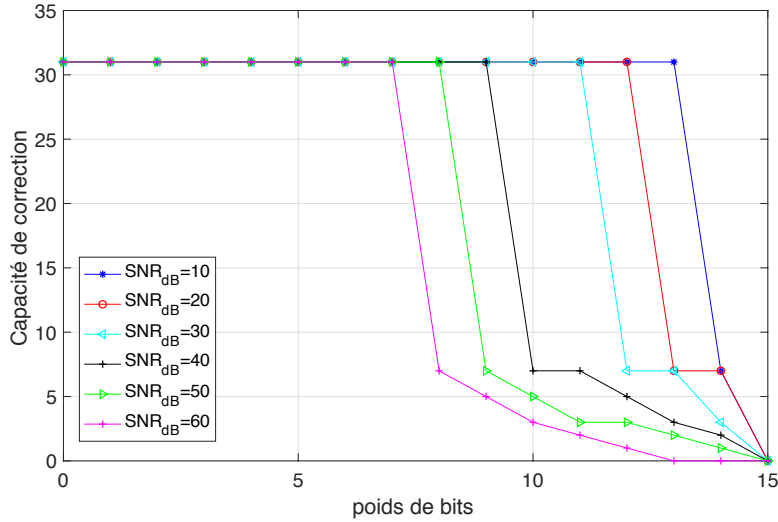


FIGURE III.5 – Capacités de correction minimales  $t_i$  en fonction du poids de bit pour plusieurs valeurs du SNR du canal de transmission et pour des mots de code de longueur  $n = 31$ . Sous l’hypothèse que les signaux suivent une loi de distribution uniforme et sont bruités par un bruit blanc Gaussien, signaux représentés sur 16 bits.

De ces résultats, on en déduit les codeurs correspond à partir de ces capacités de correction selon deux cas possible, pour chaque poids de bit  $i$ ,  $0 \leq i \leq L - 1$  :

- S’il existe des probabilités d’erreurs après décodage inférieures  $10^{-5}$ , on choisit le codeur  $\zeta_i(n_i, k_i, t_i)$  dont la capacité de correction  $t_i$  vérifie la formule ci-dessous :

$$t_i = \min\{t : P_t^{decod} \leq 10^{-5} \text{ et } P_t^{decod} < P_e^i\},$$

- sinon si la probabilité d’erreur après décodage nécessite une capacité de correction supérieure à la capacité maximale permise par la famille de codeurs utilisée, on choisit le codeur de dimension 1, i.e on code par le mot de code  $[00\dots0]$  ou bien par le mot de code  $[11\dots11]$ . Ces poids de bit sont connus par le récepteur, ainsi la probabilité d’erreur après décodage est nulle pour ces poids de bit quel que soit le nombre d’erreurs avant décodage. On a donc une capacité de correction  $t_i = n$  la longueur des codeurs.



FIGURE III.6 – Chaîne de transmission proposée.

La figure III.5 représente pour chaque rapport signal à bruit, les capacités de correction par poids de bit correspondant aux codeurs choisis suivant la méthode illustrée ci-dessus.

### III.3 Application à la réduction du bruit

La chaîne de transmission indiquée à la figure III.6 a été simulée pour plusieurs valeurs du rapport signal à bruit du canal de transmission, le bruit du canal de transmission a été considéré comme bruit additif blanc gaussien. Pour chaque rapport signal à bruit et chaque poids de bit, les courbes de la figure III.5 nous ont permis de choisir les codeurs  $\zeta_i$  en fonction des capacités de correction minimale requises.

Des codeurs BCH [19], [6] de longueur  $n = 31$  dont les capacités de correction sont dans l'ensemble  $\{1, 2, 3, 5, 7\}$  ont été utilisés pour effectuer la quantification. Si pour certains poids de bit  $i$  la capacité de correction  $t_i$  nécessaire pour corriger les erreurs est supérieure à 7, alors un codeur de dimension  $k=1$  et de capacité de correction  $n$  sera utilisé comme indiqué précédemment, avec le mot  $[0,0,\dots,0]$  pour les bits de poids pair et le mot de code  $[1,1,\dots,1]$  pour les bits de poids impair.

Ce codage permet de réduire l'effet du bruit dans la zone des bits de poids faible où on ne dispose pas de codeurs capables de corriger les erreurs. Le tableau Table III.2 indique les dimensions des codeurs choisies pour chaque rapport signal à bruit et pour chaque poids de bit.

#### III.3.1 Application aux signaux à distribution uniforme

Nous avons simulé la chaîne de transmission de la figure III.6 sur des signaux générés aléatoirement suivant une loi de distribution uniforme dans l'intervalle  $A$  défini ci-dessus.

SNR \ Poids	15	14	13	12	11	10	9	8	7	6...0
10 dB	31	6	1	1	1	1	1	1	1	1
20 dB	31	6	6	1	1	1	1	1	1	1
30 dB	31	11	6	6	1	1	1	1	1	1
40 dB	31	21	16	11	6	6	1	1	1	1
50 dB	31	26	21	16	16	11	6	1	1	1
60 dB	31	31	31	26	21	16	11	6	1	1

TABLE III.2 – Dimensions des codeurs BCH choisis pour chaque  $SNR_{dB}$  et pour chaque poids de bit, satisfaisant la condition sur les capacités de correction indiquée à la figure III.5.

Ces signaux ont été transmis via la chaîne de transmission indiquée à la figure Fig III.6 dans le cas d'un bruit blanc gaussien. Pour chaque rapport signal à bruit du canal de transmission, un rapport signal à bruit de décodage (quantification après réception) a été calculé puis comparé au rapport signal à bruit du canal de transmission. Cette opération a été répétée un million de fois sur des trames différentes de  $n = 31$  échantillons temporels pour chaque valeur du rapport signal à bruit du canal de transmission.

Les résultats des simulations sont représentés à la figure III.7. Comme attendu, la quantification assure une réduction du bruit de transmission; le rapport signal à bruit après décodage des signaux bruités (rapport signal à bruit du décodage) est supérieur au rapport signal à bruit du canal de transmission. Le rapport signal à bruit est amélioré de 5 dB à 25 dB selon sa valeur initiale.

### III.3.2 Application aux signaux d'alarmes

Afin d'étudier l'apport de la quantification pour les signaux d'alarmes de voitures prioritaires, nous avons repris la simulation précédente avec un signal d'alarme de sapeurs-pompiers de longueur 10 s échantillonné à la fréquence  $F_s = 44100$  Hz.

Les deux méthodes de quantification proposées ont été utilisées pour ces signaux : dans le cas de la quantification temporelle le signal est divisé en trames de longueur  $n$  afin

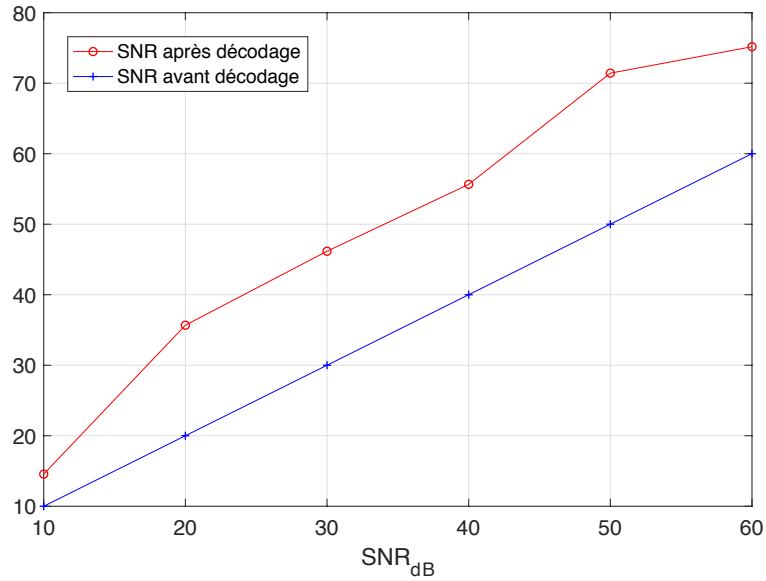


FIGURE III.7 – Rapport signal à bruit (SNR) après décodage, en fonction du SNR avant décodage. Résultats sur  $10^6$  simulations effectuées sur des signaux de taille  $n = 31$  générés aléatoirement suivant une loi de distribution uniforme sur l'intervalle entier  $[-2^L + 1 + 1, 2^L - 1]$

d'effectuer la quantification ; dans le cas de la quantification fréquentielle chaque fenêtre fréquentielle (la transformée temps-fréquence utilisée est la MDCT) de longueur  $\frac{N}{2} = 1024$  est découpée en trames de longueur  $n$ , qui sont par la suite quantifiées. Si le nombre 1024 n'est pas divisible par  $n$ , on quantifie uniquement les  $M = \lfloor \frac{N}{2n} \rfloor$ , les  $(\frac{N}{2} - nM)$  derniers coefficients non quantifiés se trouvant dans les hautes fréquences (pour les signaux d'alarme les coefficients de la MDCT des hautes fréquences sont négligeables par rapport à ceux des basses fréquences, l'information utile de ces signaux se trouve dans les basses fréquences).

Les rapports signal à bruit de décodage des deux quantifications (quantification dans le domaine fréquentiel et quantification dans le domaine temporel) sont représentés sur la figure III.8.

Dans le cas d'une quantification dans le domaine temporel, le rapport signal à bruit est amélioré de 15 dB pour un rapport signal à bruit de 10 dB sur le canal et est infini pour des rapports signal à bruit supérieurs (erreurs parfaitement corrigées).

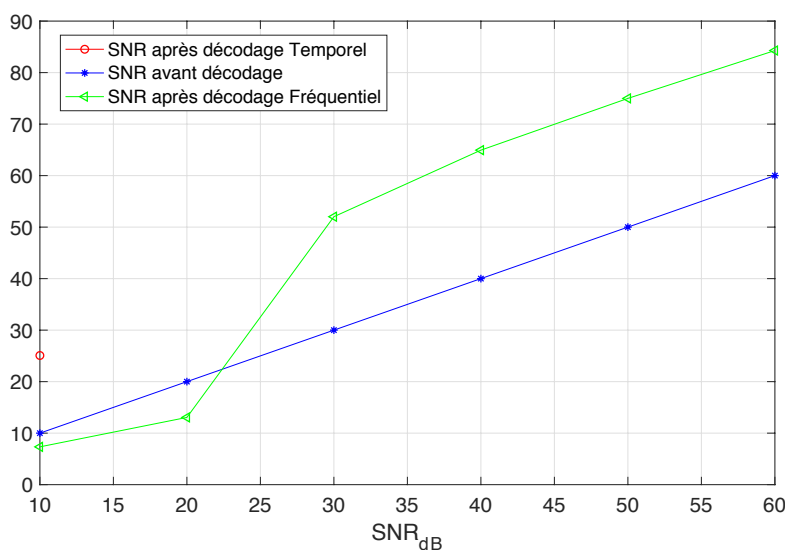


FIGURE III.8 –  $\text{SNR}_{dB}$  après décodage, en fonction du  $\text{SNR}_{dB}$  avant décodage. Résultats des simulations sur un signal d’alarme de sapeurs-pompiers.

Dans le cas d’une quantification dans le domaine fréquentiel, l’amélioration est plus faible (15 dB à 20 dB) pour des rapports signal à bruit supérieurs ou égaux à 30 dB, et le rapport signal à bruit est même dégradé pour des rapports signal à bruit inférieurs ou égaux à 20 dB.

Les signaux d’alarme n’ont pas une distribution uniforme (voir la figure III.9), le comportement du bruit sur ces signaux n’est pas forcément le même que celui sur des signaux uniformes, ce qui explique la différence entre les figures III.8 et III.7.

### III.3.3 Distorsion introduite par la quantification

La réduction du bruit du canal permise par la quantification est au prix d’une déformation du signal original que nous évaluons d’une part en termes du rapport signal à bruit de quantification, d’autre part via l’observation du spectrogramme. La figure III.10 présente le rapport signal à bruit de quantification en fonction du rapport signal à bruit du canal (les codeurs choisis pour la quantification dépendant de ce dernier).

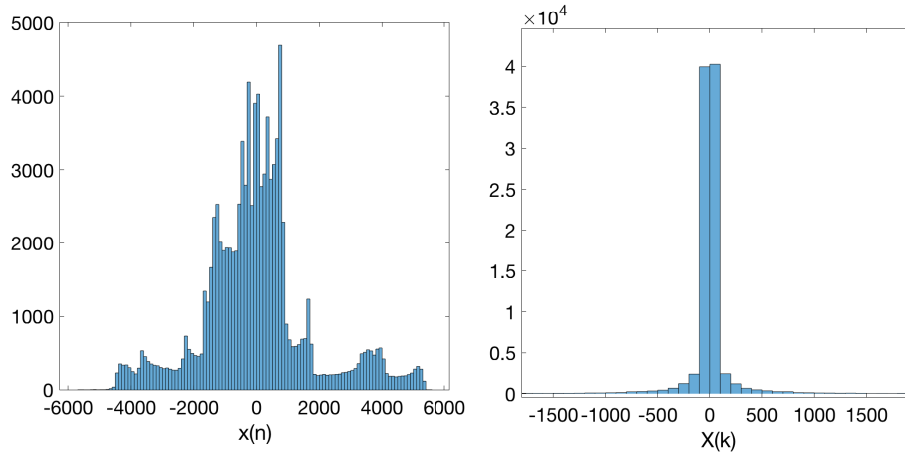


FIGURE III.9 – À gauche histogramme du signal temporel de sapeurs-pompiers, à droite histogramme de la MDCT du signal.

Pour le signal d’alarme de sapeurs-pompiers, la quantification dégrade plus le signal si elle est effectuée dans le domaine fréquentiel, tout en réduisant moins le bruit du canal (voir résultats précédents).

Les figures III.11 et III.12 représentent les spectrogrammes d’un signal d’alarme avant et après quantification, respectivement dans le domaine temporel et dans le domaine fréquentiel, pour un rapport signal à bruit de 30 dB. Dans le cas de la quantification temporelle, la structure du spectrogramme du signal quantifié est la même que celle du signal original. La quantification temporelle affecte peu la perception du signal d’alarme. En revanche, le bruit introduit par la quantification fréquentielle est audible et la structure temps-fréquence du signal quantifié est noyée dans le bruit de quantification, comme l’illustre la figure III.12.

### III.4 Conclusion

Nous avons proposé une méthode de quantification fondée sur les principes du codage de canal qui rend ces signaux robustes au bruit de transmission. Appliquée à des signaux d’alarme, la quantification temporelle assure une protection remarquable des signaux. Pour la quantification fréquentielle, la protection des signaux est obtenue pour des rap-



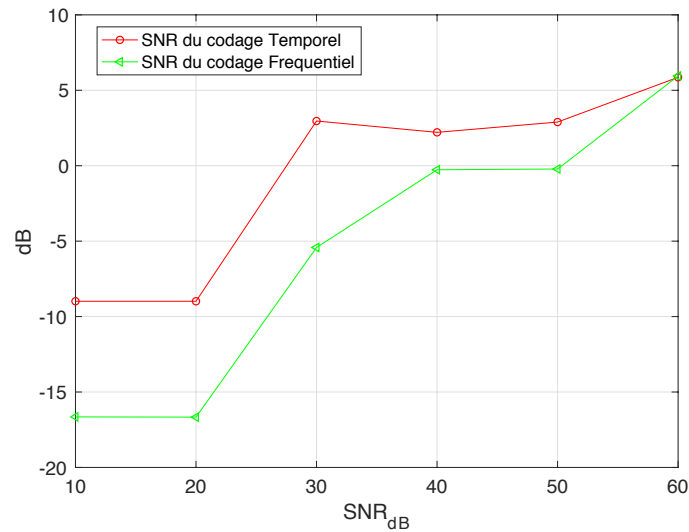


FIGURE III.10 – SNR<sub>dB</sub> de l’erreur du codage, en fonction du SNR<sub>dB</sub> du canal de transmission. Résultats des simulations sur un signal d’alarme de sapeurs-pompiers.

ports signal à bruit supérieurs à 20 dB est plus modeste. Toutefois, une telle protection se paye en termes d’erreurs de quantification. Pour le signal d’alarme testé, les rapports signal à bruit de quantification traduisent une nette altération du signal original. Outre le fait que la dimension des codeurs est supérieure à 1 uniquement pour les bits de poids fort, cette altération peut s’expliquer par le fait que la quantification utilise la distance de Hamming comme métrique binaire du codage, cette métrique réduit l’erreur binaire du codage, mais ne réduit pas l’erreur euclidienne de quantification.

Le chapitre suivant s’attache à minimiser l’erreur de quantification (au sens de la distance euclidienne) tout en assurant une protection suffisante des signaux.

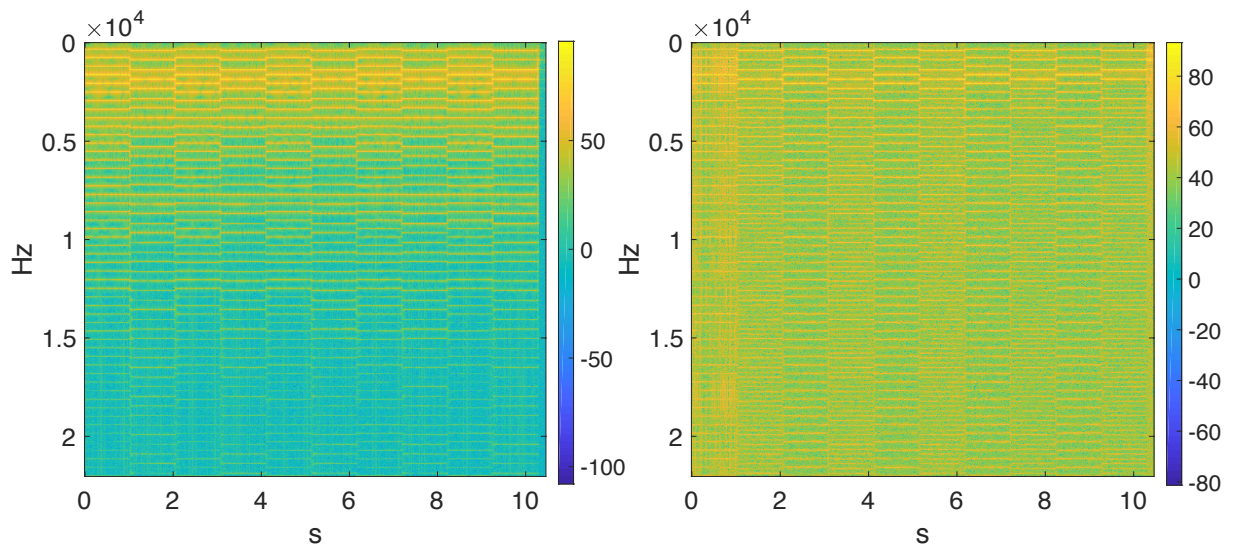


FIGURE III.11 – Spectrogramme d’un signal d’alarme de sapeurs-pompiers : Signal original à gauche et signal quantifié à droite (quantification dans le domaine temporel).

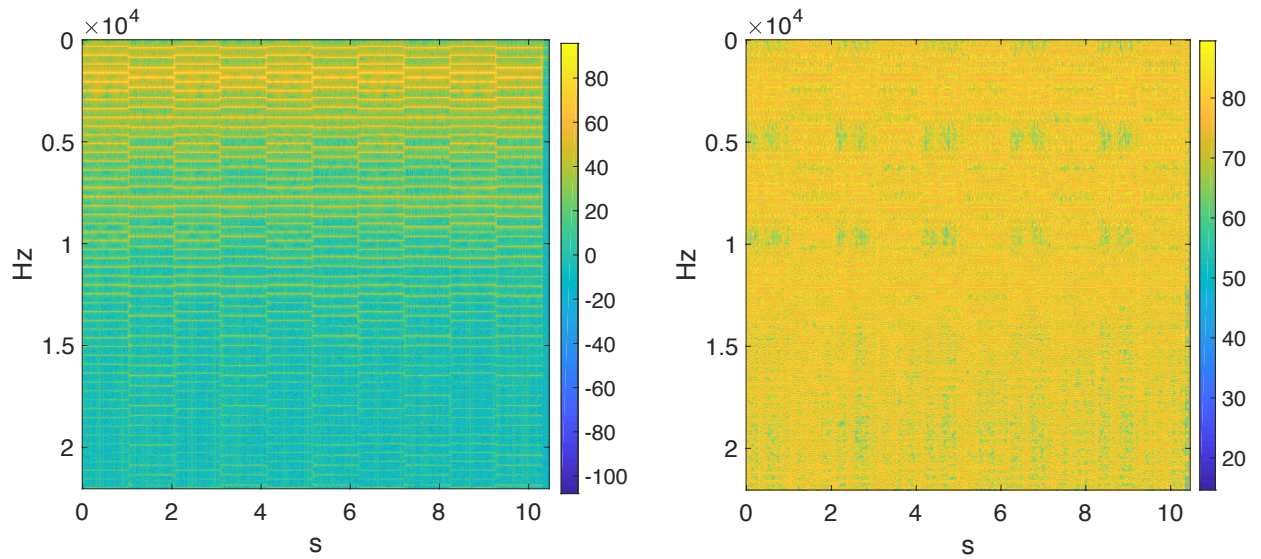


FIGURE III.12 – Spectrogramme du signal original à gauche et signal quantifié à droite (quantification dans le domaine fréquentiel)



# Chapitre IV

## Matching Pursuit pour la quantification

Dans ce chapitre nous cherchons à reformuler le problème de la quantification étudié précédemment dans le but de minimiser l'erreur de quantification.

Comme précédemment, la quantification est une projection sur des dictionnaires de mots de code qui à chaque vecteur binaire  $X_i \in F_2^n = \mathbb{Z}/2\mathbb{Z}^n$  de bits de poids  $i$  ( $0 \leq i \leq L - 1$ ) associe un mot de code  $C_i \in D_i$ . Les  $D_i$  sont des dictionnaires de mots de code générés par des codeurs en bloc  $\zeta_i(n, k_i, t_i)$  (codeurs correcteurs d'erreurs de longueur  $n$  et de dimension  $k_i$ ).

Soit  $s \in A^n$  la trame de longueur  $n$  d'un signal  $x$ , définie dans le chapitre précédent. On cherche à quantifier la trame  $s$  par la trame  $\hat{s}$  tout en minimisant l'erreur de quantification, où  $s$  et  $\hat{s}$  sont données par :

$$s = M_s X^t,$$

$$\hat{s} = M_s \hat{X}^t,$$

avec  $\hat{X}$  la trame quantifiée de  $X$ , donnée par :

$$\hat{X} = \sum_{i=0}^{L-1} 2^i C_i.$$

L'erreur de quantification de la trame  $s$  est donnée par :

$$E = \|s - \hat{s}\|_2^2 = \|X - \hat{X}\|_2^2.$$

L'objectif ici est de déterminer la combinaison de mots de code  $(C_0, \dots, C_{L-1}) \in D_0 \times \dots \times D_{L-1}$  qui minimise l'erreur de quantification  $E$ . Ce problème peut être écrit analytiquement comme suit :

Soit  $H = [D_0 \ 2D_1 \dots 2^i D_i \dots 2^{L-1} D_{L-1}]$  la matrice formée par les dictionnaires de mots de code, soit  $d = \sum_{i=0}^{L-1} 2^{k_i}$  le nombre de colonnes de  $H$ ,  $H$  est une matrice  $(n \times d)$  et soit  $\alpha = (\alpha_0, \dots, \alpha_i, \dots, \alpha_{L-1})$  un vecteur dans  $\mathbb{F}_2^d$  formé par la concaténation de plusieurs vecteurs  $\alpha_i$  tel que  $\|\alpha\|_1 = L$  et  $\alpha_i$  des vecteurs de  $\mathbb{F}_2^{k_i}$  vérifiant  $\|\alpha_i\|_1 = 1$ . Alors le signal  $\hat{X}$  qui minimise l'erreur de quantification sera donné par :

$$\hat{X} = H\hat{\alpha},$$

où  $\hat{\alpha}$  est donné par :

$$\hat{\alpha} = \arg \min_{\alpha \in \Gamma} \|X - H\alpha\|_2^2, \quad (\text{IV.1})$$

avec  $\Gamma$  un sous ensemble de  $\mathbb{F}_2^d$  donné par :  $\Gamma = \{y \in \mathbb{F}_2^d, \|\alpha\|_1 = L : \alpha = (\alpha_0, \dots, \alpha_{L-1})$  et  $\|\alpha_i\|_1 = 1\}$ .

Le problème global consiste à trouver les mots de code  $C_i$  tels que l'erreur de quantification  $E$  soit minimale est équivalent à celui de retrouver le vecteur  $\hat{\alpha}$ . Cependant la dimension de la matrice  $H$  est très grande de même que la dimension de l'espace  $\Gamma$ .

Une recherche exhaustive nécessiterait de choisir parmi  $\prod_{i=0}^{L-1} 2^{k_i}$  éléments : la recherche d'un tel vecteur peut prendre un temps de calcul très élevé. Un algorithme de programmation linéaire comme ceux du basis pursuit [9], [36], [52] peut être utilisé pour trouver la solution de ce problème. Toutefois, la contrainte posée sur les vecteurs  $\alpha$  est très forte

ce qui fait que le système à l'équation (IV.1) peut parfois ne pas avoir de solution.

Pour éviter les obstacles liés à la recherche d'une solution globale, nous avons donc choisi d'utiliser une approche itérative de type matching pursuit, de manière à obtenir une bonne approximation du minimum de l'erreur de quantification.

Nous nous attachons dans ce chapitre à rapprocher la méthode du chapitre III de l'objectif de minimisation de l'erreur de quantification propre à la quantification vectorielle, tout en limitant la complexité inhérente à cette optimisation.

## IV.1 Algorithme du matching pursuit modifié pour la quantification

Afin de déterminer les mots de code  $C_i$  pour lesquels on obtient une bonne approximation de l'erreur de quantification minimale, nous avons procédé par une recherche itérative mono-dimensionnelle détaillée dans l'algorithme ci-dessous. Cet algorithme est une nouvelle version du Matching Pursuit adaptée à notre problème.

Le nombre d'itérations de cet algorithme est égal au nombre  $L$  de bits sur lesquels sont représentés les signaux sans compter les bits de signe (car ces bits ne sont pas concernés par la quantification). Chaque itération de cet algorithme correspond à une quantification par poids de bit. Pour tout poids de bit  $i$ ,  $0 \leq i \leq L - 1$ , la quantification par poids de bit peut être considérée comme une application  $f_i$  définie par :

$$f_i : A^n \times F_2^n \longrightarrow D_i$$

$$(X, X_i) \longrightarrow C_i$$

La fonction  $f_i$  effectue une projection de  $F_2^n$  dans un sous-espace de mots de code, le dictionnaire  $D_i$ .

---

**Algorithme 2** : Algorithme du Matching Pursuit Modifié pour la quantification

---

**Entrée** :  $X$  le signal à quantifier,  $D_i, i = 0..L - 1$ .

```
1 Initialisation :  $\hat{X} \leftarrow 0$   $R \leftarrow X$  ;
2 for  $l \leftarrow L - 1$  to  $0$  do
3    $C_l \leftarrow \arg \max_{C \in D_l} Corr(2^l C, R)$ ;
4    $R = R - 2^l C_l$ ;
5    $\hat{X} \leftarrow \hat{X} + 2^l C_l$ ;
6 end
```

**Sortie** :  $\hat{X}$

---

En développant l'expression de l'erreur de quantification on obtient :

$$\begin{aligned} E &= \|X - \hat{X}\|_2^2 = \|X\|_2^2 - 2 \langle X, \hat{X} \rangle + \|\hat{X}\|_2^2 \\ &= \|X\|_2^2 - 2 \langle X, \sum_{i=0}^{L-1} 2^i C_i \rangle + \|\sum_{i=0}^{L-1} 2^i C_i\|_2^2 \\ &= \|X\|_2^2 - 2 \sum_{i=0}^{L-1} \langle X, 2^i C_i \rangle + \|\sum_{i=0}^{L-1} 2^i C_i\|_2^2. \end{aligned} \tag{IV.2}$$

On remarque sur la figure IV.1 que l'algorithme 2 ne réduit pas la norme du résidu par itération. Ceci provient du fait que le critère de maximisation de la corrélation privilégie les mots de code ayant un poids de Hamming très grand. Or on observe à l'équation (IV.2) que la minimisation de l'erreur de quantification ne dépend pas uniquement de la corrélation mais dépend aussi d'un troisième terme d'autant plus grand que le poids de Hamming est grand.

Dans le but de pouvoir lier le choix des mots de code  $C_l$  à l'erreur de quantification, l'étape suivante consiste à déterminer une fonction coût  $f$  qui va remplacer la corrélation au point 3 de l'algorithme 2 de manière à approcher au mieux le minimum de l'erreur de quantification.

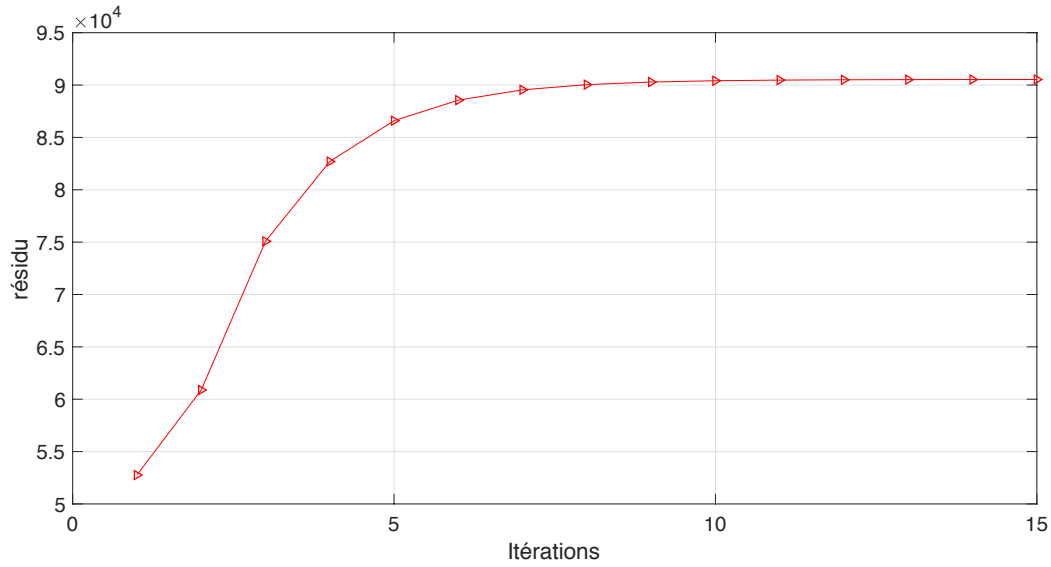


FIGURE IV.1 – La norme du résidu à chaque itération de l’algorithme ; l’erreur de quantification est égale au résidu à la dernière itération. Simulations effectuées sur un nombre  $N = 10^5$  de trames de longueur  $n$  et de distribution uniforme dans  $A^n$ .

### IV.1.1 Minimisation du résidu à chaque itération

À chaque itération  $l$ , au lieu de rechercher le mot de code  $C_l$  qui maximise la corrélation  $\langle 2^l C, R \rangle$ , nous recherchons le mot de code  $C_l$  qui minimise directement la norme  $E_l = \|R - 2^l C\|_2^2$  du résidu.

La norme du résidu à la dernière itération de l’algorithme est égale à l’erreur de quantification. En effet, le résidu par itération est donné par  $R_l = R_{l+1} - 2^l C_l$ ,  $0 \leq l \leq L - 1$ , avec  $R_L = X$ . Ces résidus sont liés par la relation de récurrence suivante :

$$R_{L-1} = X - 2^{L-1} C_{L-1}.$$

$$R_{L-2} = R_{L-1} - 2^{L-2} C_{L-2} = X - 2^{L-1} C_{L-1} - 2^{L-2} C_{L-2}.$$

.

.

$$R_0 = R_1 - 2^0 C_0 = X - \sum_{i=1}^{L-1} 2^i C_i - 2^0 C_0 = X - \hat{X}.$$

À la dernière itération de l’algorithme  $l = 0$ ,  $R_0 = X - \hat{X}$  et l’erreur de quantification



$E$  est donnée par :

$$E = \|R_0\|_2^2. \quad (\text{IV.3})$$

Intuitivement, le remplacement de la maximisation de la corrélation  $\langle 2^l C, R \rangle$  à chaque itération par la minimisation de la norme du résidu correspondant,  $R_l = R_{l+1} - 2^l C$ , devrait minimiser l'erreur de quantification globale.

La norme du résidu  $R_l$  est donnée par :

$$E_l = \|R_{l+1} - 2^l C\|_2^2 = \|R_{l+1}\|_2^2 - 2 \cdot 2^l R_{l+1} C^t + 2^{2l} C C^t.$$

$$E_l = \|R_{l+1}\|_2^2 - 2^{l+1} (R_{l+1} C^t - 2^{l-1} P_c).$$

$$E_l = \|R_{l+1}\|_2^2 - 2^{l+1} (\langle C, R_{l+1} \rangle - 2^{l-1} P_c), \quad (\text{IV.4})$$

avec  $P_c = C C^t$  le poids de Hamming de  $C$ .

D'après (IV.4), un mot de code qui possède un poids de Hamming élevé augmente le produit scalaire mais en parallèle augmente la norme du résidu. Ainsi, la maximisation de la corrélation, sans contrainte supplémentaire, ne permet pas de minimiser l'erreur résiduelle à chaque itération, et sans doute pas l'erreur de quantification globale.

Par conséquent, pour minimiser l'erreur résiduelle à chaque itération, le point 3 de l'algorithme sera remplacé par :

$$C_l = \arg \max_{C \in D_l} f(C, R). \quad (\text{IV.5})$$

avec,

$$f : A^n \times D_l \longrightarrow \mathbb{Z}$$

$$(R, C) \longrightarrow \langle C, R \rangle - 2^{l-1} P_c.$$

Le nombre d'itérations de l'algorithme est limité à  $L = 15$  itérations correspondant au nombre de poids de bit qui seront codés, contrairement à un algorithme classique

du Matching Pursuit. Ce nombre d'itérations n'est pas suffisant pour bien approcher le minimum de l'erreur de quantification. Dans la suite nous cherchons à définir une fonction coût tenant compte de l'objectif final de minimisation de l'erreur de quantification.

### IV.1.2 Minimisation de l'erreur finale de quantification : approche empirique

Maximiser le produit scalaire  $\langle C, X \rangle$  favorise les mots de code avec un fort poids de Hamming, ce qui augmente la norme du résidu à chaque itération, donnée par l'équation (IV.4). La fonction coût proposée dans l'algorithme 2 ne tenait pas compte de l'effet du poids de Hamming. Dans l'objectif de minimiser l'erreur finale de quantification, nous proposons ici de pénaliser le poids de Hamming comme dans l'équation (IV.5), mais avec un coefficient de pénalisation  $\lambda$  qui permettra une réduction de l'erreur de quantification finale.

Pour avoir un meilleur compromis entre la maximisation du produit scalaire et la minimisation du poids de Hamming, on définit une nouvelle fonction coût  $J_{\lambda,l}$  donnée par :

$$J_{\lambda,l}(C, R) = \langle C, R \rangle - \lambda 2^l P_c.$$

Pour que le poids de Hamming  $P_c$  soit comparable avec le produit scalaire  $\langle C, R \rangle$  il a été multiplié par  $2^l$  qui correspond au coefficient multipliant le vecteur binaire des bits de poids  $l$  provenant de la combinaison linéaire formant  $X$ .

Alors, pour chaque itération  $l$ , le mot de code  $C_l$  qui représente au mieux la trame  $X$ , par pénalisation  $\lambda$  est donné par :

$$C_l^\lambda = \arg \max_{C \in D_l} J_{\lambda,l}(C, R)$$

Si  $\lambda = 1/2$ , on retrouve le résultat précédent correspondant à la minimisation du

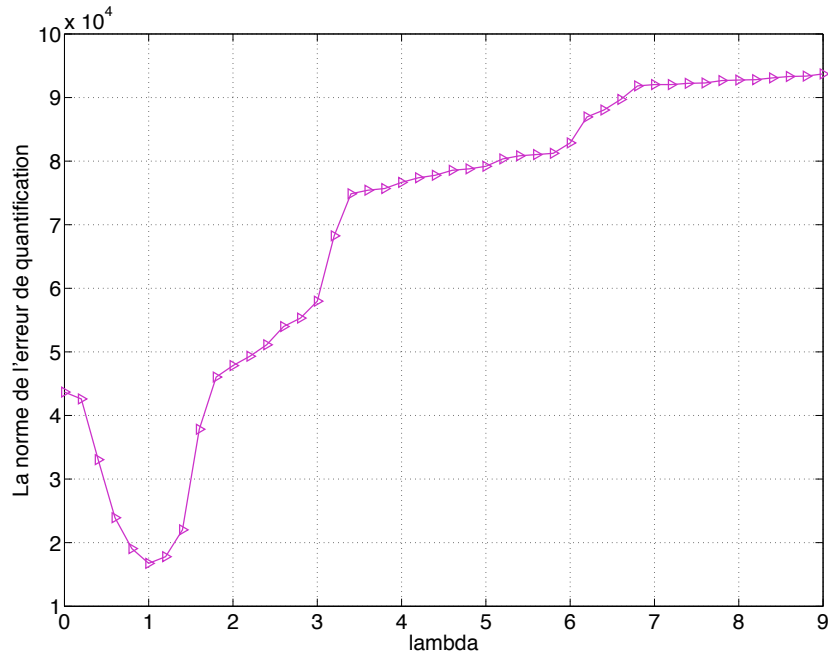


FIGURE IV.2 – La norme  $E^\lambda$  de l'erreur de quantification en fonction de  $\lambda$ .

résidu à chaque itération de l'algorithme, voir l'équation (IV.5).

### Recherche du $\lambda$ qui minimise l'erreur de quantification :

Soit  $E^\lambda$  l'erreur de quantification par pénalisation  $\lambda$ ,  $E^\lambda$  est donnée par :

$$E^\lambda = \|X - \hat{X}_\lambda\|_2^2.$$

$\hat{X}_\lambda = \sum_{i=0}^{L-1} 2^i C_i^\lambda$ , le signal quantifié en fonction de  $\lambda$ .

Des expériences ont été effectuées sur  $10^5$  trames de longueur  $n$  à distribution uniforme générées aléatoirement dans l'alphabet  $A^n$ , où ces signaux ont été quantifiés à l'aide de l'algorithme 2 avec comme fonction coût  $J_{\lambda,i}$ . Les dictionnaires utilisés dans ces expériences sont des dictionnaires de mots de code générés par des codeurs BCH.

Pour ces expériences on a utilisé un dictionnaire de mots de code généré par le codeur

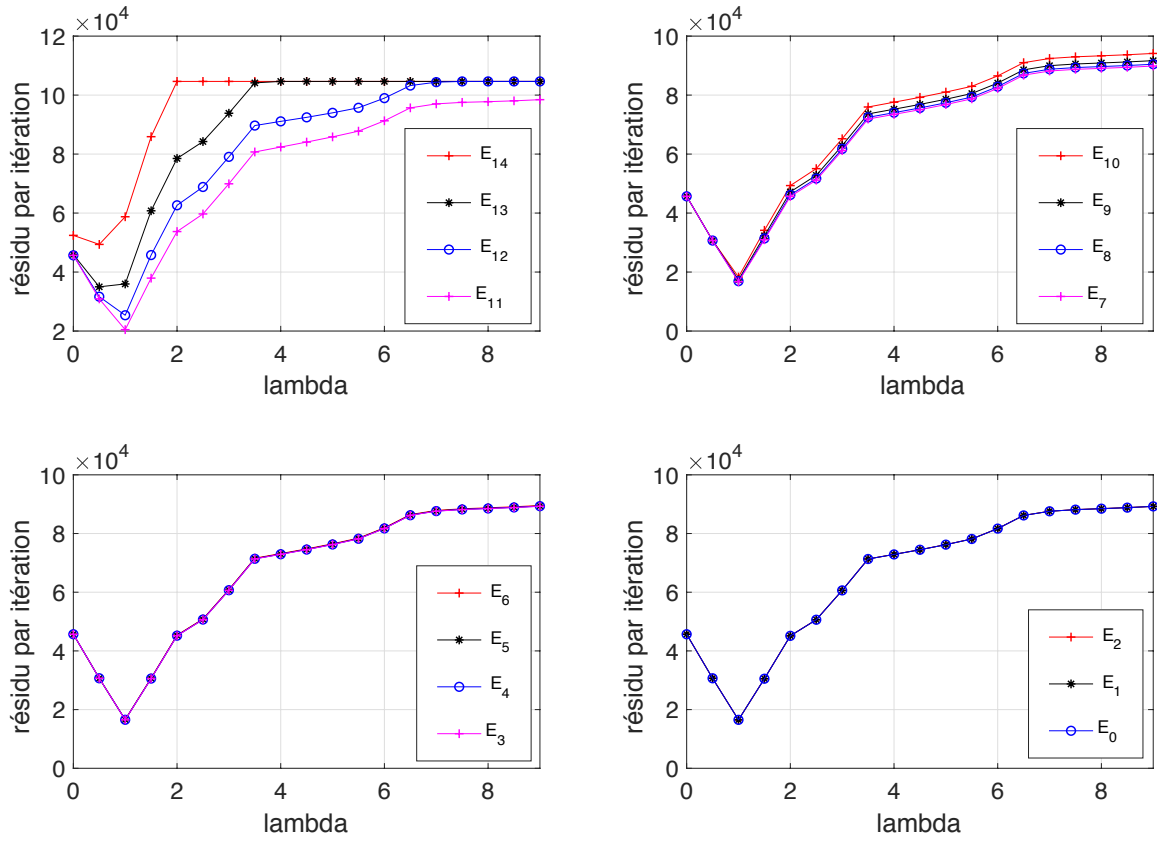


FIGURE IV.3 – La norme  $E_l$  du résidu par itération en fonction de  $\lambda$ ,  $0 \leq l \leq L - 1$ .

$\zeta(n = 31, k = 16, t = 3)$ , ce dictionnaire est pris identique pour toutes les itérations de l'algorithme 2,  $\zeta_i(n, k_i, t_i) = \zeta(31, 16, 3)$ . Ce choix n'est pas basé sur une contrainte spécifique, on peut prendre n'importe quel dictionnaire de mots de code, l'objectif est de faire une comparaison des erreurs de quantification pour plusieurs valeurs de  $\lambda$ .

Pour chaque valeur de  $\lambda$  une moyenne des résidus par itération a été calculée, voir la figure IV.3. Pour chaque valeur de  $\lambda$  l'erreur de quantification moyenne  $E^\lambda$  a été calculée, voir la figure IV.2. Les valeurs de  $\lambda$  sont prises dans l'intervalle  $[0, 9]$ .

D'après ces simulations l'erreur de quantification  $E^\lambda$  présente un minimum pour  $\lambda = 1$  bien que les résidus à chaque itération ne présentent pas tous des minima pour  $\lambda = 1$  (voir les figures IV.3 et IV.2).

On peut en déduire que l'approche précédente IV.1.1 fondée sur la minimisation de la norme du résidu à chaque itération correspondant à cette approche avec  $\lambda = \frac{1}{2}$ , ne minimise pas l'erreur de quantification finale.

Les expériences effectuées ci-dessus suggèrent que le compromis entre la maximisation du produit scalaire et la minimisation du poids de Hamming assure une réduction de l'erreur de quantification.

### IV.1.3 Minimisation de l'erreur finale de quantification : approche théorique

Nous avons obtenu empiriquement dans la section précédente une fonction coût permettant de minimiser l'erreur de quantification dans le cadre de l'algorithme du Matching Pursuit. Nous cherchons dans cette partie à trouver une autre fonction coût  $f_l$  de manière théorique. Pour chaque itération  $l$ , cette fonction est définie de  $A^n \times D_l$  dans  $\mathbb{R}$  par :

$$f_l : A^n \times D_l \longrightarrow \mathbb{R}$$

$$(R, C) \longrightarrow f_l(R, C)$$

À l'itération  $l$ ,  $R = [r_1, \dots, r_j, \dots, r_n] \in A^n$  est le résidu à partir duquel on cherche à approcher le vecteur binaire  $X_l = (x_1, \dots, x_j, \dots, x_n)$  par un mot de code  $C = (c_1, \dots, c_j, \dots, c_n) \in D_l$ . Cette approximation doit être telle que l'erreur finale du codage soit minimale.

À l'itération  $l$  et dans le cas idéal où le codage des étapes précédentes n'introduit pas d'erreurs, les composantes  $r_j$  du résidu sont ou bien encadrées entre 0 et  $2^l$ ,  $0 \leq r_j < 2^l$ , ou bien elles sont encadrées entre  $2^l$  et  $2^{l+1}$ ,  $2^l \leq r_j < 2^{l+1}$ ,  $1 \leq j \leq n$ . Dans le cas idéal les valeurs de  $x_j$  sont comme suit :

- $x_j=1$  si  $2^l \leq r_j < 2^{l+1}$ .
- $x_j=0$  si  $0 \leq r_j < 2^l$ .

Dans le cas non idéal où le codage des étapes précédentes introduit des erreurs, les composantes  $r_j$  peuvent prendre des valeurs négatives et des valeurs supérieures ou égales à  $2^{l+1}$  en plus des deux cas précédents ( $0 \leq r_j < 2^l$  et  $2^l \leq r_j < 2^{l+1}$ ). Ainsi dans ce qui suit nous exploitons les valeurs des composantes  $r_j$  du résidu pour exprimer l'erreur introduite par le codage à chaque itération  $l$  de l'algorithme.

I . Si  $0 \leq r_j < 2^l$ , alors l'erreur introduite par cette étape dans le résultat final est donnée par :

$$e_j^l = \begin{cases} r_j - 2^l, & \text{si } c_j = 1 \\ 0, & \text{si } c_j = 0 \end{cases} \iff e_j^l = (r_j - 2^l)c_j.$$

Pour l'itération suivante, la composante  $r_j$  du résidu devient  $r_j - 2^l c_j$ .

II . Si  $2^l \leq r_j < 2^{l+1}$ , alors l'erreur introduite par cette étape dans le résultat final est donnée par :

$$e_j^l = \begin{cases} 2^l, & \text{si } c_j = 0 \\ 0, & \text{si } c_j = 1 \end{cases} \iff e_j^l = (1 - c_j)2^l.$$

Pour l'itération suivante, la composante  $r_j$  du résidu devient  $r_j - 2^l c_j$ .

III . Si  $2^{l+1} \leq r_j$ , alors l'erreur introduite par cette étape dans le résultat final est donnée par :

$$e_j^l = \begin{cases} -2^l, & \text{si } x_j = 0 \text{ et } c_j = 1 \\ 2^l, & \text{si } x_j = 1 \text{ et } c_j = 0 \\ 0, & \text{sinon} \end{cases} \iff e_j^l = (x_j - c_j)2^l.$$

Pour l'itération suivante, la composante  $r_j$  du résidu devient  $r_j - 2^l c_j$ .

IV . Si  $r_j < 0$ , alors l'erreur introduite par cette étape dans le résultat final est donnée par :

$$e_j^l = \begin{cases} -2^l, & \text{si } x_j = 0 \text{ et } c_j = 1 \\ 2^l, & \text{si } x_j = 1 \text{ et } c_j = 0 \\ 0, & \text{sinon} \end{cases} \iff e_j^l = (x_j - c_j)2^l.$$

Pour l'itération suivante, la composante  $r_j$  du résidu devient  $r_j - 2^l c_j$ .

Dans le 1<sup>er</sup> cas,

- si  $c_j = 0$ , alors on reste dans ce même cas pour l'étape suivante ;
- si  $c_j = 1$ , alors  $e_j^l < 0$  et le nouveau résidu  $r_j < 0$  devient le quatrième cas pour l'étape suivante.

Dans le second cas,

- si  $c_j = 1$ , alors deux cas sont possibles pour l'étape suivante ; ou bien on reste dans ce même cas ou bien on passe au premier cas ;
- si  $c_j = 0$ , alors  $e_j^l > 0$  et le nouveau résidu  $r_j$  devient dans le troisième cas pour l'étape suivante.

Dans le troisième cas,

- si  $c_j = x_j$ , alors on reste dans le même cas pour l'étape suivante ;
- si  $c_j = 0$  et  $x_j = 1$ , alors  $e_j^l > 0$  et le nouveau résidu  $r_j$  reste dans le même cas pour l'étape suivante,
- si  $c_j = 1$  et  $x_j = 0$ , alors  $e_j^l < 0$  et le nouveau résidu  $r_j$  reste dans le même cas pour l'étape suivante.

Dans le quatrième cas,

- si  $c_j = x_j$ , alors on reste dans le même cas pour l'étape suivante ;
- si  $c_j = 0$  et  $x_j = 1$ , alors  $e_j^l > 0$  et le nouveau résidu  $r_j$  reste dans le même cas pour

l'étape suivante,

- si  $c_j = 1$  et  $x_j = 0$ , alors  $e_j^l < 0$  et le nouveau résidu  $r_j$  reste dans le même cas pour l'étape suivante.

La formation des erreurs successives peut être synthétisée par le diagramme d'état de la figure IV.4.

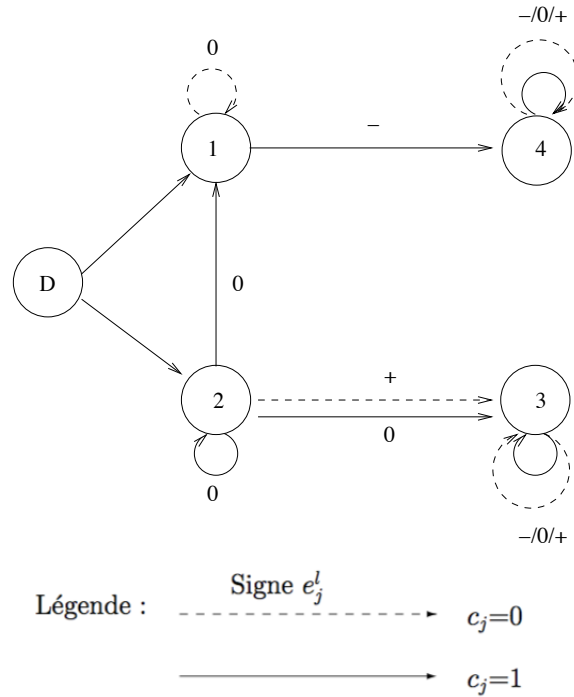


FIGURE IV.4 – Diagramme d'état schématisant la formation des erreurs au cours de l'algorithme du matching poursuit.

L'objectif de minimisation de l'erreur finale de quantification se traduit par :

$$\min \sum_{j=1}^n \left( \sum_{l=0}^{L-1} e_j^l \right)^2$$

À cause des termes croisés  $e^l e^{l'}$  il n'est pas possible de minimiser cette somme par un processus itératif de la forme de l'algorithme 2. À la place, nous proposons de minimiser à chaque itération  $l$  la somme des valeurs absolues des  $n$  erreurs élémentaires  $e_j^l$ ,  $j = 1 \dots n$ .



À chaque itération  $l$  on minimise la somme suivante :

$$\begin{aligned} \min\left(\sum_{j=1}^n |e_j^l|\right) = & \min\left(\sum_{j|0 \leq r_j < 2^l} (2^l - r_j)c_j + \sum_{j|2^l \leq r_j < 2^{l+1}} 2^l(1 - c_j)\right) \\ & + \sum_{j|r_j \geq 2^{l+1}} 2^l(x_j - c_j)^2 + \sum_{j|r_j < 0} 2^l(x_j - c_j)^2 \end{aligned} \quad (\text{IV.6})$$

Dans l'équation (IV.6),  $|x_j - c_j|$  est remplacée par  $(x_j - c_j)^2$  car,  $(x_j - c_j)$  est de la forme de  $y = \pm z$  avec  $z$  un binaire, ainsi dans ce cas  $|x_j - c_j| = (x_j - c_j)^2$  et  $(x_j - c_j)^2 = (x_j - 2c_jx_j + c_j)$ .

$$\begin{aligned} \min\left(\sum_{j=1}^n |e_j^l|\right) = & \min\left(\sum_{j|0 \leq r_j < 2^l} (2^l - r_j)c_j - \sum_{j|2^l \leq r_j < 2^{l+1}} 2^l c_j + \sum_{j|r_j \geq 2^{l+1}; j|r_j < 0} 2^l(x_j - 2c_jx_j + c_j)\right) \\ \min\left(\sum_{j=1}^n |e_j^l|\right) = & \min\left(\sum_{j|0 \leq r_j < 2^l} (2^l - r_j)c_j - \sum_{j|2^l \leq r_j < 2^{l+1}} 2^l c_j + \sum_{j|r_j \geq 2^{l+1}; j|r_j < 0} 2^l(1 - 2x_j)c_j\right) \\ = & \max\left(\sum_{j|2^l \leq r_j < 2^{l+1}} 2^l c_j + \sum_{j|0 \leq r_j < 2^l} (r_j - 2^l)c_j + \sum_{j|r_j \geq 2^{l+1}; j|r_j < 0} 2^l(2x_j - 1)c_j\right). \end{aligned} \quad (\text{IV.7})$$

On pose :

$$f_l(R, C) = \sum_{j|2^l \leq r_j < 2^{l+1}} 2^l c_j + \sum_{j|0 \leq r_j < 2^l} (r_j - 2^l)c_j + \sum_{j|r_j \geq 2^{l+1}; j|r_j < 0} 2^l(2x_j - 1)c_j, \quad (\text{IV.8})$$

Le mot de code cherché est ainsi donné par :

$$C_l = \arg \max_{C \in D_l} f_l(R, C). \quad (\text{IV.9})$$

Lors du décodage, les  $x_j$  ne sont pas connus ; ils sont remplacés par leur version bruitée disponible en réception.

En utilisant les mêmes conditions expérimentales que dans la section IV.1.2, nous avons comparé les résidus successifs de l'algorithme du Matching Pursuit modifié pour la

quantification utilisant la fonction coût  $J_{1,l}$  et l'algorithme du Matching Pursuit utilisant la fonction coût  $f_l$ . Les deux fonctions conduisent à des résidus et une erreur de quantification finale très proches, quoique légèrement inférieurs pour la fonction  $J_{1,l}$ .

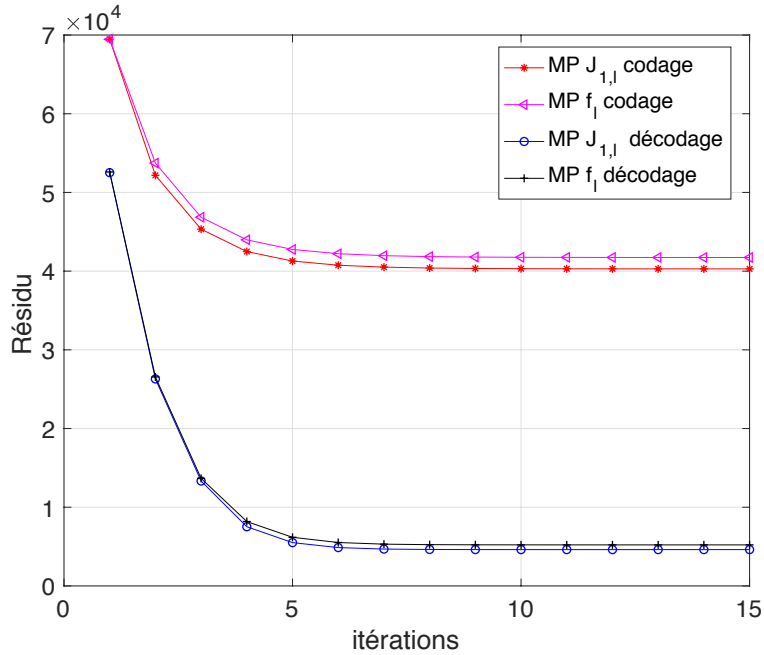


FIGURE IV.5 – Résidu du codage et du décodage par itération :  $f_l$  vs  $J_{1,l}$ , pour un bruit gaussien et un SNR=20 dB.

D'autres simulations ont été effectuées sur  $10^5$  expériences pour tester la correction binaire de ces deux méthodes. Le dictionnaire utilisé dans ces expériences est un dictionnaire de mots de code généré par le codeur BCH  $\zeta(n = 31, k = 11)$ . La figure IV.6 représente les probabilités d'erreur après décodage par poids de bit, pour des signaux à distribution uniforme bruités par un bruit gaussien.

En termes de correction binaire les deux fonctions conduisent à des résultats différents pour les bits de poids le plus faible (bits de poids 0), pour le reste des poids de bit les résultats sont proches et légèrement meilleurs pour la fonction  $J_{1,l}$ .

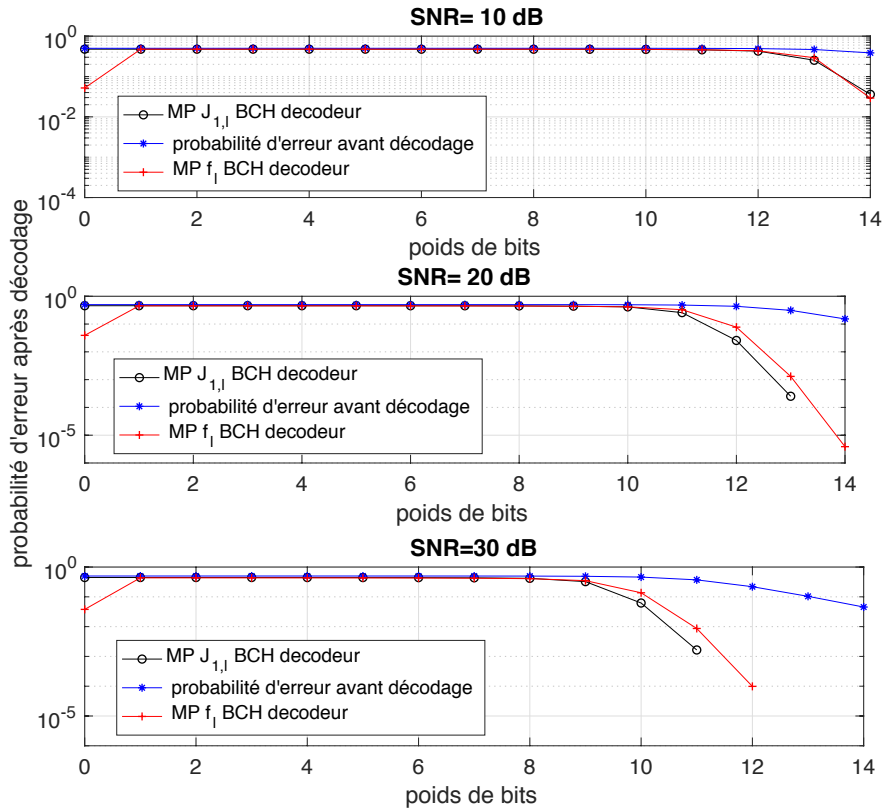


FIGURE IV.6 – Probabilités d’erreur après décodage : décodage utilisant  $f_l$  vs décodage utilisant  $J_{1,l}$ , signaux à distribution uniforme, bruit gaussien.

## IV.2 Comparaison des codes BCH avec des turbo-codes

Pour cette comparaison, on utilise un codeur BCH et un turbo-codeur de rendements similaires :  $R = \frac{11}{31} = \frac{k}{n}$  pour le code BCH et  $R = \frac{20}{72}$  pour le turbo-code. Ces codeurs ont été utilisés pour générer deux dictionnaires de mots de code qui seront par la suite utilisés pour effectuer la quantification. Des expériences ont été effectuées sur  $10^6$  trames à distribution uniforme dans  $\mathcal{A}^n$  de longueur  $n = 31$  dans le cas des codes BCH et de longueur  $n = 72$  dans le cas des turbo-codes. La chaîne de transmission III.6 du chapitre III a été simulée pour deux cas différents : une quantification effectuée par le dictionnaire de mots de code généré par le turbo code et une quantification effectuée par un dictionnaire de mots de code BCH. Les dictionnaires  $D_i$  par poids sont identiques dans ces expériences.

Ainsi un résidu moyen par itération et des probabilités d'erreur après décodage par poids de bit ont été calculées.

La quantification fondée sur l'algorithme du matching pursuit utilisant les dictionnaires de codes BCH conduit à une erreur de quantification et une erreur euclidienne de décodage plus petites que celles introduites par une quantification utilisant des dictionnaires de turbo codes de rendement  $R = k/(3k + 12)$ , voir les figures IV.7 et IV.8. Par contre quand il s'agit de la correction d'erreurs binaire les dictionnaires de turbo codes sont plus robustes que les dictionnaires de codes BCH, ceci provient des propriétés des turbo codes ; le pouvoir de correction des turbo codes est plus grand que celui des codes BCH, voir la figure IV.9.

Malgré l'efficacité des dictionnaires de turbo code en termes de correction d'erreurs binaire, les dictionnaires de codes BCH seront plus adaptés pour la quantification.

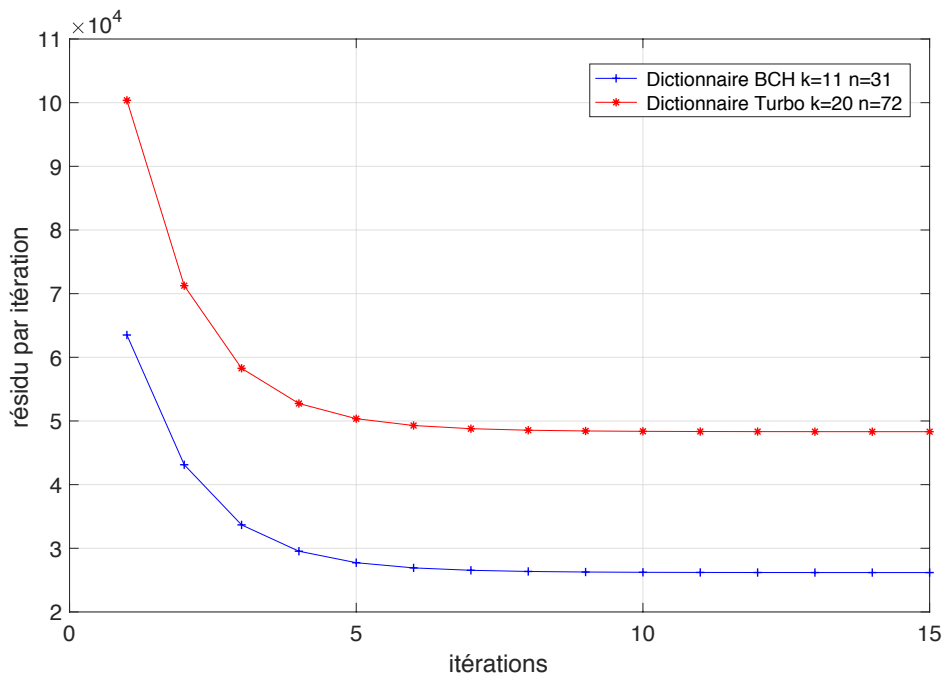


FIGURE IV.7 – Résidu du codage du Matching pursuit : dictionnaire de Turbo Code  $\zeta(n = 72, k = 20, R = k/(3k + 12))$  et dictionnaire de codes BCH  $\zeta(n = 31, k = 11)$ .

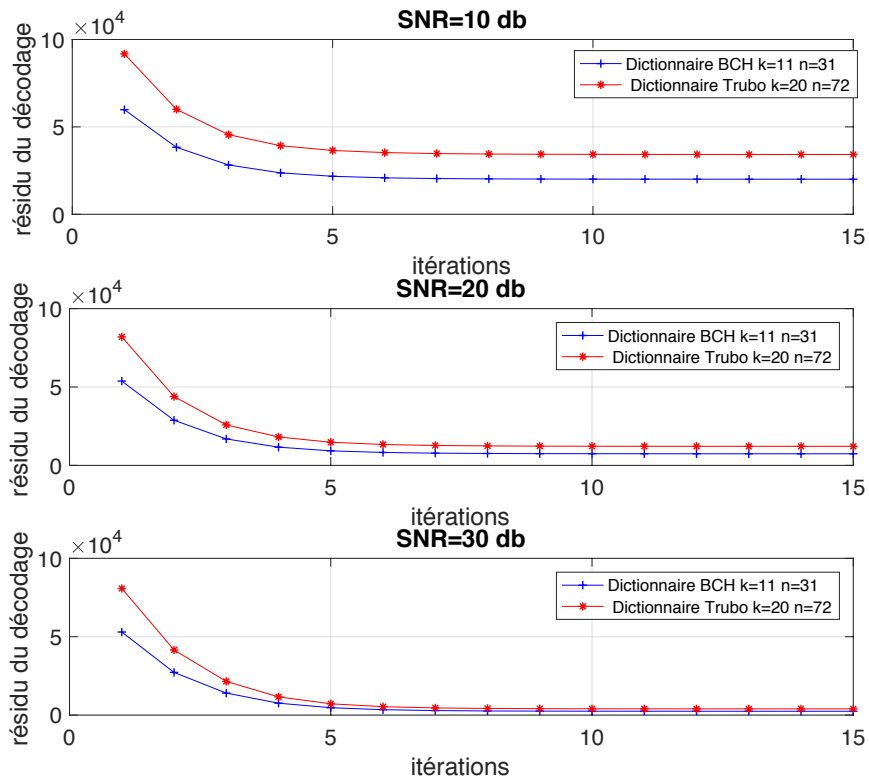


FIGURE IV.8 – Résidu du décodage du Matching pursuit : dictionnaire de Turbo Code  $\zeta(n = 72, k = 20, R = k/(3k + 12))$  et dictionnaire de codes BCH  $\zeta(n = 31, k = 11)$ .

## IV.3 Choix des dictionnaires de mots de code

### IV.3.1 Problématique

L'objectif de cette partie est de trouver les meilleurs dictionnaires de mots de code BCH capables de protéger le signal contre les erreurs introduites par le canal de transmission tout en gardant une erreur de quantification minimale. La difficulté dans le contexte de notre travail réside dans la représentation binaire de nos signaux. Les bits de poids faible sont très vulnérables au bruit du canal ; le taux d'erreur au niveau des bits de poids faible est très élevé, voir figure III.2.

Pour des codes BCH de longueur  $n = 31$ , la figure IV.10 représente, en fonction du nombre d'erreurs par mots avant décodage, le nombre moyen d'erreurs après décodage par

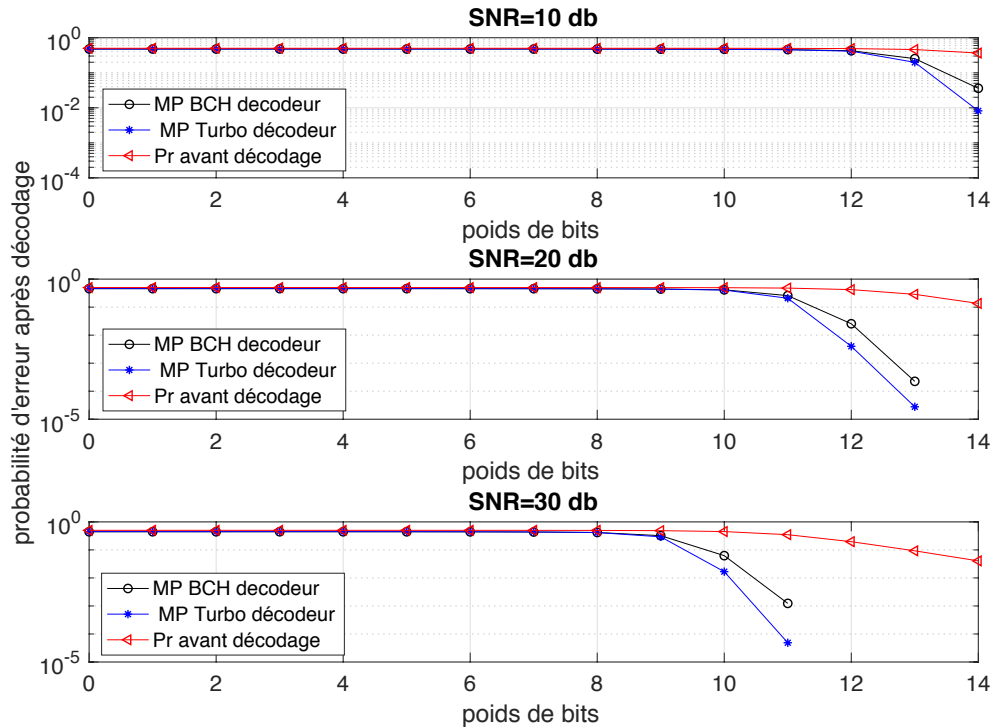


FIGURE IV.9 – Probabilité d’erreur du décodage par le Matching pursuit, dictionnaire de Turbo codes  $\zeta(n = 72, k = 20, R = k/(3k + 12))$  et dictionnaire de codes BCH  $\zeta(n = 31, k = 11)$ .

l’algorithme du matching pursuit modifié pour la quantification et le nombre moyen d’erreurs après décodage classique correcteur d’erreurs des codes BCH. On remarque sur cette figure que le pouvoir de correction des codeurs a augmenté en utilisant comme technique de décodage l’algorithme du matching pursuit pour la quantification. Prenons l’exemple du codeur de dimension  $k = 6$ , ce codeur est d’une capacité de correction de  $t = 7$ . En utilisant le dictionnaire de mots de code généré par ce codeur pour la quantification fondée sur l’algorithme du MPMQ (matching pursuit modifié pour la quantification) comme algorithme de décodage on remarque alors que le pouvoir de correction de ce codeur devient trois fois plus grand.

Par conséquent, le choix des dictionnaires de mots de code par poids de bit ne correspond plus à celui du choix des codeurs par poids de bit effectué dans le chapitre III. Dans les sections qui suivent on va choisir ces dictionnaires pour le cas particulier des signaux

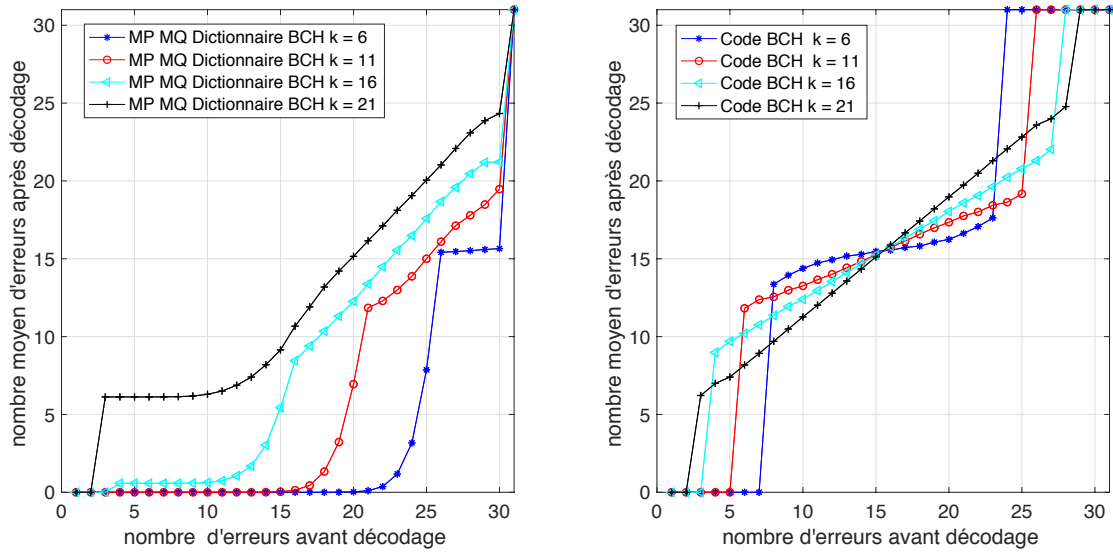


FIGURE IV.10 – Moyenne du nombre d’erreurs après décodage vs nombre d’erreurs avant décodage : à gauche décodage par l’algorithme du matching pursuit modifié pour la quantification utilisant des dictionnaires de codes BCH, à droite décodage correcteur d’erreur par les codes BCH de capacité de correction  $t = 2,3,5,7$ .

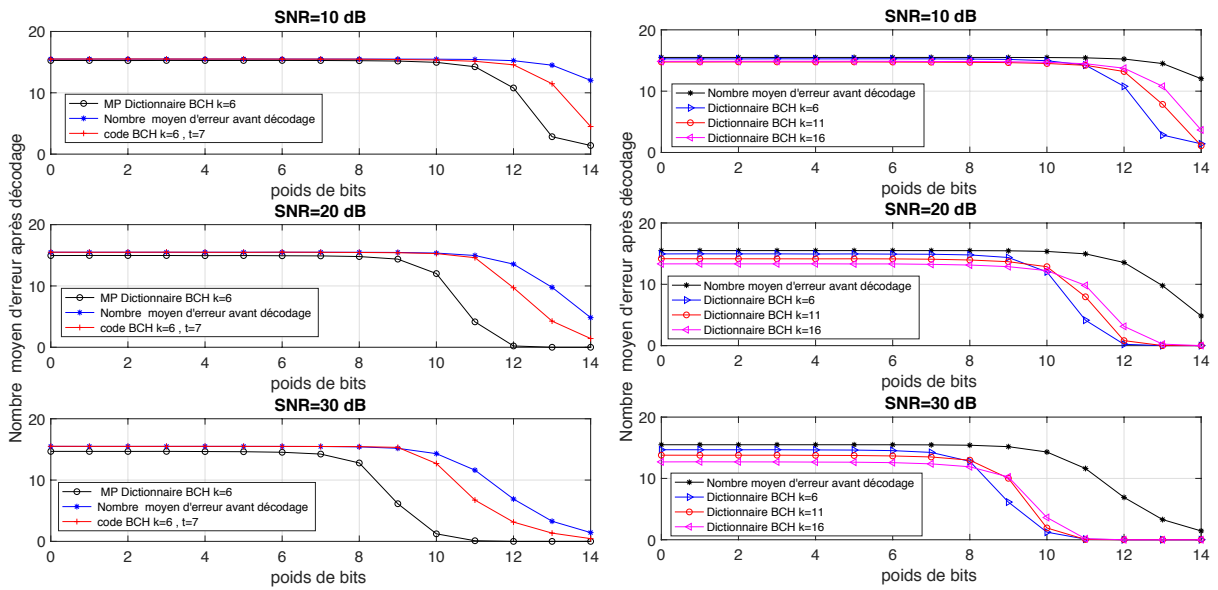


FIGURE IV.11 – Moyenne du nombre d’erreurs après décodage par poids de bit : à gauche décodage du matching pursuit vs décodage correcteur d’erreur, à droite décodage du matching pursuit utilisant plusieurs codes BCH de dimension  $k = 6,11,16$ .

à distribution uniforme et pour les signaux d'alarmes de voitures prioritaires.

### IV.3.2 Application aux signaux à distribution uniforme

Les simulations à la section précédente montrent que la nouvelle technique de quantification proposée dans ce chapitre augmente le pouvoir de correction des codeurs BCH. Ce comportement est aussi vérifié pour les codes convolutifs et les turbo codes. Pour déterminer quel codeur choisir par poids de bit afin de générer les dictionnaires de mots de code nécessaires pour la quantification, nous avons exploité la formule de la probabilité d'erreurs après décodage donnée au chapitre III à l'équation (III.21). Cette probabilité est une fonction de la probabilité  $P_e^i$  d'erreur par poids de bit avant décodage et du nombre moyen d'erreurs après décodage par poids de bit  $N_i^{decod}$ . La probabilité d'erreur binaire avant décodage reste valable car elle ne dépend pas du décodage. Par contre le nombre moyen d'erreurs après décodage par poids de bit ne correspond plus au nombre d'erreurs après décodage calculé au chapitre III.

L'équation (III.21) permet de calculer les probabilités  $P_{t_i}^{decod}$  d'erreur après décodage en fonction des capacités de correction  $t_i$ , pour chaque rapport signal à bruit et pour chaque poids de bit  $i$ . La figure IV.12 représente cette probabilité pour un rapport signal à bruit de 10 dB. Pour chaque rapport signal à bruit et pour chaque poids de bit, nous choisissons le codeur selon deux situations :

- s'il existe des capacités de correction telles que la probabilité d'erreur après décodage est inférieure à  $10^{-5}$  et inférieure à  $P_e^i$ , alors on choisit le codeur de capacité de correction  $t_i$  tel que :

$$t_i = \min\{t : P_t^{decod} \leq 10^{-5} \text{ et } P_t^{decod} < P_e^i\},$$

- sinon si  $\forall t, P_t^{decod} > 10^{-5}$ , alors on choisit le codeur de capacité de correction  $t_i$  minimisant la probabilité d'erreur après décodage.

Les capacités de correction correspondant aux codeurs choisis suivant les deux situa-



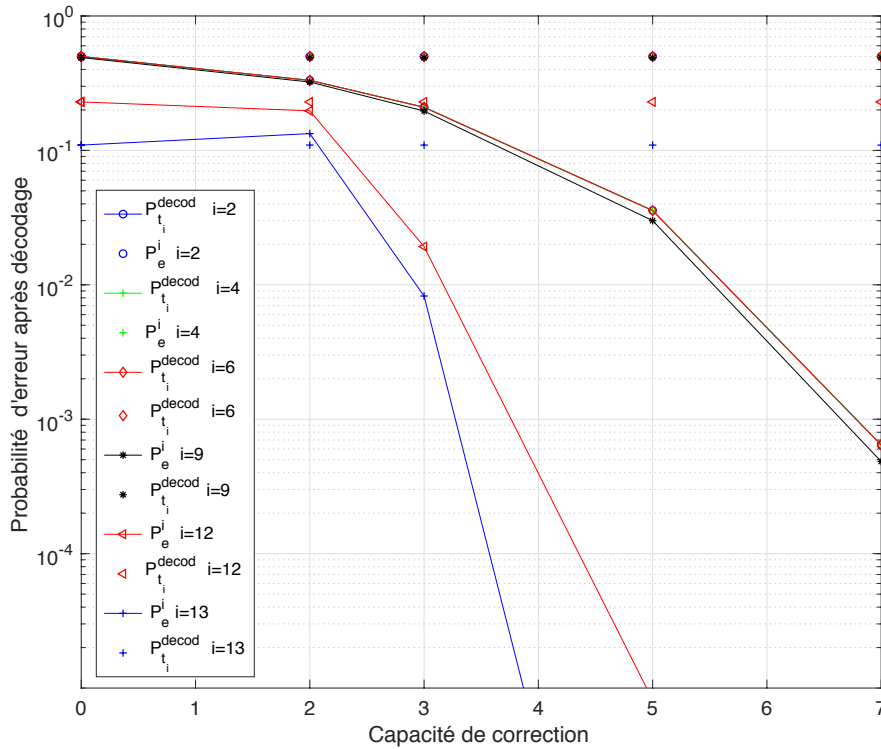


FIGURE IV.12 – Probabilité d’erreur après décodage par poids de bit : simulations effectuées sur un signal de distribution uniforme et pour un SNR=10 dB,  $i$  indice de poids de bit.

tions ci-dessus sont illustrées dans la figure IV.13.

## IV.4 Comparaison avec la méthode du chapitre III

Dans cette partie, nous comparons l’erreur de quantification avant transmission et après transmission introduite par la quantification fondée sur l’algorithme du matching modifié et celle introduite par la quantification fondée sur le décodage correcteur d’erreur proposée au chapitre III.

Des expériences ont été effectuées sur des signaux aléatoires à distribution uniforme dans  $A^n$ . Ces signaux ont été transmis via la chaîne de transmission de la figure III.6, le bruit de transmission dans ces expériences est un bruit blanc gaussien. Les résidus moyens par itération, l’erreur de quantification moyenne avant transmission (codage) et l’erreur

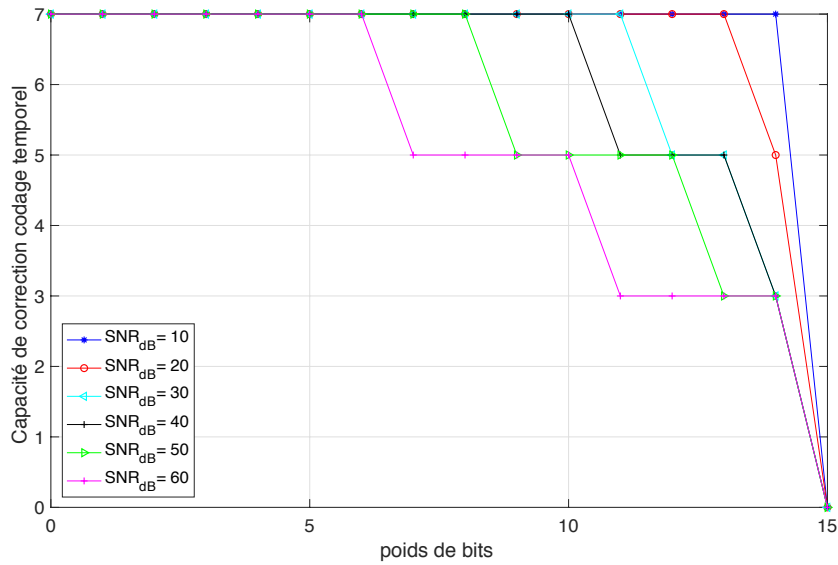


FIGURE IV.13 – Capacités de correction nécessaire pour corriger les erreurs introduites par un bruit blanc gaussien sur une trame de longueur  $n=31$  d'un signal de distribution uniforme.

de quantification après transmission (décodage) ont été calculés en fonction de plusieurs valeurs du rapport signal à bruit du canal de transmission. Ces expériences ont été effectuées pour les deux méthodes de quantification proposées : la quantification proposée dans ce chapitre et la quantification du chapitre III. Ces expériences ont été répétées sur  $10^5$  signaux de longueur  $n$ .

La notion du résidu par itération n'existe pas pour la quantification proposée au chapitre III, car celle-ci n'a pas d'ordre précis pour effectuer le codage. Le codage des bits de poids  $i$  est indépendant de celui des bits de poids  $j$ ,  $i \neq j$ . Cependant le codage effectué pour chaque poids de bits contribue à l'erreur de quantification finale. Ainsi, pour effectuer des comparaisons avec la méthode de ce chapitre on considère que cette quantification effectue un codage commençant par les bits de poids forts et se terminant aux bits de poids le plus faible. Le résidu par itération correspondant à cette quantification est calculé de la même façon que pour la quantification proposée dans ce chapitre, le résidu à l'itération  $l$  est  $R_l = R_{l+1} - 2^l C_l$  (ce dernier sert juste à effectuer des comparaisons et non au codage).

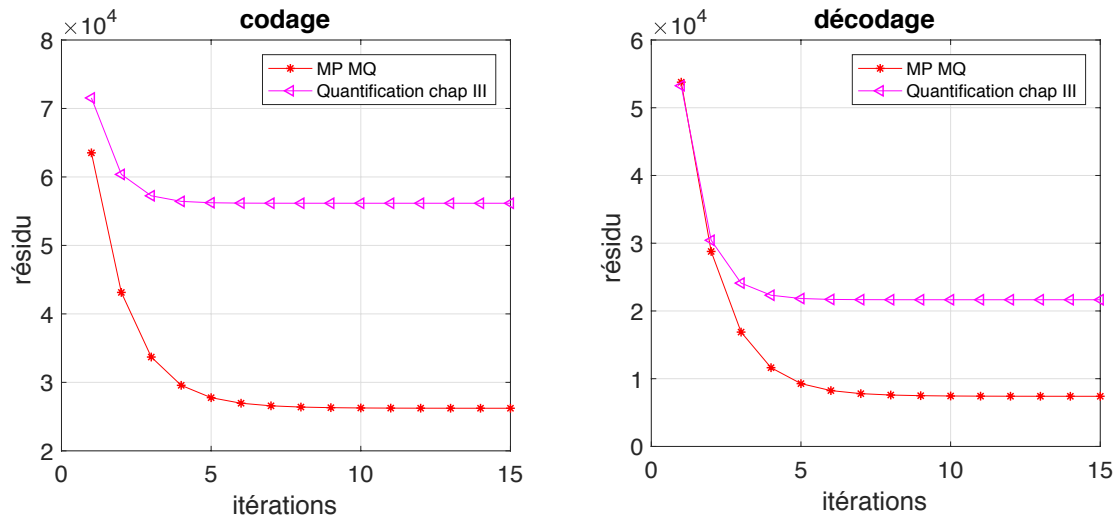


FIGURE IV.14 – Résidu du codage du Matching pursuit et du codage classique : dictionnaire de codes BCH et Codeurs BCH, simulations effectuées pour un rapport signal à bruit de 20 dB

La figure IV.14 représente les résidus par itération du codage et du décodage des deux méthodes de quantification proposées utilisant le même dictionnaire de mots de code BCH de dimension  $k = 11$  et de longueur  $n = 31$ . Le résidu à la dernière itération correspond à l'erreur de quantification pour une quantification au niveau de l'émetteur et correspond à l'erreur du décodage quand il s'agit d'une quantification au niveau du récepteur.

La quantification fondée sur l'algorithme du matching pursuit modifié pour la quantification permet de réduire à la fois l'erreur du décodage et l'erreur du codage d'un facteur 2 par rapport à la quantification proposée au chapitre III.

Comparons à présent la quantification proposée au chapitre III utilisant les codeurs choisis à la section III.2 du chapitre III et la quantification fondée sur l'algorithme du matching pursuit modifié utilisant les dictionnaires de mots de code choisis à la section IV.3.2 de ce chapitre.

La figure IV.15 représente le résidu du codage et du décodage des méthodes de quantification proposées pour un rapport signal à bruit de 30 dB. On remarque sur cette figure que l'algorithme du matching pursuit apporte une réduction de l'erreur de quantification

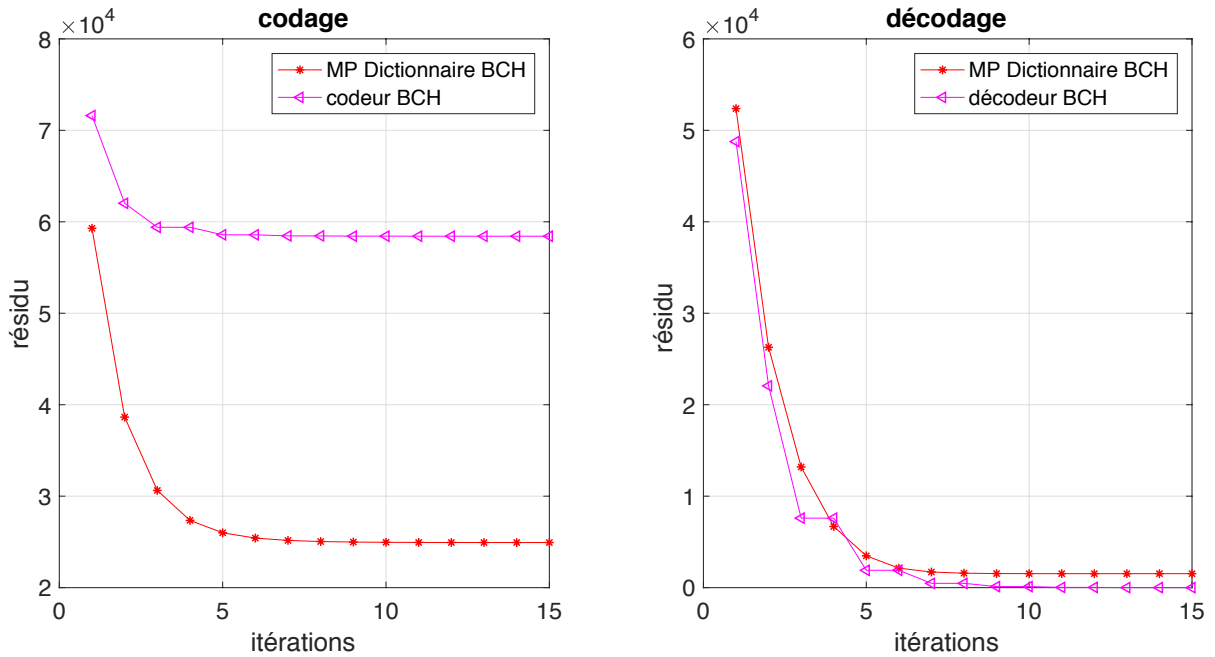


FIGURE IV.15 – Résidu du codage et décodage de la quantification fondée sur le Matching poursuit et du codage correcteur d’erreur : dictionnaire de codes BCH et Codeurs BCH, expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit gaussien, pour un rapport signal bruit de 30 dB

(le résidu à la dernière itération) par rapport à la quantification du chapitre III. Cependant on remarque sur cette même figure que le résidu du décodage à la dernière itération de la méthode de quantification du chapitre III est plus faible que celui de la méthode de quantification proposée dans ce chapitre.

Ceci ne provient pas d’une capacité de correction supérieure pour la méthode du chapitre III au contraire, comme l’illustre la figure IV.10 la quantification par le matching poursuit a une capacité de correction plus grande que celle de la méthode proposée au chapitre III. Mais pour cette dernière, le récepteur a une connaissance au préalable d’une grande partie du signal : la partie codée par le codeur de dimension 1 qui code avec le mot de code  $[0\dots0\dots0]$  les bits de poids pair et avec le mot de code  $[1\dots1\dots1]$  les bits de poids impair, laissant ainsi un taux d’erreurs égal à zéro au niveau de ces bits connus par le récepteur.

Lors du décodage, la quantification fondée sur l'algorithme du matching pursuit modifié n'a aucune information sur les signaux reçus, et par conséquent l'erreur du décodage de cette dernière est plus grande que celle de la méthode de quantification proposée au chapitre III.

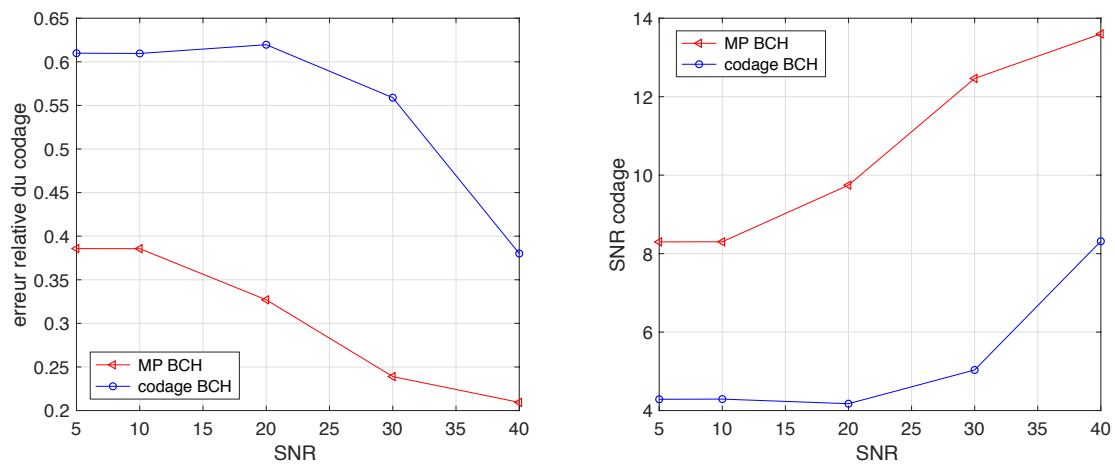


FIGURE IV.16 – Erreur relative du codage (à gauche) et SNR du codage (à droite) par le Matching pursuit et par le codage correcteur d'erreurs. Expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit blanc gaussien.

Les figures IV.16 et IV.17 représentent respectivement les rapports signal à bruit de quantification et l'erreur de quantification et les rapports signal à bruit du décodage et l'erreur du décodage des deux méthodes de quantification proposées ci-dessus en fonction du rapport signal à bruit du canal. On retrouve pour différents rapports signal à bruit les résultats vus précédemment pour un rapport signal à bruit de 30 dB.

## IV.5 Application aux signaux d'alarme

L'algorithme du matching pursuit modifié pour la quantification a été appliqué à des signaux d'alarme. Une étude a été effectuée sur ces signaux afin de déterminer les dictionnaires de mots de code par poids de bit qui seront adaptés à la correction des erreurs

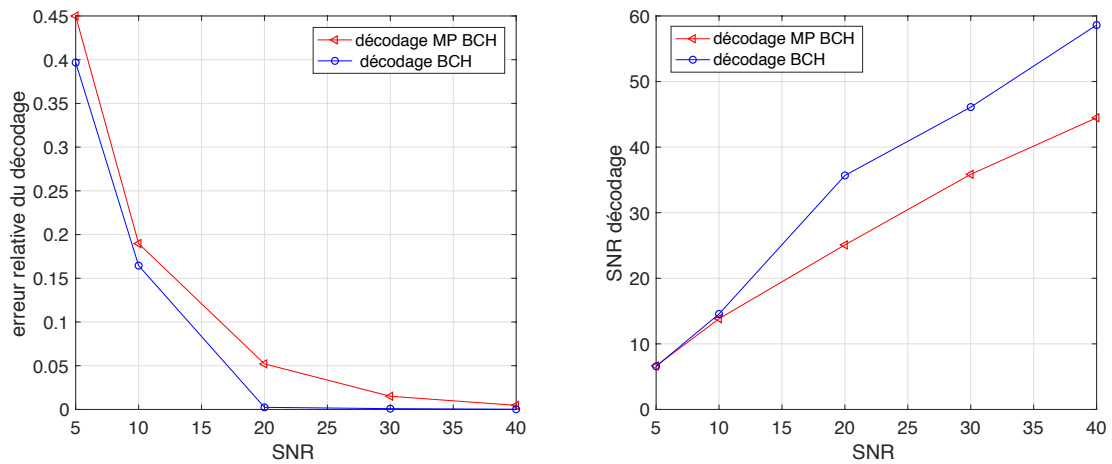


FIGURE IV.17 – Erreur relative du décodage (à gauche) et SNR du décodage (à droite) par le Matching pursuit et par le décodage correcteur d’erreurs. Expérience effectuée sur des signaux de distribution uniforme perturbés par un bruit blanc gaussien.

introduites par le bruit de trafic. Les signaux d’alarme ont été ainsi transmis via la chaîne de transmission illustrée à la figure III.6 où le bruit du canal correspond au bruit de trafic.

#### IV.5.1 Quelques caractéristiques des signaux d’alarme et du bruit de trafic

Les figures IV.18 et IV.19 représentent respectivement le spectre de la MDCT d’un signal d’alarme d’une voiture prioritaire et celui du bruit de trafic. La MDCT est calculée sur des fenêtres de longueur  $N = 2048$ . L’information de ces signaux est localisée dans les basses fréquences ; de même on remarque que le spectre du bruit de trafic est localisé dans les basses fréquences. Ainsi dans ce cas seuls les coefficients de la MDCT des basses fréquences (0 Hz à 1.5 kHz) sont codés (cette information est indiquée au récepteur ainsi pour le décodage le récepteur saura dans quelles bandes de fréquence il doit effectuer le décodage).

Le choix de la MDCT comme transformée temps-fréquence dans ce travail provient du fait qu’on a besoin des coefficients entiers qui peuvent être représentés en binaire.

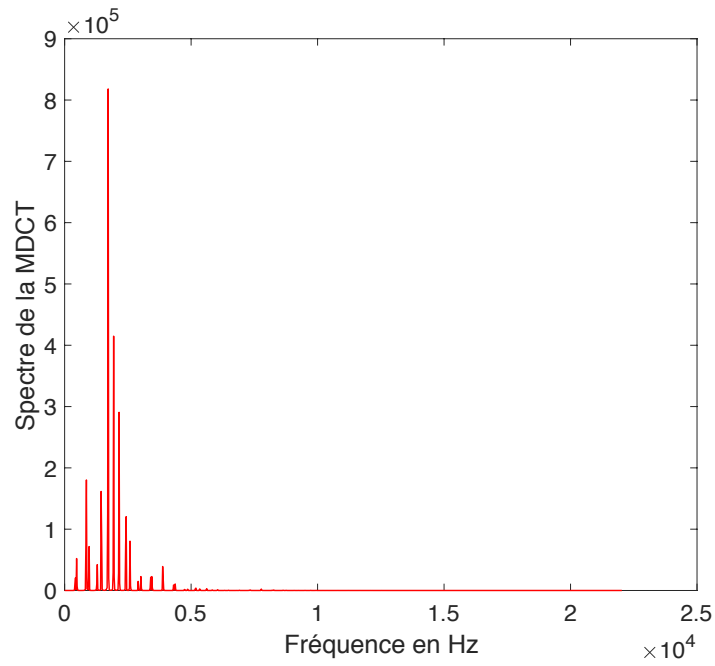


FIGURE IV.18 – Spectre de la MDCT d’un signal d’alarme de sapeurs-pompier

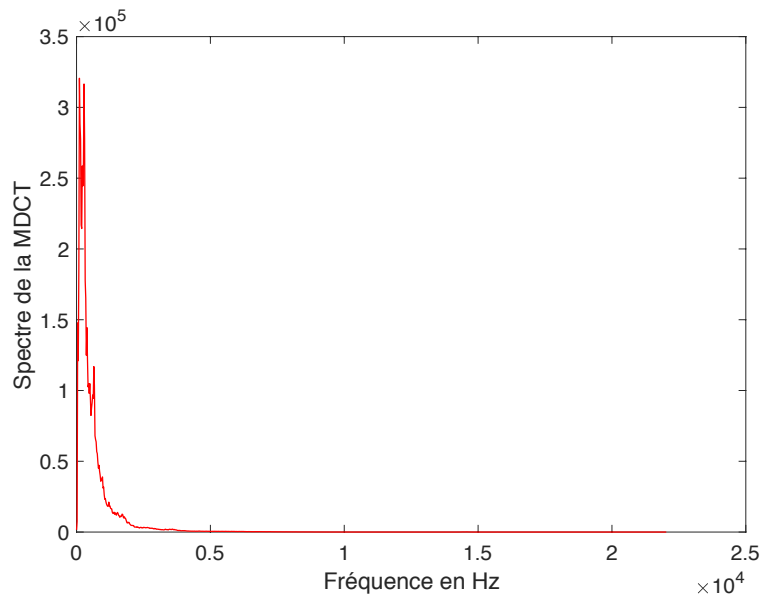


FIGURE IV.19 – Spectre de la MDCT du bruit de trafic

Toutefois, les coefficients de la MDCT sont des réels. Pour avoir des coefficients entiers un arrondi a été appliqué sur ces coefficients. L’inconvénient ici est que cet arrondi introduit

des erreurs à ces coefficients. Les coefficients temporels des signaux étant entiers, il existe une transformée temps-fréquence à valeur entière, la INTMDCT (integer MDCT) [20], [40]. L'inconvénient de cette dernière est qu'elle introduit une complexité très grande quand elle est effectuée sur de longues fenêtres du signal ( $N=2048$ ,  $N=1024$ ). Pour réduire la complexité du calcul de cette transformée, il faut prendre des fenêtres de petite taille, ce qui réduit la résolution fréquentielle du signal.

## IV.5.2 Probabilité d'erreurs introduites par le bruit de trafic

Nous reprenons la démarche précédente :

- détermination de la probabilité d'erreur par poids de bits  $i$  avant décodage, pour différents rapports signal à bruit du canal.
- calcul de la probabilité d'erreur après décodage par poids de bit, pour chaque rapport signal à bruit du canal.
- déduction des codeurs par poids de bit.

Ici, la première étape est réalisée par des simulations permettant de calculer la probabilité empirique d'erreurs par poids de bit relative au bruit de trafic (bruit de durée 44 s) dont l'amplitude a été normalisée pour avoir des rapports signal à bruit de -1 dB, 5 dB, 10 dB, 15 dB, 20 dB et 30 dB. Les probabilités d'erreurs par poids de bit avant décodage ont été calculées pour les échantillons temporels et pour les coefficients MDCT : voir figure IV.20. Notons que les probabilités d'erreurs sont plus faibles dans le domaine fréquentiel que dans le domaine temporel.

Les probabilités d'erreur par poids de bit après décodage sont calculées à partir des résultats de la figure IV.20 et des formules (III.21) et (III.4) du chapitre III.

Les courbes à la figure IV.21 représentent la probabilité d'erreur après décodage pour un SNR=10 dB et pour quelques poids de bit. À partir des courbes similaires à celles des figures IV.21 et IV.22 des capacités de correction par poids de bit ont été déterminées en fonction du rapport signal à bruit, voir la figure IV.23.



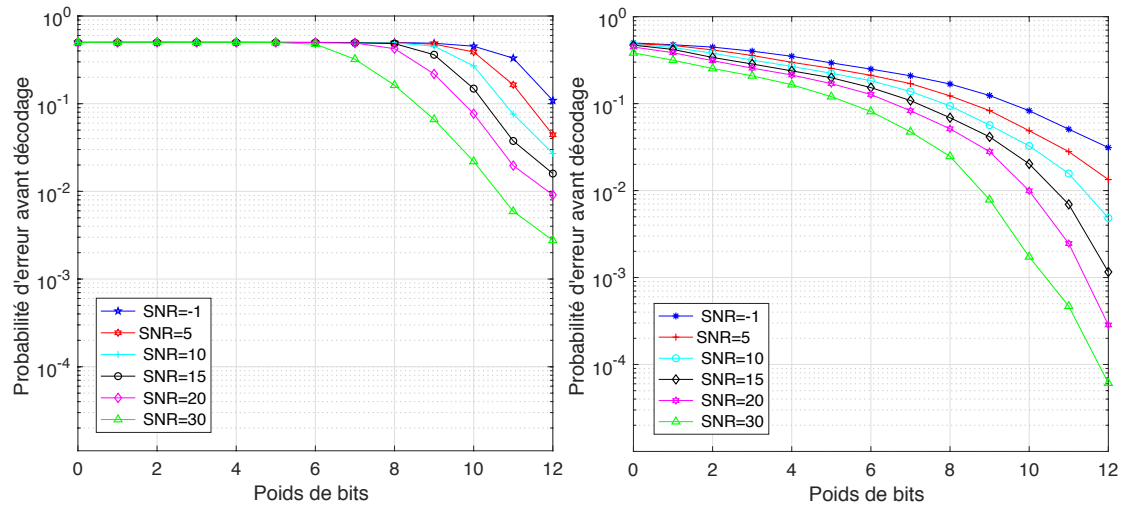


FIGURE IV.20 – Probabilité d’erreur par poids de bit : à gauche probabilité d’erreurs temporelles, à droite probabilité d’erreurs fréquentielles, simulations effectuées sur un signal de sapeurs-pompiers.

Ainsi pour chaque poids de bit, les codeurs correspondant à ces capacités de correction ont été choisis pour effectuer la quantification. Contrairement à la méthode de quantification du chapitre III, les bits de poids faible ne sont pas codés par le codeur de dimension 1 indiqué à la section III.2 du chapitre III, mais plutôt avec des dictionnaires de mots de code réduisant la probabilité d’erreur du décodage.

### IV.5.3 Application de l’algorithme du Matching Pursuit modifié pour la quantification des signaux d’alarme

Dans la section précédente, les courbes à la figure IV.23 nous ont permis de choisir les codeurs  $\zeta_i(n_i, k_i, t_i)$  en fonction du rapport signal à bruit. Ces codeurs sont utilisés pour quantifier les signaux d’alarme à l’aide de l’algorithme du matching pursuit pour la quantification. Dans cette section nous utilisons une quantification des signaux dans le domaine temporel, car d’une part c’est moins coûteux en complexité (moins d’opérations à effectuer), d’autre part pour les signaux d’alarme cette dernière ajoute une erreur de quantification non audible (même si l’erreur de quantification fréquentielle a été aussi réduite dans ce chapitre).

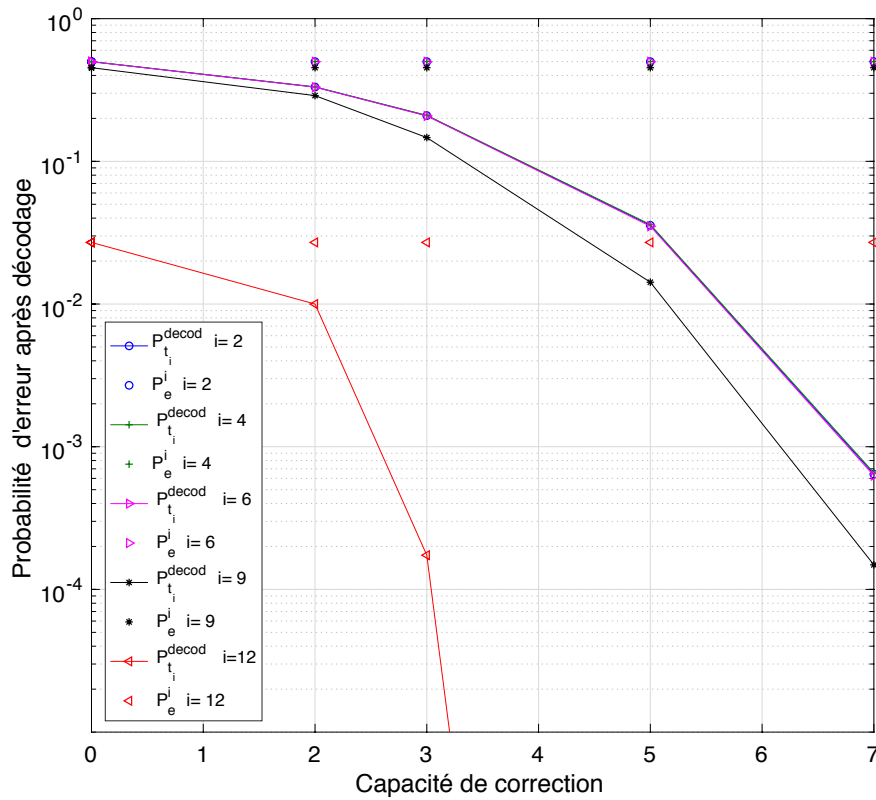


FIGURE IV.21 – Probabilité d’erreur après décodage par poids de bit : probabilité d’erreurs temporelles, simulations effectuées sur un signal de sapeurs-pompiers et pour un SNR=10 dB,  $i$  indice de poids de bit.

La figure IV.24 représente les résidus par itération du codage et du décodage par l’algorithme du matching pursuit modifié pour la quantification et les résidus par itération du codage et du décodage par la quantification proposée au chapitre III. La figure IV.25 représente l’erreur relative de quantification  $er_q$  donnée par l’équation (IV.10) et le rapport signal à bruit de quantification.

$$er_q = \frac{\|X - \hat{X}\|_2^2}{\|X\|_2^2} \quad (\text{IV.10})$$

La figure IV.26 représente l’erreur relative de quantification du décodage et le rapport signal à bruit du décodage par l’algorithme du matching pursuit et par la quantification proposée au chapitre III. Soit  $\tilde{X}$ , le signal décodé, alors l’erreur relative du décodage est

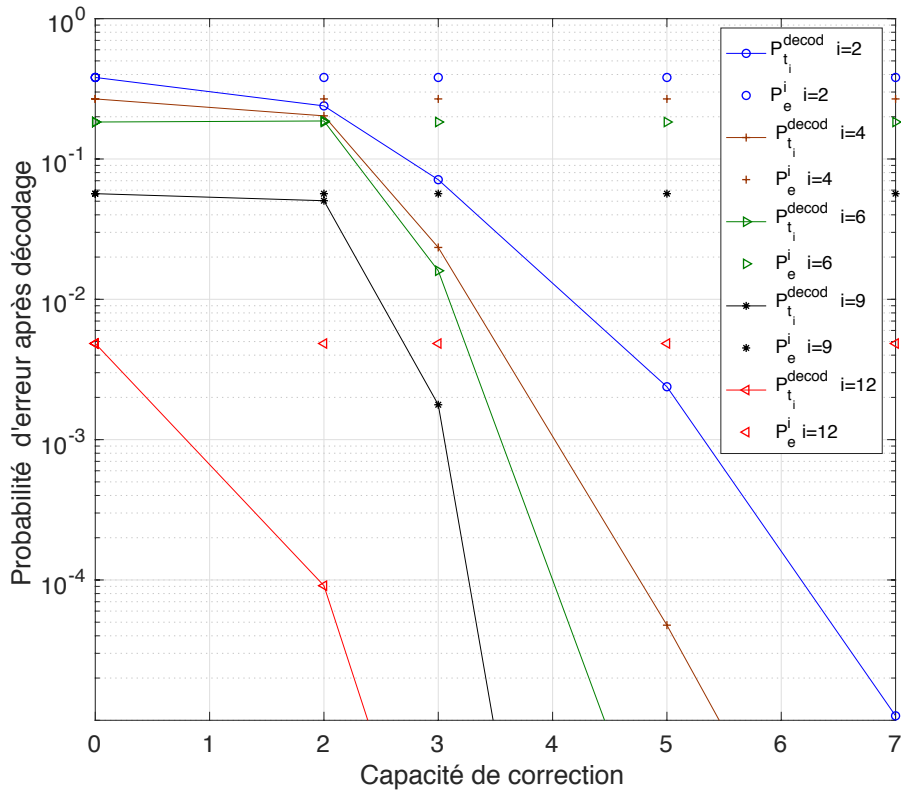


FIGURE IV.22 – Probabilité d’erreur après décodage par poids de bit : probabilité d’erreurs fréquentielles, simulations effectuées sur un signal de sapeurs-pompiers et pour un SNR=10 dB,  $i$  indice de poids de bit.

donnée par l’équation (IV.11).

$$er_d = \frac{\|\tilde{X} - \hat{X}\|_2^2}{\|\hat{X}\|_2^2} \quad (IV.11)$$

Les comparaisons entre les figures IV.16 et IV.17 et les figures IV.25 et IV.26 montrent que l’effet de la quantification est le même pour les signaux à distribution uniforme que pour les signaux d’alarme.

L’erreur de quantification introduite par l’algorithme du matching pursuit est plus faible que celle introduite par la quantification proposée au chapitre III. Cependant, l’erreur du décodage (quantification après transmission) de cette dernière est plus faible que celle de l’algorithme du matching pursuit (NB : pour le décodage binaire l’algorithme du

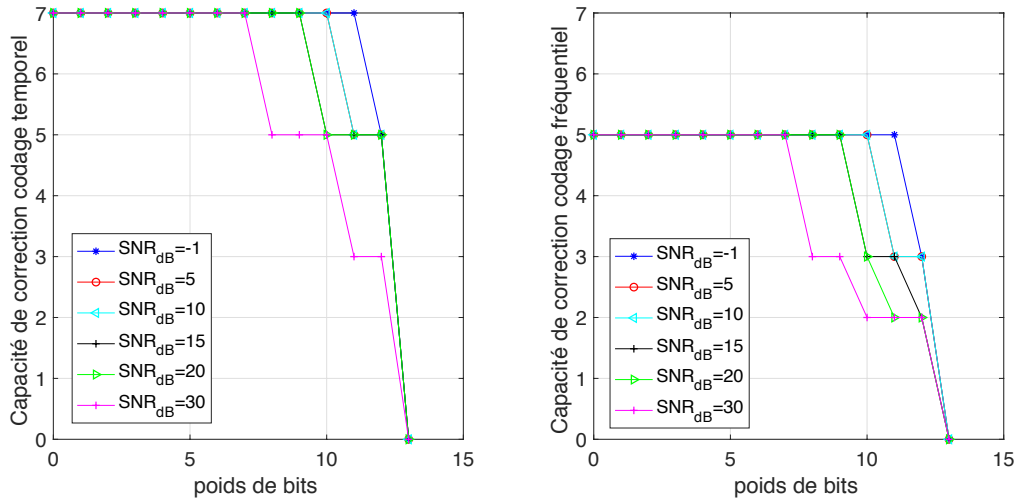


FIGURE IV.23 – Capacités de correction nécessaire pour corriger les erreurs du bruit introduites sur une trame de longueur  $n=31$  du signal de sapeurs-pompiers.

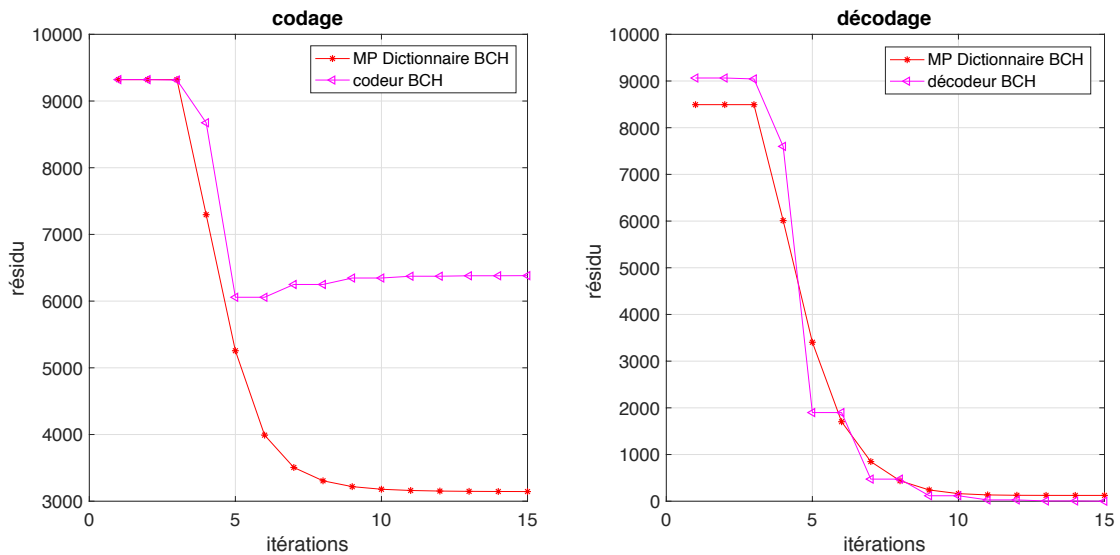


FIGURE IV.24 – Résidu du décodage du Matching pursuit et du codage classique : dictionnaire de codes BCH, expérience effectuée sur un signal d'ambulance, pour un rapport signal à bruit de 30 dB.

matching pursuit est plus robuste que celui du décodage du chapitre III).

La figure IV.27 représente l'erreur relative globale de quantification donnée par l'équation (IV.12) et le rapport signal signal à bruit correspondant à cette erreur. Cette figure

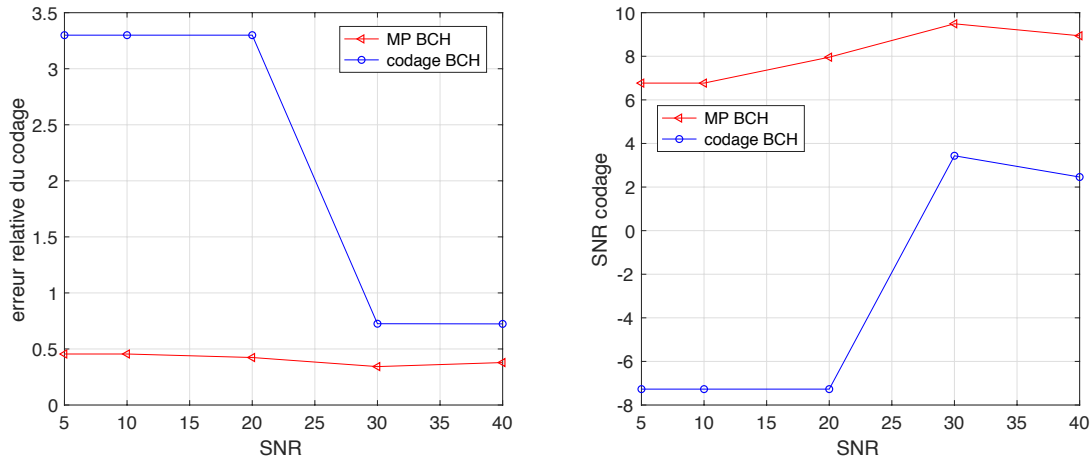


FIGURE IV.25 – SNR et erreur relative de quantification utilisant le codage de l’algorithme du Matching pursuit modifié et du codage classique : dictionnaire de codes BCH, expérience effectuée sur un signal d’ambulance.

montre que l’erreur globale du décodage par la quantification fondée sur le matching pursuit est plus faible que celle du décodage par la quantification introduite au chapitre III.

$$er_g = \frac{\|\tilde{X} - X\|_2^2}{\|X\|_2^2} \quad (IV.12)$$

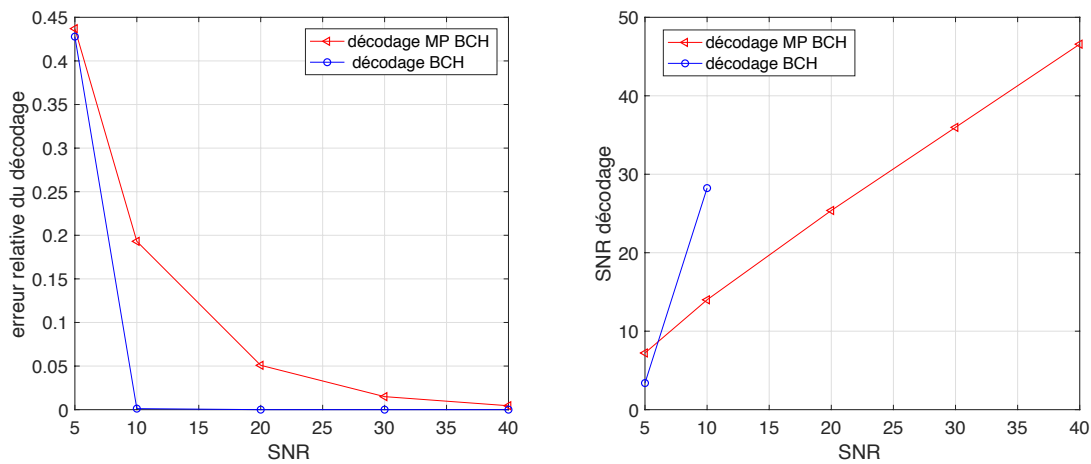


FIGURE IV.26 – SNR et erreur relative du décodage par le Matching pursuit et du décodage par le décodage correcteur d’erreur : dictionnaire de codes BCH. expérience effectuée sur un signal d’ambulance.

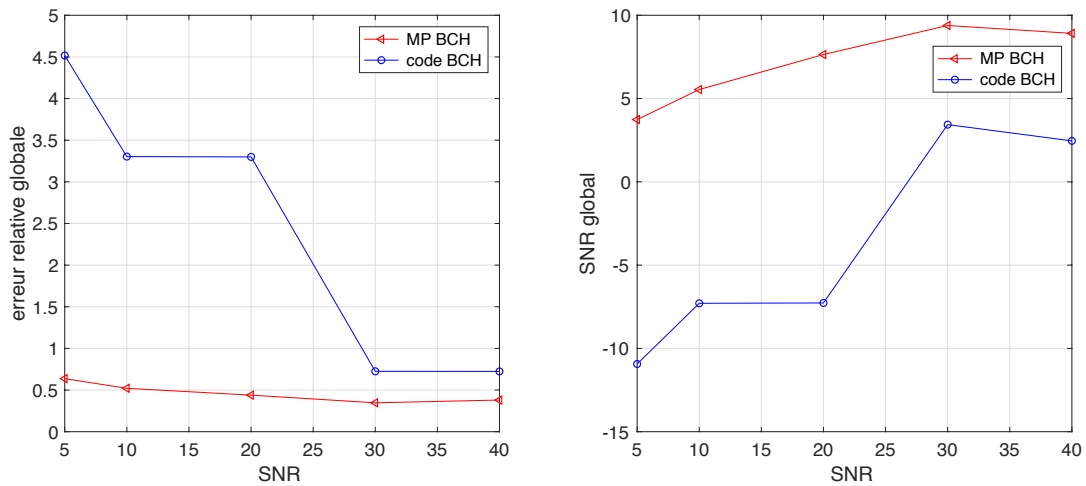


FIGURE IV.27 – SNR global et erreur relative globale du décodage par le Matching pursuit et du décodage par le décodage correcteur d’erreur : dictionnaire de codes BCH. expérience effectuée sur un signal d’ambulance.

## IV.6 Conclusion

Dans ce chapitre nous avons proposé une amélioration de la méthode de quantification proposée au chapitre III. L’algorithme du matching pursuit modifié pour la quantification réduit l’erreur de quantification tout en augmentant la capacité de correction des codeurs correcteurs d’erreur. Cet algorithme a été utilisé pour des dictionnaires de codes BCH, des dictionnaires de turbo codes et des dictionnaires de codes convolutifs. Quel que soit le type du codeur parmi les codeurs cités précédemment, cet algorithme assure une protection contre les erreurs de transmission plus grande que celle assurée par le décodage classique correcteur d’erreur correspondant à ces codeurs, pour un même codeur, voir la figure IV.10. Les résultats sur les codes convolutifs ont été vérifiés, toutefois ces derniers ne sont pas présentés dans la thèse.

L’artifice utilisé au chapitre III pour le codage des bits de poids faible n’ayant pas été utilisé, les performances du décodage s’avèrent inférieures, mais le compromis erreur de quantification / erreur du décodage est globalement meilleur (voir la figure IV.27).

Dans le chapitre qui suit nous proposons des méthodes de détection et d'identification des signaux fondées sur la quantification robuste proposée dans ce chapitre.

# Chapitre V

## Détection et identification des signaux

Dans cette partie nous exploitons les travaux sur la quantification proposée au chapitre précédent afin de réaliser la détection et l'identification des signaux audio. La détection et l'identification seront réalisées dans le domaine fréquentiel ou dans le domaine temporel, selon le domaine dans lequel la quantification a été effectuée.

Nous proposons deux méthodes de détection et d'identification des signaux, la première consiste à quantifier le signal avant la transmission (voir la chaîne à la figure V.1) et la seconde consiste à transmettre le signal non quantifié (voir la chaîne à la figure V.2). À la réception le signal reçu est quantifié (décodé) puis identifié. L'identification et la détection sont conjointes.

Dans ce chapitre, nous appliquons aussi aux signaux d'alarmes quantifiés des voitures prioritaires la méthode de détection et d'identification des signaux d'alarmes proposée au chapitre II. Les signaux d'alarmes seront ainsi quantifiés à l'aide de la quantification proposée au chapitre IV, puis détectés à l'aide de l'algorithme de détection des signaux d'alarme proposé à la section II.1.4 du chapitre II.



## V.1 Modèle de détection et identification des signaux dans le domaine temporel

Dans cette partie, on effectue la détection et l'identification des signaux dans le domaine temporel. Le processus de détection et d'identification est fondé sur la quantification dans le domaine temporel (codage temporel).

Deux situations sont envisagées, la première situation correspond à une détection et une identification dans le cas des signaux quantifiés avant transmission et à la réception. La seconde situation correspond à une détection et une identification des signaux transmis sans quantification, mais quantifiés après transmission au niveau du récepteur, voir les chaînes de transmission des figures V.1 et V.2, respectivement.

Les codeurs utilisés pour générer les dictionnaires  $D_i$  de la quantification sont des codeurs BCH de longueur  $n = 31$  et de dimension  $k_i = 6, 11, 16, 21, 0 \leq i \leq L - 1$ ,  $L$  la résolution binaire des signaux.

Pour effectuer la détection et l'identification des signaux, on ne s'intéresse pas à retrouver le signal entièrement, mais à récupérer l'information contenue dans les bits de poids fort, qui sera extraite via la quantification.

Supposons qu'on soit dans le cas de la chaîne de transmission indiquée à la figure V.1.

On note :

- $x$  le signal original.
- $\hat{x}$  le signal quantifié.
- $\hat{x}_b$  le signal bruité par le canal de transmission.
- $\tilde{x}$  le signal décodé après transmission.

Les signaux  $x, \hat{x}_b, \hat{x}$  et  $\tilde{x}$  sont liés par les relations suivantes :

1.  $\hat{x} = x + e_q$

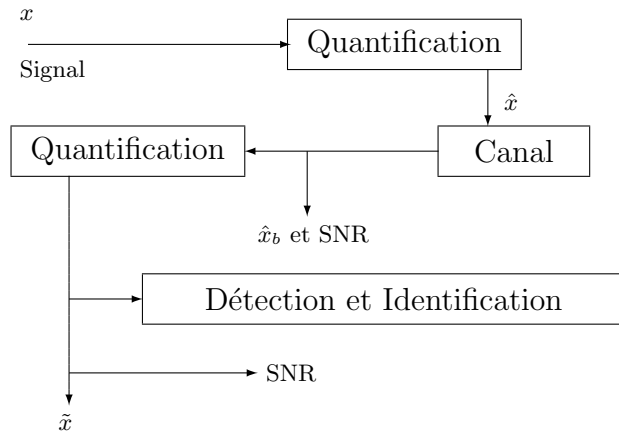


FIGURE V.1 – Chaîne de transmission : cas des signaux quantifiés avant transmission (domaine temporel).

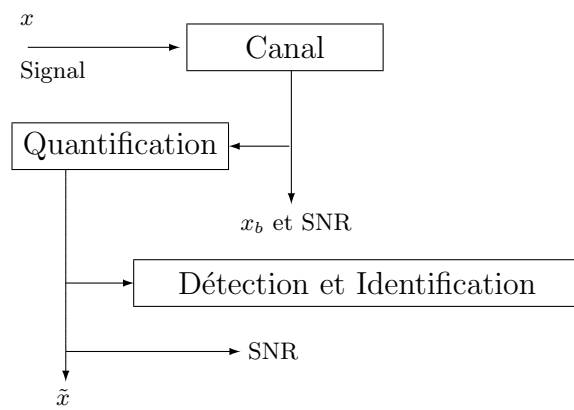


FIGURE V.2 – Chaîne de transmission : cas des signaux transmis sans quantification (domaine temporel).

2.  $\hat{x}_b = \hat{x} + e_b$

3.  $\tilde{x} = \hat{x} + e_d$

où  $e_q$ ,  $e_b$  et  $e_d$  sont respectivement les erreurs introduites par le bruit de quantification (codage), le bruit du canal de transmission et la combinaison canal plus décodage (quantification au niveau du récepteur).

Le signal  $\tilde{x}$  doit être le plus proche possible de  $\hat{x}$ .

Supposons maintenant qu'on soit dans le cas de la chaîne de transmission indiquée à la figure V.2, alors le point 2 indiqué ci-dessus change pour devenir :

2.  $x_b = x + e_b = \hat{x} - e_q + e_b$ ,

où  $x_b$  est le signal  $x$  bruité par le canal de transmission. À l'encontre de la situation précédente où on a juste  $e_b$  à corriger pour retrouver le signal  $\tilde{x}$  le plus proche de  $\hat{x}$ , pour cette situation il faut corriger les erreurs binaires introduites par l'erreur euclidienne  $-e_q + e_b$ .

Comme il s'agit d'une transmission sans quantification, alors à la réception pour retrouver le signal  $\tilde{x}$  le plus proche de  $\hat{x}$ , on peut utiliser les dictionnaires de mots de code générés par les codeurs ayant la plus grande capacité de correction, sachant que ces dictionnaires sont utilisés pour générer  $\hat{x}$ . Il s'agit dans ce travail des codeurs BCH de la plus petite dimension et/ou des turbo codes de rendement  $R = 1/3$ .

## V.2 Modèle de détection et identification des signaux dans le domaine fréquentiel

Les signaux sont quantifiés dans le domaine fréquentiel, soit à l'émission et en réception (voir la figure V.3), soit en réception seulement (voir la figure V.4). La détection et l'identification des signaux seront effectuées dans le domaine fréquentiel.

La transformée temps-fréquence utilisée est la MDCT appliquée sur des fenêtres recouvrantes de taille  $N = 2048$  échantillons du signal. Pour limiter la complexité, au lieu d'utiliser la InTMDCT, un simple arrondi est effectué sur les coefficients de la MDCT afin d'avoir des coefficients entiers représentables en binaire suivant la décomposition indiquée par le tableau III.1.

Les codeurs utilisés pour générer les dictionnaires  $D_i$  de la quantification sont identiques à ceux de la partie précédente.

Supposons qu'on soit dans la situation où les signaux sont quantifiés avant transmission (voir la figure V.3).

On note :

- $\chi$  la MDCT du signal original  $x$ .
- $\hat{\chi}$  la MDCT du signal quantifié  $\hat{x}$ .
- $\hat{\chi}_b$  la MDCT du signal bruité  $\hat{x}_b$ .
- $\tilde{\chi}$  la MDCT du signal décodé après transmission  $\tilde{x}$ .

On a alors les relations suivantes entre les signaux  $\chi$ ,  $\hat{\chi}_b$ ,  $\hat{\chi}$  et  $\tilde{\chi}$  :

1.  $\hat{\chi} = \chi + e_q$
2.  $\hat{\chi}_b = \hat{\chi} + e_b$
3.  $\tilde{\chi} = \hat{\chi} + e_d$

où  $e_q$ ,  $e_b$  et  $e_d$  sont respectivement les erreurs introduites par le bruit de quantification (codage), le bruit du canal de transmission et la combinaison canal plus décodage (quantification au niveau du récepteur). Le signal  $\tilde{\chi}$  doit être le plus proche possible de  $\hat{\chi}$ .

Supposons maintenant qu'on soit dans le cas de la chaîne de transmission illustrée à la figure V.4, alors la relation 2 définie précédemment change pour devenir :

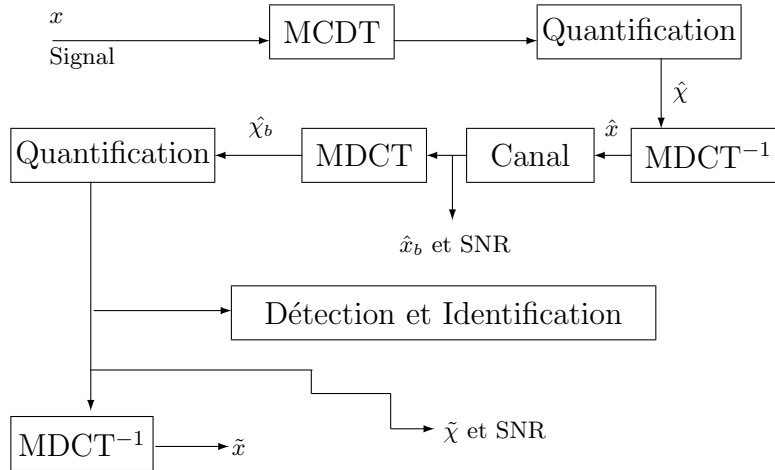


FIGURE V.3 – Chaîne de transmission : cas des signaux quantifiés avant transmission (domaine fréquentiel).

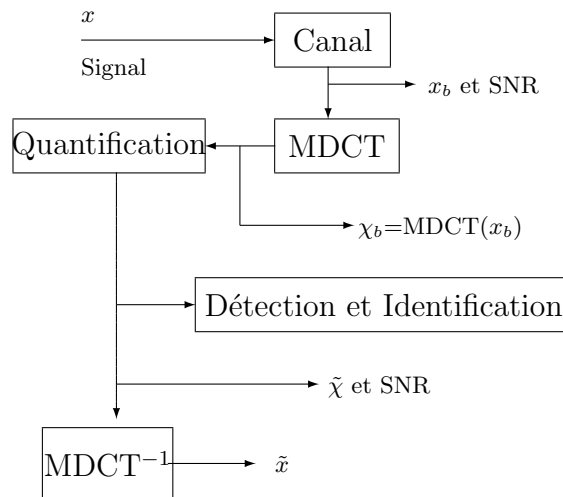


FIGURE V.4 – Chaîne de transmission : cas des signaux transmis sans quantification (domaine fréquentiel).

$$2. \hat{\chi}_b = \chi + e_b = \hat{\chi} - e_q + e_b,$$

où  $\hat{\chi}_b$  est le signal  $\hat{\chi}$  bruité par le canal de transmission. Ainsi à la réception, pour retrouver le signal  $\tilde{\chi}$  le plus proche de  $\hat{\chi}$ , il faut corriger les erreurs binaires introduites par l'erreur euclidienne  $-e_q + e_b$  de telle sorte que l'erreur du décodage soit très petite ou nulle.

De même que la méthode temporelle, comme il s'agit d'une transmission sans quantification, pour retrouver le signal  $\tilde{\chi}$  le plus proche de  $\hat{\chi}$ , le récepteur utilise des dictionnaires de mots de code générés par les codeurs ayant la plus grande capacité de correction, sachant que ces dictionnaires sont utilisés pour générer  $\hat{\chi}$ .

Ces deux méthodes (détection des signaux transmis sans quantification et détection des signaux quantifiés avant transmission) présentent des avantages et des inconvénients. Quand il s'agit d'un signal audio, il est conseillé d'utiliser la chaîne de transmission introduite à la figure V.4, car si l'on utilise la chaîne introduite à la figure V.3 pour les signaux audio l'erreur de quantification (codage) peut être audible, de même dans le cas d'application à l'identification des personnes, où il n'est pas possible d'intervenir sur les signaux avant transmission, la chaîne de transmission sera par défaut cette chaîne. En utilisant cette chaîne dans le cas des signaux audio nous n'avons plus à nous soucier de l'audibilité de l'erreur de quantification, car les signaux sont transmis sans être modifiés. La méthode de détection des signaux quantifiés avant transmission (chaîne V.3) peut être utilisée pour tous les autres types de signaux.

### V.3 Processus de détection et d'identification

Le processus de détection et d'identification est fondé sur un vecteur formé par les mots de code qui se trouvent au niveau des poids de bit corrigibles du signal quantifié, dont la version de référence est supposée connue du récepteur.

On a déjà vu que les erreurs introduites par le bruit de transmission deviennent corrigibles à partir d'un certain poids de bit. Ce poids de bit varie en fonction du rapport signal à bruit du canal de transmission, soit  $p$  ce poids. Le signal  $x$  est découpé en  $m$  trames de longueur  $n$ . Chaque trame codée  $\hat{x}_j = [\hat{s}_{j1}, \hat{s}_{j2}, \dots, \hat{s}_{ji}, \dots, \hat{s}_{jn}]$  du signal est représentée en binaire sous forme d'une matrice binaire dont les lignes correspondent à des suites de mots de code,  $0 \leq j \leq m-1$  (voir le tableau V.1). La restriction de cette matrice

aux bits de poids allant de  $p$  à  $L - 1$  est une matrice  $((L - p) \times n)$  dont les lignes sont les suites de mots de code  $C_{ji}$ ,  $C_{ji} \in D_i$ ,  $p \leq i \leq L - 1$ .

...	$\hat{s}_{j1}$	$\hat{s}_{j2}$	.	.	.	.	.	$\hat{s}_{jn}$	...
$2^0$	$c_{01}$	$c_{02}$	.	.	.	.	.	$c_{0n}$	$C_{j0} \in D_0$
$2^1$	$c_{11}$	$c_{12}$	.	.	.	.	.	$c_{1n}$	$C_{j1} \in D_1$
.	.	.	.	.	.	.	.	.	.
$2^p$	$c_{p1}$	$c_{p2}$	.	.	.	.	.	$c_{pn}$	$C_{jp} \in D_p$
.	.	.	.	.	.	.	.	.	.
$2^{L-1}$	$c_{(L-1)1}$	$c_{(L-1)2}$	.	.	.	.	.	$c_{(L-1)n}$	$C_{j(L-1)} \in D_{L-1}$
signe $X_{L1}$	$X_{L2}$	.	.	.	.	.	$X_{Ln}$	$X_L$	

TABLE V.1 – Représentation binaire d'une trame codée d'un signal quantifié

Soit  $V_j$  un vecteur dans  $A^n$  obtenu par combinaison linéaire des lignes de la matrice formée par les  $L - p$  mots de code de la zone corrigible,  $V_j$  est donné par :

$$V_j = \sum_{i=p}^{L-1} 2^i C_{ji}. \quad (\text{V.1})$$

Ainsi sur l'ensemble des trames constituant le signal  $x$  nous obtenons un vecteur  $V$  formé d'une succession de vecteurs  $V_j$ .

$$V = [V_1 \dots V_j \dots V_m].$$

Après transmission du signal on récupère le signal décodé  $\tilde{x}$ , ensuite pour chaque trame  $\tilde{x}_j$  on calcule le vecteur  $\tilde{V}_j$ , où  $\tilde{V}_j$  est donné par :

$$\tilde{V}_j = \sum_{i=p}^{L-1} 2^i \tilde{C}_{ji}. \quad (\text{V.2})$$

Ensuite le vecteur  $\tilde{V}$  est constitué,  $\tilde{V}$  est donné par :

$$\tilde{V} = [\tilde{V}_1 \dots \tilde{V}_j \dots \tilde{V}_m].$$

La présence du nombre maximal de vecteurs  $V_j$  aux positions  $j$  dans le vecteur  $\tilde{V}$  indique la présence du signal  $x$ , ainsi la détection et l'identification sont effectuées.

Nous proposons deux approches de détection et d'identification des signaux. La première approche est présentée dans l'algorithme 3, où le signal bruité  $x_b$  (ou  $\hat{x}_b$  pour les signaux quantifiés avant transmission) est quantifié au niveau du récepteur. Ensuite le vecteur  $\tilde{V}$  est récupéré. Si au long du vecteur  $\tilde{V}$  le nombre de vecteurs  $V_j$  détectés aux positions  $j$  dans  $\tilde{V}$  est inférieur à *seuil*, alors le signal est considéré comme absent sinon le signal est considéré comme présent.

La règle de décision peut s'écrire :

$$\Lambda_0 = \{\tilde{V}_j, \tilde{V}_j = V_j, j = 1, \dots, m - 1\} : \eta_0 = \text{card } \Lambda_0 > \textit{seuil} \Rightarrow \text{détection} \quad (\text{V.3})$$

La règle de décision V.3 peut être relaxée en tolérant une petite erreur au lieu de l'égalité stricte entre  $\tilde{V}_j$  et  $V_j$  :

$$\Lambda_\epsilon = \{\tilde{V}_j, \|\tilde{V}_j - V_j\|_1 \leq \epsilon, \epsilon > 0\} : \eta_\epsilon = \text{card } \Lambda_\epsilon > \textit{seuil} \Rightarrow \text{détection} \quad (\text{V.4})$$

Les valeurs optimales de  $\epsilon$  et *seuil* pour effectuer la détection et l'identification des signaux seront déterminées à partir des courbes COR.



---

**Algorithme 3** : Méthode 1 de détection et d'identification

---

**Entrée** :  $x_b$  (ou  $\hat{x}_b$ ) le signal bruité,  $D_i$ ,  $p$ ,  $V$ ,  $\epsilon$  et  $L$ .

```
1 Décoder le signal  $x_b$  (ou  $\hat{x}_b$ )  $\rightarrow \tilde{x}$ ;  
2 Récupérer la représentation binaire du signal décodé  $\tilde{x}$ ;  
3  $\eta_\epsilon = 0$ ;  
4 for  $j \leftarrow 1$  to  $m$  do  
5    $\tilde{V}_j = \sum_{i=p}^{L-1} 2^i \tilde{C}_{ji}$ ;  
6   if ( $\|\tilde{V}_j - V_j\|_1 \leq \epsilon$ ) then  
7      $\eta_\epsilon = \eta_\epsilon + 1$ ;  
8   end  
9 end  
10  $\tilde{V} = [\tilde{V}_1 \dots \tilde{V}_j \dots \tilde{V}_m]$ ;  
11 if ( $\eta_\epsilon < \text{seuil}$ ) then  
12   Signal absent;  
13 else  
14   Signal présent, l'identité est donnée par le vecteur  $V$  ;  
15 end
```

**Sortie** : Présence ou absence du signal

---

Dans la première approche (algorithme 3), la détection et l'identification sont fondées sur le nombre  $\eta_\epsilon$  de vecteurs  $\tilde{V}_j$  tel que  $\|\tilde{V}_j - V_j\|_1 \leq \epsilon$ . Pour indiquer la présence ou l'absence du signal cherché, ce nombre est comparé à un seuil, si le nombre  $\eta_\epsilon$  est supérieur ou égal au seuil alors le signal est considéré comme présent sinon le signal est considéré comme absent.

Pour la seconde approche représentée par l'algorithme 4, une contrainte de succession des  $\tilde{V}_j$  a été ajoutée. Posons  $W$  la concaténation des  $\eta_\epsilon$  vecteurs  $V_j$  successifs et  $\tilde{W}$  le vecteur formé par la concaténation des  $\eta_\epsilon$  vecteurs  $\tilde{V}_j$  successifs,  $\tilde{W}$  et  $W$  sont donnés par :

$$W = [V_1 \dots V_j \dots V_{\eta_\epsilon}].$$

et,

$$\tilde{W} = [\tilde{V}_1 \dots \tilde{V}_j \dots \tilde{V}_{\eta_\epsilon}].$$

Ainsi le signal est conjointement détecté et identifié si le nombre  $\eta_\epsilon$  de vecteur  $\tilde{V}_j$  successifs est supérieur ou égal à *seuil* et  $\tilde{W}$  et  $W$  vérifient :

$$\|\tilde{W} - W\|_1 = 0.$$

Pour des raisons de simplicité et pour réduire la complexité on se limite au nombre  $\eta_\epsilon = \textit{seuil}$ , car l'égalité uniquement est suffisante pour détecter la présence du signal. Pour rendre la contrainte de détection et d'identification des signaux moins contraignante on autorise une petite erreur inférieure à un certain  $\epsilon$ ,  $\epsilon \geq 0$ . La présence du signal est ainsi indiquée si les inégalités suivantes sont vérifiées :

$$\|\tilde{W} - W\|_1 \leq \epsilon \text{ et } \eta_\epsilon = \textit{seuil}$$

---

**Algorithme 4 :** Méthode 2 de détection et d'identification

---

**Entrée :**  $x_b$  (ou  $\hat{x}_b$ ) le signal bruité,  $D_i$ ,  $p$ ,  $V$ ,  $\epsilon$  et  $L$ , *seuil*.

- 1 Décoder le signal  $x_b$  (ou  $\hat{x}_b$ )  $\rightarrow \tilde{x}$ ;
- 2 Récupérer la représentation binaire du signal décodé  $\tilde{x}$ ;
- 3 **for**  $j \leftarrow 1$  **to**  $m$  **do**
- 4      $\tilde{V}_j = \sum_{i=p}^{L-1} 2^i \tilde{C}_{ji}$ ;
- 5 **end**
- 6  $\tilde{W} = [\tilde{V}_1 \dots \tilde{V}_j \dots \tilde{V}_{\textit{seuil}}]$ ;
- 7 **if**  $\|\tilde{W} - W\|_1 \leq \epsilon$  **then**
- 8     Signal présent , l'identité est donnée par le vecteur  $V$  ;
- 9 **else**
- 10     Signal absent;
- 11 **end**

**Sortie :** Présence ou absence du signal

---

L'algorithme 3 présente un retard de détection par rapport à l'algorithme 4 ; ce dernier

a uniquement besoin d'une partie du signal de longueur *seuil* du signal pour effectuer la détection et l'identification, alors que le premier algorithme a besoin d'une grande partie du signal pour effectuer la détection et l'identification.

La structure des dictionnaires utilisés pour la quantification nous permet de remplacer chaque vecteur  $V_j$  de longueur  $n$  par un vecteur d'indices  $I_j$  de longueur  $L - p$ . Ce vecteur contient les indices des  $L - p$  mots de code permettant de constituer le vecteur  $V_j$ , ce qui est plus économique en mémoire.

Dans le cas d'utilisation de la contrainte d'égalité entre les  $V_j$  et les  $\tilde{V}_j$ , on n'aura pas besoin de reconstruire les vecteurs  $V_j$  pour effectuer la détection et l'identification des signaux. Les comparaisons seront effectuées directement sur les vecteurs  $I_j$  et  $\tilde{I}_j$ , où  $\tilde{I}_j$  est le vecteur d'indice des mots de code constituant le vecteur  $\tilde{V}_j$ .

## V.4 Application à la détection et l'identification des signaux d'alarme

Dans cette partie nous appliquons les algorithmes proposés à la détection et l'identification des signaux d'alarme des voitures prioritaires. Dans un premier temps l'étude du comportement du bruit de trafic sur la représentation binaire de ces signaux illustrée à la section IV.5.2 du chapitre IV est exploitée afin de déterminer les paramètres  $V_j$  et  $p$  nécessaires à la détection et l'identification de ces signaux.

Les probabilités d'erreur par poids de bit avant et après décodage sont exploitées pour déterminer pour chaque rapport signal à bruit la valeur du paramètre  $p$ .

Prenons l'exemple d'un signal d'alarme de sapeurs-pompiers. Les courbes de probabilités d'erreur après décodage représentées à la figure IV.9 du chapitre IV montrent qu'à partir d'un certain poids de bit (bits de poids fort) les erreurs introduites par le canal de transmission deviennent corrigibles. La valeur de ce poids de bit correspond au

paramètre  $p$  à partir duquel la détection et l'identification des signaux sera effectuée. Ce poids de bit varie en fonction du rapport signal à bruit du canal de transmission.

Toutefois, dans la suite, nous considérons différentes valeurs de  $p$  autour de celles données par ces courbes de probabilité d'erreur, et la valeur optimale de  $p$  sera déterminée par les courbes COR.

Pour les signaux transmis sans quantification, le nombre  $L - p$  des poids de bit corrigibles peut devenir plus faible ou plus grand que celui des signaux quantifiés avant transmission. Ceci provient du fait que l'erreur de quantification qui s'ajoute à l'erreur du bruit du canal de transmission peut augmenter ou diminuer l'amplitude de l'erreur totale à corriger, l'erreur  $-e_q + e_b$ , voir les sections V.1 et V.2. Ainsi dans ce cas le poids de bit  $p$  à partir duquel les erreurs deviennent corrigibles n'est plus le même que celui des signaux quantifiés.

#### V.4.1 Détection et identification des signaux d'alarme

L'étude du comportement du bruit de trafic sur les signaux d'alarmes illustrée au chapitre IV a permis de choisir les codeurs pour le cas spécifique des signaux d'alarme des voitures prioritaires. Ces codeurs ont été utilisés pour effectuer la quantification de ces signaux et par la suite leur détection et identification.

Les algorithmes 3 et 4 ont été utilisés pour la détection et l'identification d'un signal d'alarme d'une voiture de sapeurs-pompiers de longueur  $l_s = 10$  s, échantillonné à une fréquence  $F_s = 44100$  Hz. La résolution binaire utilisée pour représenter ces signaux en binaire est  $L = 14$  bits. Le bruit du canal de transmission est un bruit de trafic normalisé pour avoir des rapports signal à bruit du canal variant entre  $-1$  dB et  $15$  dB. Les quatre chaînes de transmission des figures V.1 à V.3 ont été considérées.

Les performances des algorithmes pour chaque chaîne de transmission seront analysées via les courbes COR paramétrées par  $\epsilon$ , en considérant plusieurs combinaisons  $(\eta, L - p)$ , ou  $L - p$  désigne le nombre de niveaux de bits sur lequel sont effectuées la détection et l'identification des signaux, et  $\eta$  le nombre de vecteurs  $V_j$  utilisé pour la détection et l'identification. Ce nombre permettra de choisir le seuil optimal de la détection et l'identification, ainsi *seuil* peut être pris comme tout nombre  $\eta$  offrant les meilleures performances des algorithmes. En observant les expressions de  $V$  et  $\tilde{V}$  données par les équations (V.1) et (V.2), on remarque que la norme  $\|\tilde{V}_j - V_j\|_1$  calculée au point 6 de l'algorithme 3 est proportionnelle à une puissance de 2, de même la norme  $\|\tilde{W} - W\|_1$  calculée au point 7 de l'algorithme 4 est proportionnelle à une puissance de 2. Pour cette raison  $\epsilon$  a été pris à valeurs dans  $\{0; 2^{L-4}; 2^{L-2}; 2^L; 2^{L+2}; 2^{L+4}\}$ .

#### V.4.2 Analyse de la détection et de l'identification dans le domaine temporel : Signal quantifié avant transmission

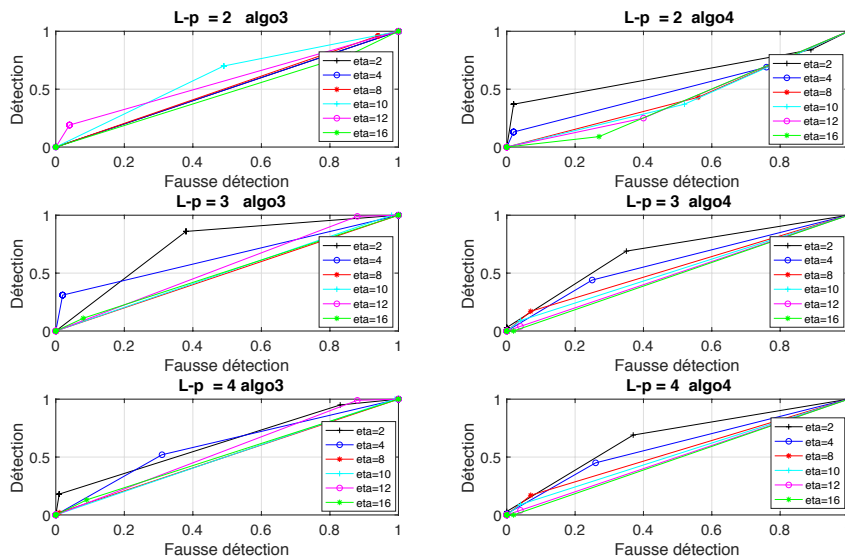


FIGURE V.5 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=-1 dB, signal quantifié avant transmission, domaine temporel.

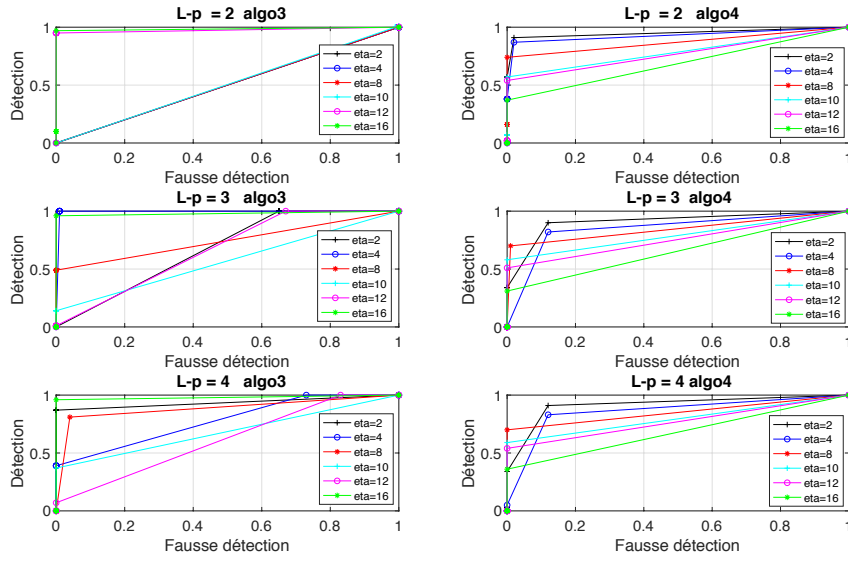


FIGURE V.6 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=5 dB, signal quantifié avant transmission, domaine temporel.

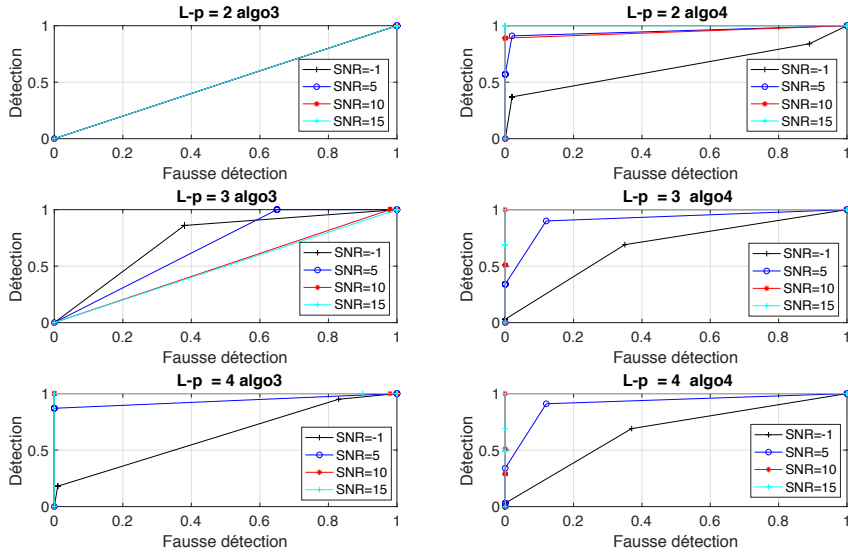


FIGURE V.7 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 2$ , pour plusieurs SNR, signal quantifié avant transmission, domaine temporel.

Les résultats de simulations représentés sur les figures V.5 à V.8 montrent que les algorithmes proposés pour la détection et l'identification des signaux arrivent à détecter et identifier les signaux quantifiés avant transmission dans le domaine temporel pour des

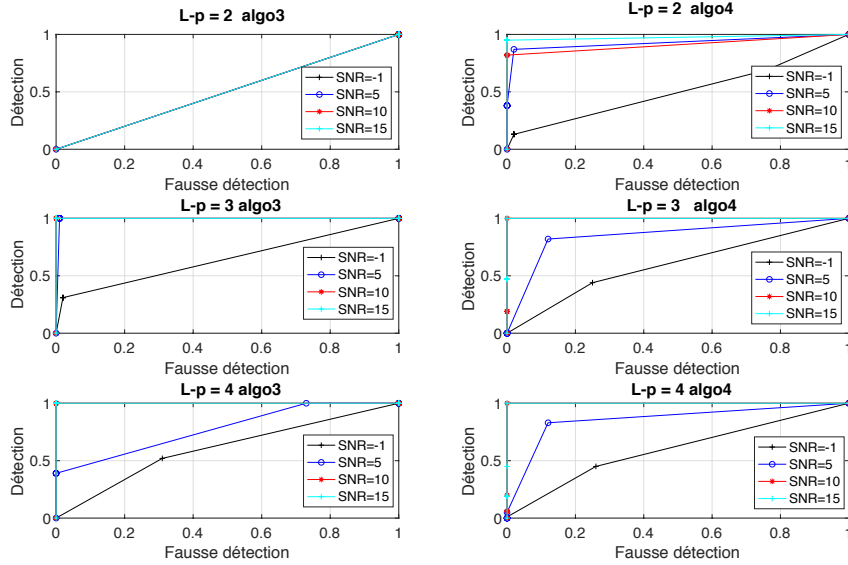


FIGURE V.8 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 4$ , pour plusieurs SNR, signal quantifié avant transmission, domaine temporel.

rappports signal à bruit modeste (de l'ordre de 5 dB). Par contre pour un rapport signal à bruit de -1 dB, les résultats obtenus ne sont pas satisfaisants, voir la figure V.5. La figure V.6 montre que pour un SNR=5 dB les meilleures performances de ces algorithmes sont obtenues pour  $\eta=2,4,6$  et  $L-p=2, 3, 4$  bits pour l'algorithme 4, pour l'algorithme 3, les meilleures performances sont obtenues pour  $\eta=4,16$  et  $L-p=2,3$  bits et pour  $\eta=2,16$  et  $L-p=4$  bits.

Pour les rapports signal à bruit allant de 5 dB à 10 dB, les figures V.8 et V.7 montrent que les meilleures performance de l'algorithme 4 sont obtenues pour  $\eta = 2, 4$  et  $L-p = 2$  bits. Cependant pour l'algorithme 3 ces performances sont atteintes pour  $\eta = 2$  et  $L-p = 4$  bits et pour  $\eta = 4$  et  $L-p = 3$  bits.

### V.4.3 Analyse de la détection et de l'identification dans le domaine fréquentiel : signal quantifié avant transmission

Comme indiqué par les figures V.9 à V.12 les algorithmes proposés détectent et identifient les signaux (signaux transmis via la chaîne de transmission V.3) y compris pour

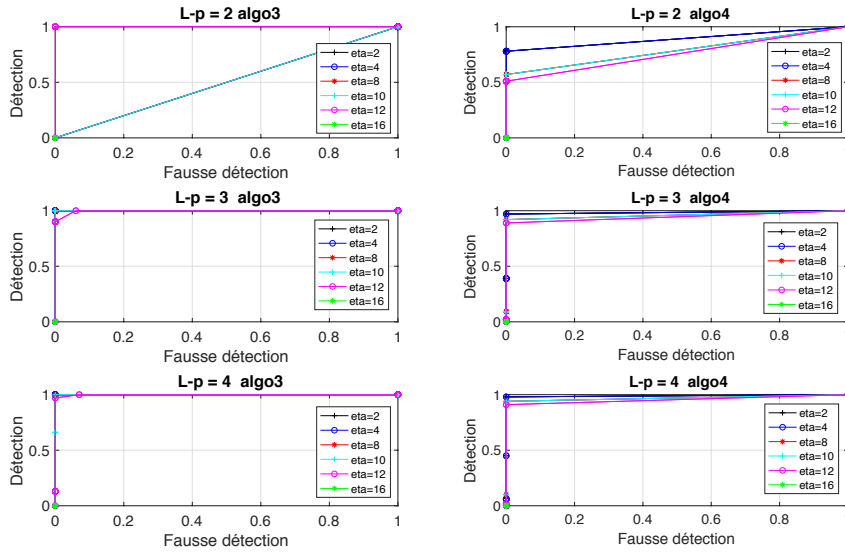


FIGURE V.9 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=-1 dB, signal quantifié avant transmission, domaine fréquentiel.

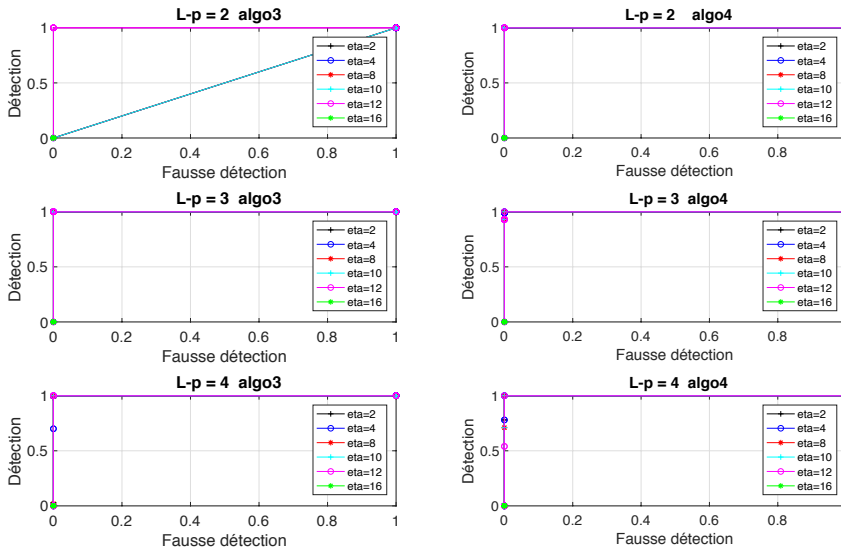


FIGURE V.10 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=5 dB, signal quantifié avant transmission, domaine fréquentiel.

des rapports signal à bruit faibles. Pour un rapport signal à bruit de -1 dB les meilleures performances des algorithmes 3 et 4 sont atteintes pour un nombre de bits  $L - p = 3, 4$  quelle que soit la valeur de  $\eta$ .



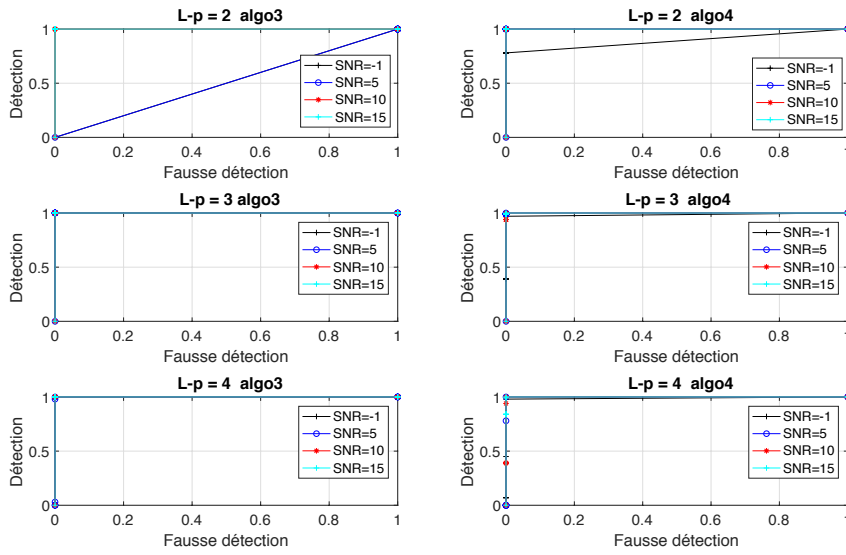


FIGURE V.11 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 2$ , pour plusieurs SNR, signal quantifié avant transmission, domaine fréquentiel.

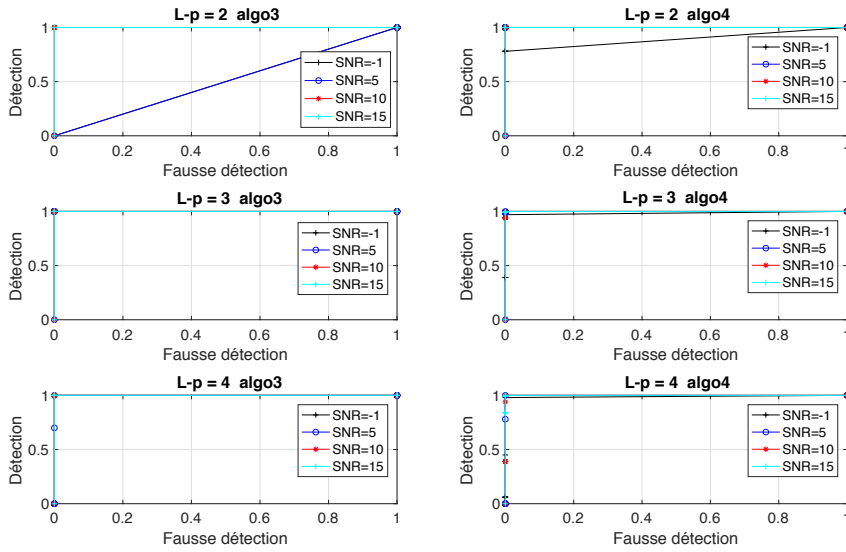


FIGURE V.12 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 4$ , pour plusieurs SNR, signal quantifié avant transmission, domaine fréquentiel.

En effectuant une comparaison entre les figures V.9 à V.12 et les figures V.5 à V.8 on remarque que ces algorithmes donnent des résultats meilleurs quand il s'agit d'une détection et d'une identification dans le domaine fréquentiel que dans le domaine temporel.

### V.4.4 Analyse de la détection et de l'identification dans le domaine temporel : signal non quantifié avant transmission

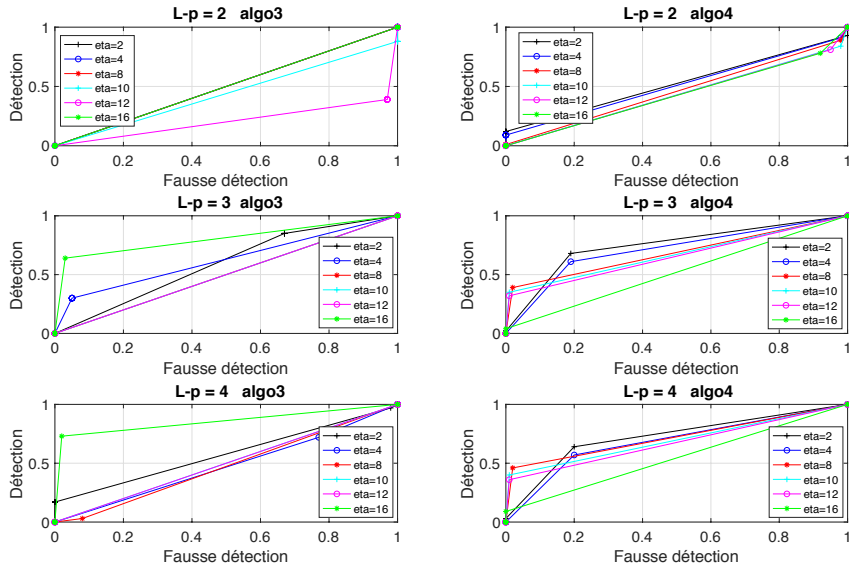


FIGURE V.13 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=5 dB, signal non quantifié avant transmission, domaine temporel.

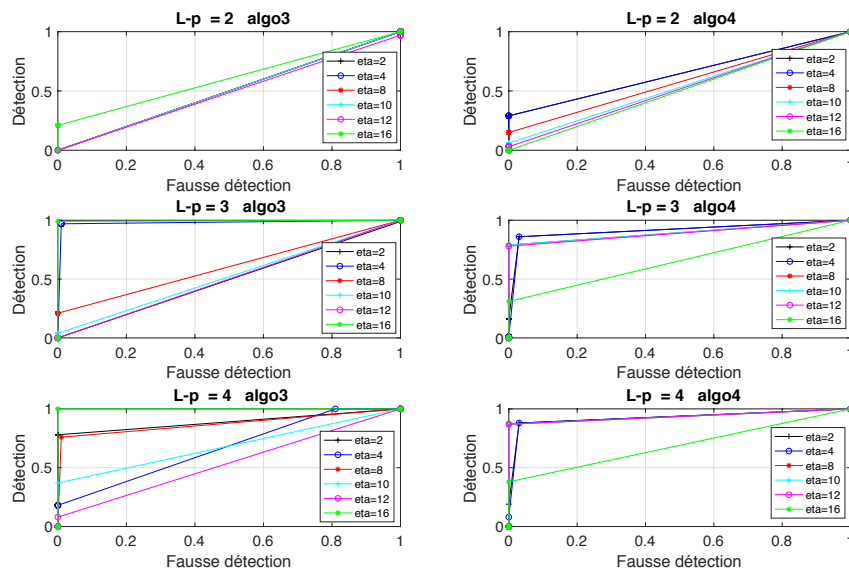


FIGURE V.14 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=10 dB, signal non quantifié avant transmission, domaine temporel.

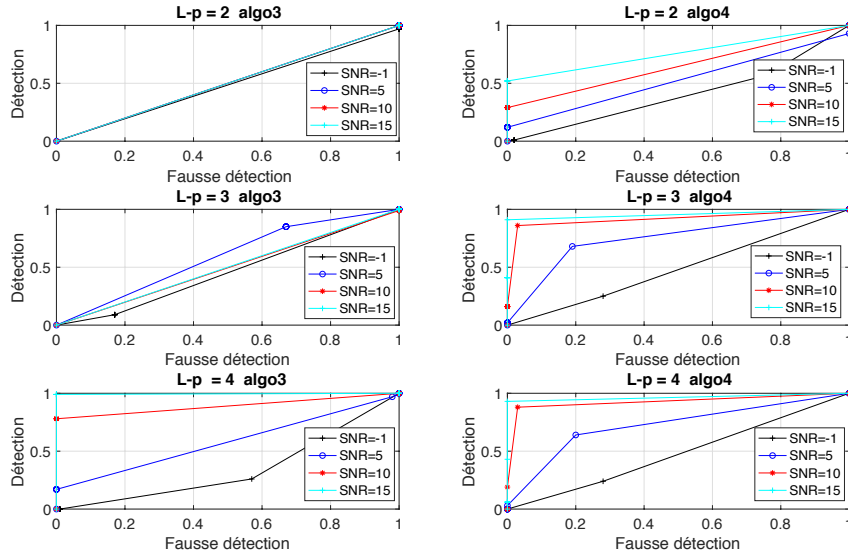


FIGURE V.15 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 2$ , pour plusieurs SNR, signal non quantifié avant transmission, domaine temporel.

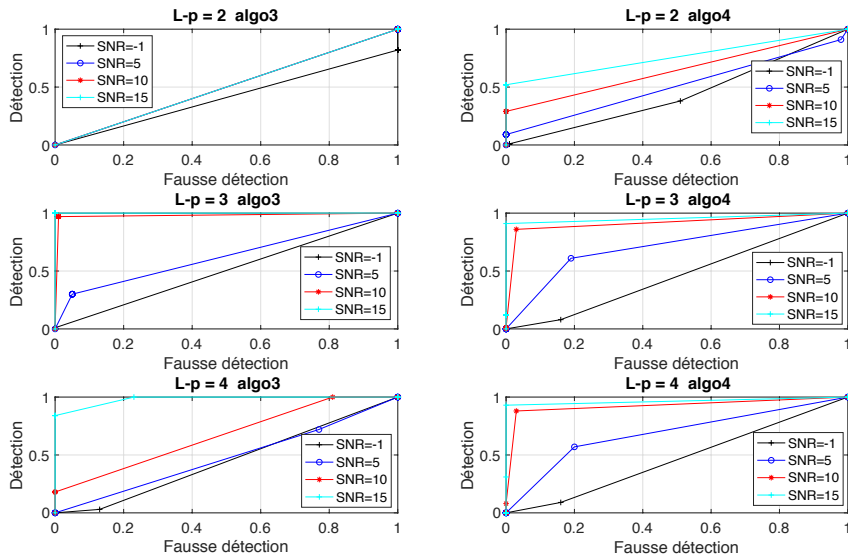


FIGURE V.16 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 4$ , pour plusieurs SNR, signal non quantifié avant transmission, domaine temporel.

Les simulations montrent à partir des figures V.13 à V.16 que les algorithmes proposés pour la détection et l'identification des signaux sont capables de détecter et d'identifier les signaux non quantifiés avant transmission (la chaîne de transmission illustrée à la figure V.2) et pour des rapports signal à bruit faibles de l'ordre de 5 dB et plus. Cependant,

pour un rapport signal à bruit de -1 dB, les performances de ces algorithmes se dégradent. Ces algorithmes sont capables de détecter et d'identifier un signal à partir d'une petite portion de ce signal :  $\eta=4$  et  $\eta=2$ , voir les figures V.15 et V.16.

Pour un rapport de 5 dB les meilleures performances de l'algorithme 3 sont obtenues pour  $\eta=16$  et  $L-p=3,4$  bits. Pour l'algorithme 4 les meilleures performances sont obtenues pour  $\eta=2, 4$  et  $L-p=3,4$  bits, voir la figure V.13. Cependant pour des rapports signal à bruit de 10 dB et 15 dB les meilleures performances de ces algorithmes sont obtenues pour  $\eta=4$  et  $L-p=3$  bits. Pour  $\eta = 2$ , les meilleures performances de l'algorithme 3 sont obtenues pour  $L-p=4$  bits, cependant celles de l'algorithme 4 sont obtenues pour  $L-p=3,4$  bits, voir les figures V.15 et V.16.

En comparant les figures V.13 à V.16 et les figures V.5 à V.8 on remarque que la détection et l'identification des signaux non quantifiés est moins efficace que celle des signaux quantifiés. En effet, ceci découle de la propriété de protection contre le bruit qu'apporte la quantification aux signaux. Pour les signaux non quantifiés, le bruit de transmission introduit une distorsion sur ces derniers qui rend la détection et l'identification de ces signaux moins performante que celle des signaux quantifiés avant transmission, voir les sections V.2 et V.1.

#### **V.4.5 Analyse de la détection et de l'identification dans le domaine fréquentiel : signal non quantifié avant transmission**

Les résultats des simulations représentés par les figures V.17 à V.20 (simulations effectuées dans le cas de la chaîne de transmission illustrée à la figure V.4) montrent que les algorithmes proposés pour la détection et l'identification des signaux sont capables de détecter et d'identifier des signaux non quantifiés avant transmission pour des rapports signal à bruit de l'ordre de 5 dB ou plus. Toutefois, les performances de ces algorithmes se dégradent quand il s'agit des rapports signal à bruit de l'ordre de -1 dB. Ces al-

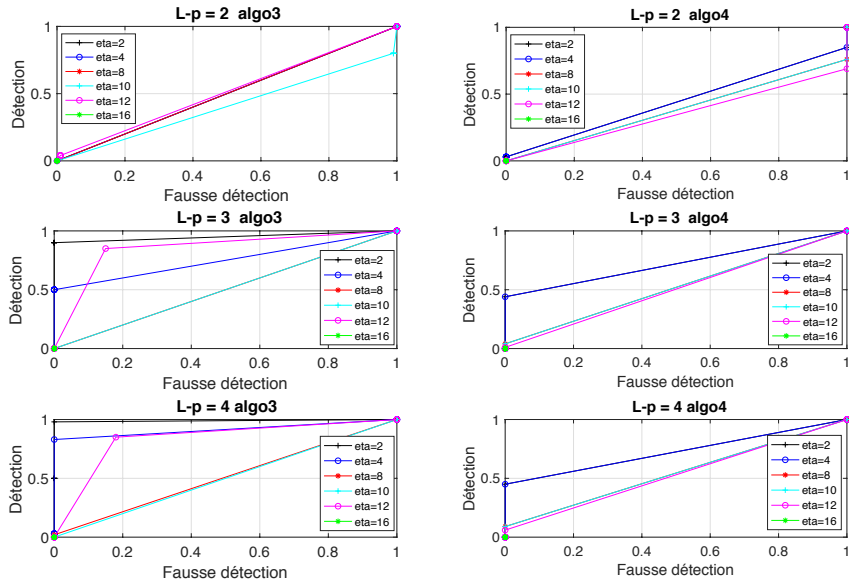


FIGURE V.17 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=5 dB, signal non quantifié avant transmission, domaine fréquentiel.

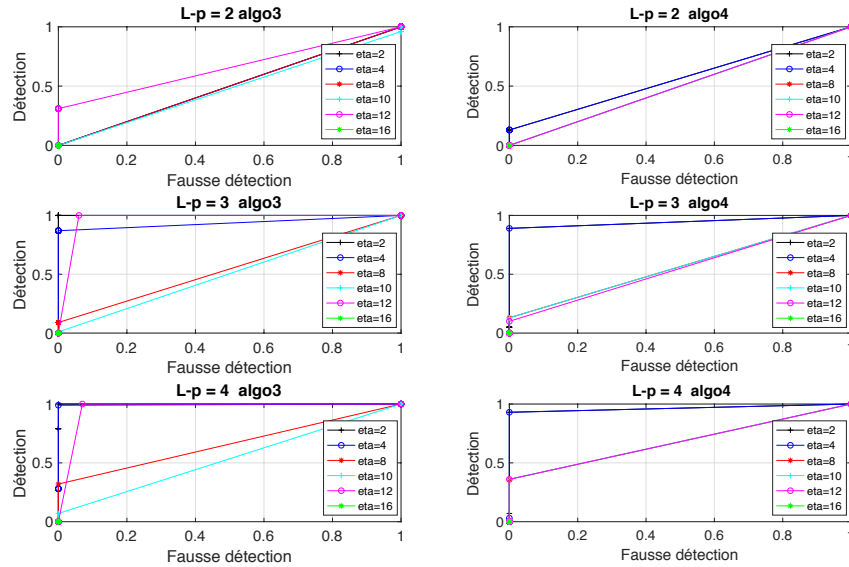


FIGURE V.18 – Courbes COR paramétrées par  $\epsilon$ , pour plusieurs valeurs de  $\eta$ , pour un SNR=10 dB, signal non quantifié avant transmission, domaine fréquentiel.

algorithmes sont capables d'effectuer une détection et une identification d'un signal non quantifié à partir d'une petite portion de ce signal, voir les figures V.19 à V.20 ( $\eta=2$  et  $\eta=4$ ).

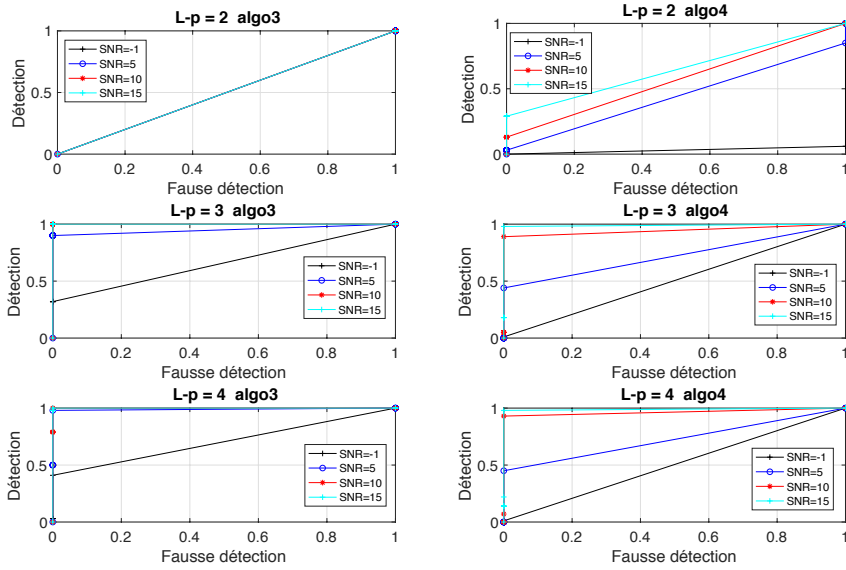


FIGURE V.19 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 2$ , pour plusieurs valeurs du SNR, signal non quantifié avant transmission, domaine fréquentiel.

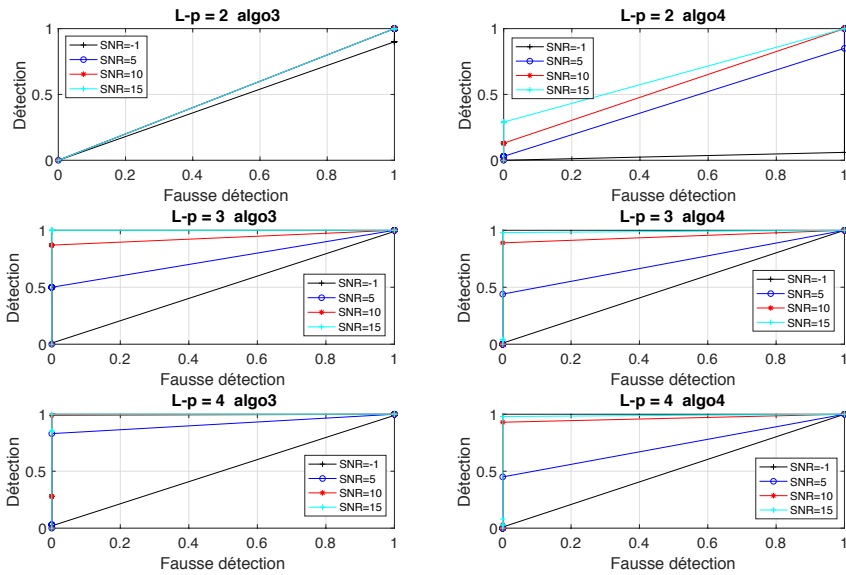


FIGURE V.20 – Courbes COR paramétrées par  $\epsilon$ ,  $\eta = 4$ , pour plusieurs valeurs du SNR, signal non quantifié avant transmission, domaine fréquentiel.

Les figures V.19 à V.20 montrent que pour des rapports signal à bruit de 5 dB, 10 dB et 15 dB, les meilleures performances de ces algorithmes sont obtenues pour  $\eta = 2, 4$  et

$L-p=3, 4$  bits. Toutefois, pour ces rapports signal à bruit les performances de l'algorithme 3 sont meilleures que celles de l'algorithme 4.

En effectuant des comparaisons entre les figures V.17 à V.20 et les figures V.13 à V.16 on remarque que les algorithmes de détection et d'identification appliqués aux signaux non quantifiés donnent un meilleur résultat dans le domaine fréquentiel que dans le domaine temporel pour une détection et une identification effectuées sur  $L-p=3$  bits ou  $L-p=4$  bits. Par contre pour une détection et une identification des signaux effectuées sur un nombre de bits  $L-p=2$  bits, ces algorithmes ont de mauvaises performances.

#### V.4.6 Analyse de la détection et de l'identification des signaux : comparaison avec la méthode de corrélation

La méthode de corrélation calcule la fonction de corrélation donnée par l'équation (V.5) entre les signaux stockés et l'information  $V_b$  du signal reçu.

$$\varphi(k) = \frac{\sum_{n=k}^{N+k-1} V_b(n)V(n-k)}{\sqrt{\sum_{n=k}^{N+k-1} V_b(n)^2 \sum_{n=0}^{N-1} V(n)^2}}, \quad (\text{V.5})$$

où  $N$  désigne la longueur du vecteur  $V$  d'information stocké et  $k$  un retard. Le vecteur  $V_b$  est la combinaison des  $L-p$  vecteurs binaires de poids fort du signal reçu. Ce vecteur  $V_b$  correspond au vecteur  $V$  bruité en présence du signal à chercher et au bruit de transmission en absence du signal.

Un signal est détecté par la corrélation si pour un retard  $k$ , la valeur de la fonction  $\varphi(k)$  est au dessus d'un certain seuil. Dans ces expériences le seuil  $s_d$  de détection de la corrélation est à valeurs dans  $\{0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.7; 0.8\}$ . Ainsi des courbes COR paramétrées par ce seuil ont été calculées pour effectuer des comparaisons entre cette méthode et les algorithmes de détection et d'identification des signaux proposés dans ce chapitre. Ces comparaisons sont effectuées dans le cas des signaux quantifiés avant transmission.

Pour une détection et une identification des signaux dans le domaine fréquentiel sur un

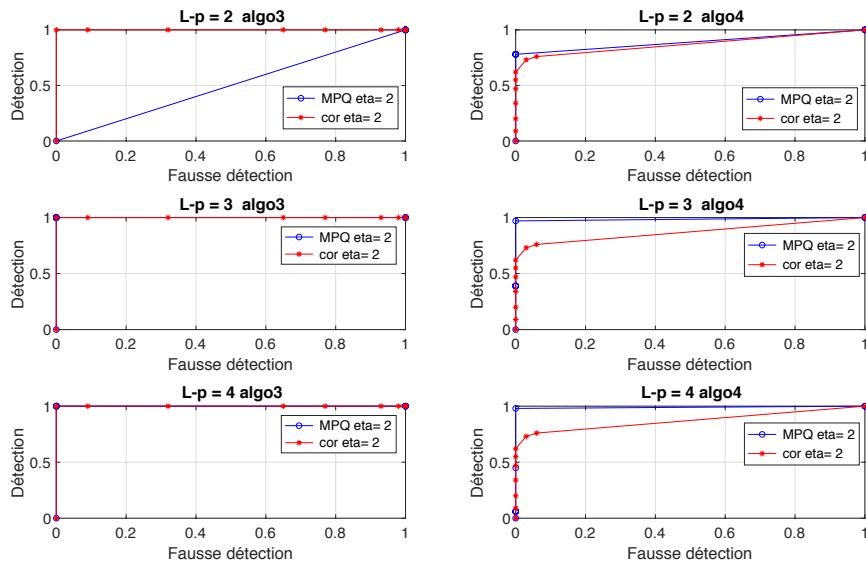


FIGURE V.21 – Courbes COR paramétrées par  $\epsilon$  pour les algorithmes proposés et par  $s_d$  pour la corrélation, pour un SNR=-1 dB,  $\eta = 2$ , signal quantifié avant transmission, domaine fréquentiel.

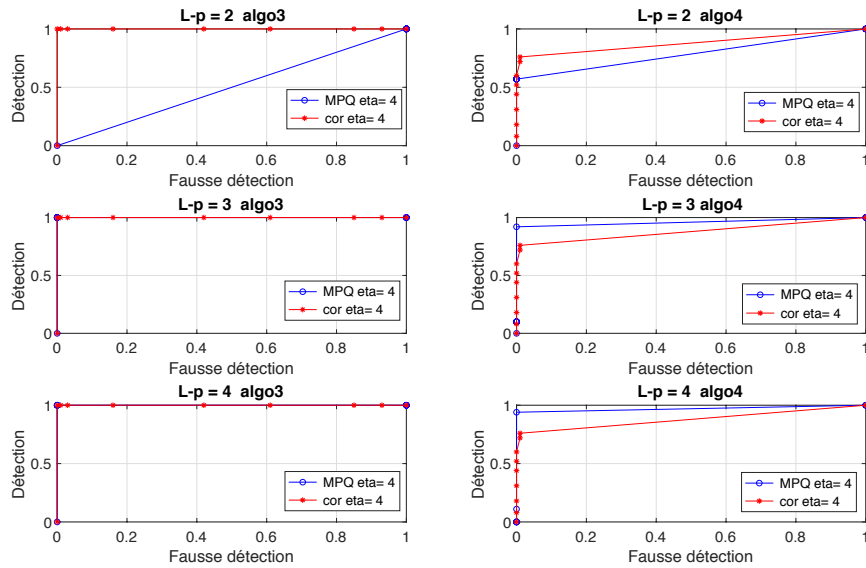


FIGURE V.22 – Courbes COR paramétrées par  $\epsilon$  pour les algorithmes proposés et par  $s_d$  pour la corrélation, pour un SNR=-1 dB,  $\eta = 4$ , signal quantifié avant transmission, domaine fréquentiel.

nombre de bits  $L-p = 3, 4$  bits, les figures V.21 et V.22 montrent que les performances des algorithmes proposés sont meilleures que celles de la méthode de corrélation. Par contre



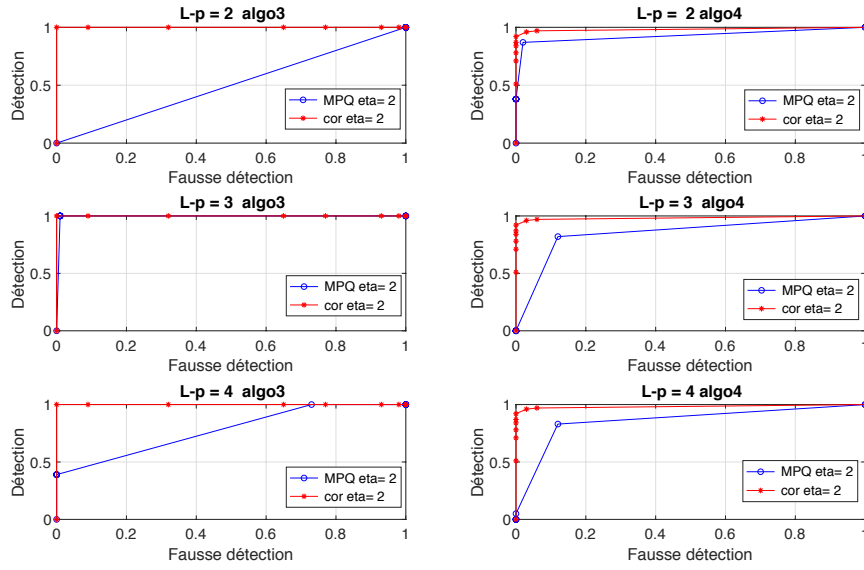


FIGURE V.23 – Courbes COR paramétrées par  $\epsilon$  pour les algorithmes proposés et par  $s_d$  pour la corrélation, pour un SNR=5 dB,  $\eta = 2$ , signal quantifié avant transmission, domaine temporel.

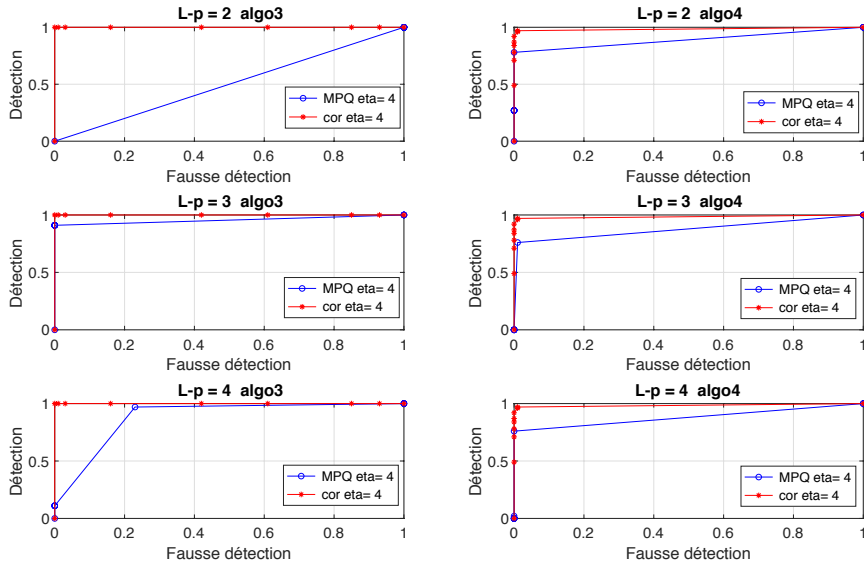


FIGURE V.24 – Courbes COR paramétrées par  $\epsilon$  pour les algorithmes proposés et par  $s_d$  pour la corrélation, pour un SNR=5 dB,  $\eta = 4$ , signal quantifié avant transmission, domaine temporel.

pour une détection et une identification établies sur un nombre de bits égal à  $L - p = 2$ , la méthode de corrélation devient plus performante.

Cependant, pour une détection et une identification des signaux dans le domaine temporel sur un nombre de bits  $L - p=3$ , 4 bits, les figures V.23 et V.24 montrent que la méthode de corrélation assure des performances meilleures que celles des algorithmes proposés.

Toutefois, l'algorithme 3 présente des performances équivalentes à celles de la méthode de corrélation pour une détection et une identification effectuées sur un nombre de bits  $L - p=3$  bits et  $\eta = 2$ , voir la figure V.23.

## V.5 Détection et identification des signaux d'alarme quantifiés par recherche de signature

Dans cette partie l'algorithme de détection des signaux d'alarme proposé à la section II.1.4 du chapitre II a été appliqué aux signaux d'alarme quantifiés. Les chaînes de transmission V.1 et V.2 ont été simulées dans le cas où le bruit de transmission est un bruit de trafic normalisé en amplitude pour avoir des rapports signal à bruit de 5 dB et 10 dB. Les chaînes de transmission V.4 et V.3 où la quantification est effectuée dans le domaine fréquentiel ne sont pas utilisées, car on a vu à la section III.3.3 du chapitre III que la signature des signaux d'alarme quantifiés dans le domaine fréquentiel est noyée dans le bruit de quantification (non détectable car camouflée par le bruit de quantification fréquentielle même si celui-ci a été réduit par la quantification proposée au chapitre II), voir la figure III.12. Des expériences ont été effectuées sur des signaux d'alarme transmis via les chaînes de transmission citées ci-dessus, où l'algorithme 1 a été utilisé pour détecter la signature fréquentielle de ces signaux. Des courbes COR ont été calculées pour mesurer les performances de cet algorithme appliqué aux signaux quantifiés.

Chacune de ces courbes est paramétrée par  $\epsilon$  ( $\epsilon$  : vérifiant  $\frac{|\epsilon|}{f} \in \{0.05; 0.08; 0.1; 0.2; 0.25\}$  avec  $f$  fréquence fondamentale de l'alarme) pour un  $\mu$  fixé (pour plus de détail sur  $\mu$  et  $\epsilon$ ,

voir chapitre II). La figure V.25 représente des courbes COR correspondant à la détection du signal d'alarme non quantifié.

La figure V.26 représente les courbes COR dans le cas d'un signal d'alarme de sapeurs-pompiers quantifié avant et après transmission pour une détection fondée sur la recherche de la signature fréquentielle de l'alarme (cas de la chaîne V.1). La figure V.27 représente des courbes COR correspondant à la détection du signal d'alarme de sapeurs-pompiers quantifié après transmission (cas de la chaîne V.2).

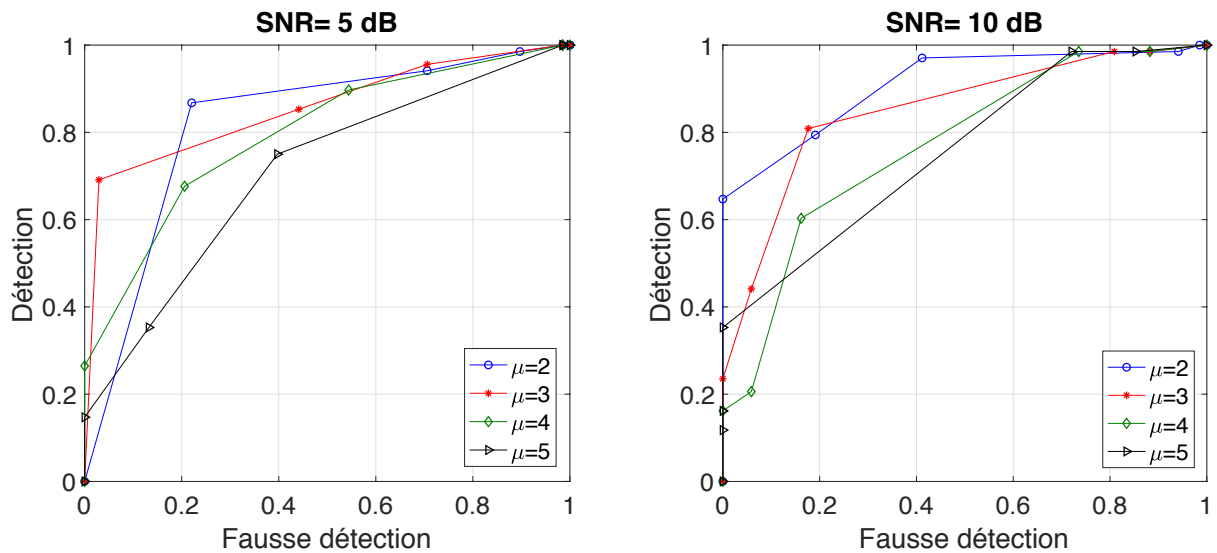


FIGURE V.25 – Courbes COR paramétrées par  $\epsilon$ , signal d'alarme de sapeurs-pompiers non quantifié.

Comme le montre la comparaison entre les figures V.25 , V.27 et V.26, la quantification dans le cas des deux chaînes de transmission V.1 et V.2 améliore aussi les performances de l'algorithme de détection proposé au chapitre II pour une détection fondée sur la recherche de la signature fréquentielle de l'alarme, voir la figure V.28 pour une comparaison dans le cas spécifique où  $\mu = 3$ .

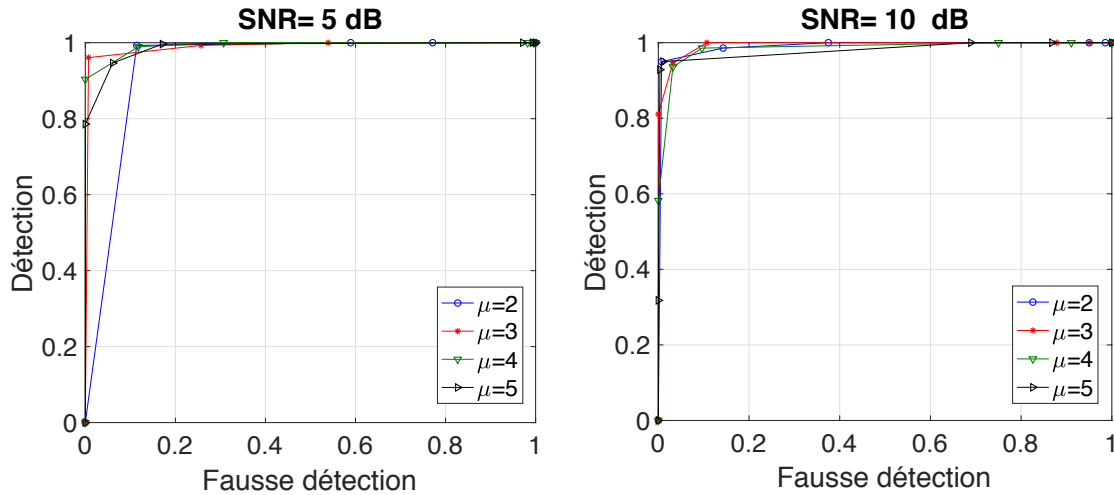


FIGURE V.26 – Courbes COR paramétrées par  $\epsilon$ , signal d’alarme de sapeurs-pompiers quantifié avant et après transmission (chaîne V.1).

## V.6 Conclusion

L’avantage des algorithmes 3 et 4 proposés dans ce chapitre est d’éviter de reconstruire entièrement le signal quantifié pour pouvoir identifier ces signaux. En effet, ces algorithmes sont capables d’effectuer la détection et l’identification des signaux sur des vecteurs  $V$ , représentés sur un nombre limité de bits  $L - p$  bits (2 à 4 bits), pour une détection et une identification des signaux dans le domaine temporel et fréquentiel, et pour des signaux quantifiés ou pas avant transmission. Les algorithmes proposés dans ce chapitre effectuent une détection et une identification conjointes.

Ces algorithmes 3 et 4 ne sont pas spécifiques aux signaux d’alarmes mais peuvent être utilisés pour la détection et l’identification de n’importe quel type de signal.

L’algorithme 1 de détection des signaux proposés au chapitre II présente des meilleurs résultats quand il est appliqué aux signaux d’alarmes quantifiés (signaux quantifiés avant et après transmission ou signaux quantifiés uniquement après transmission) que quand il est appliqué aux signaux non quantifiés.

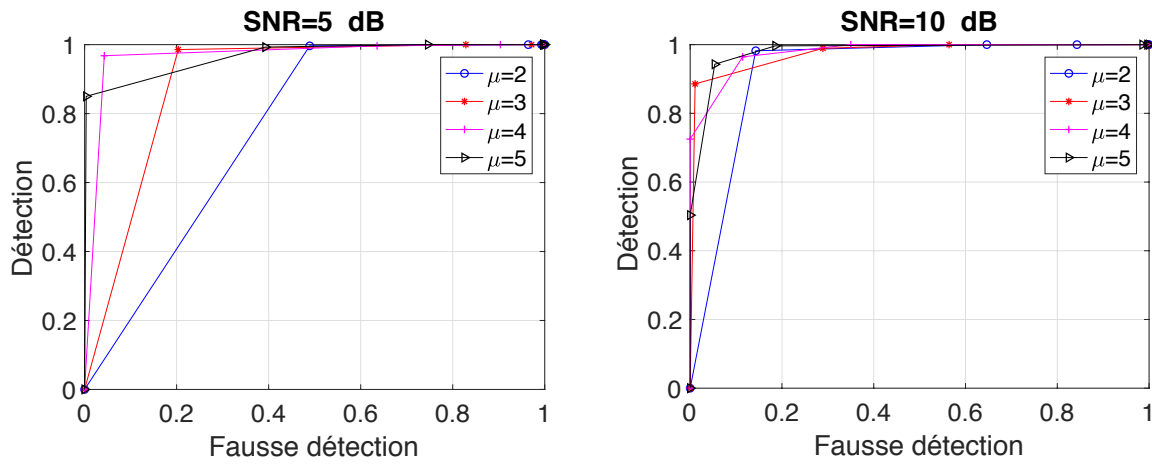


FIGURE V.27 – Courbes COR paramétrées par  $\epsilon$ , signal d’alarme de sapeurs-pompiers quantifié après transmission (chaîne V.2).

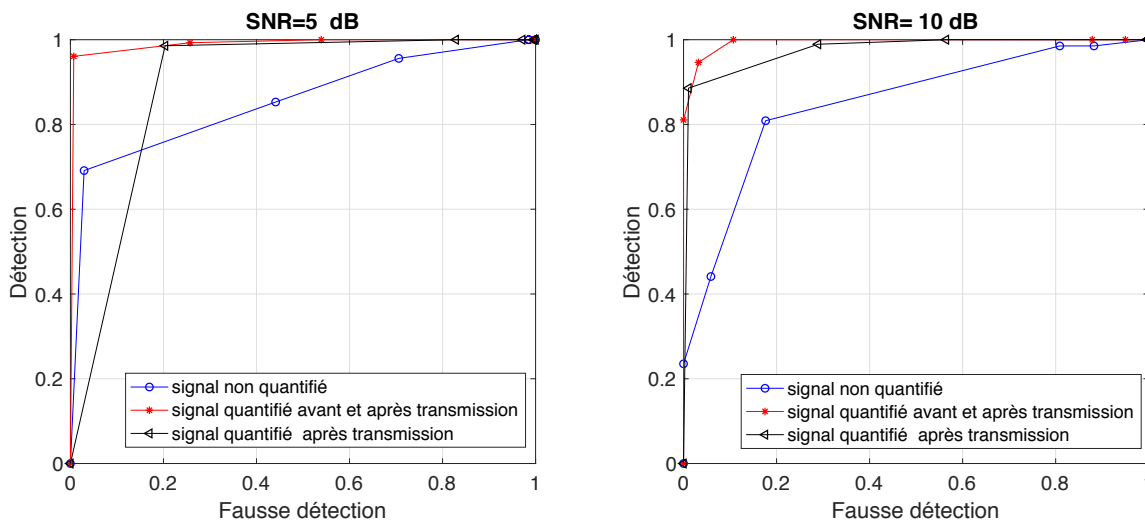


FIGURE V.28 – Courbes COR paramétrées par  $\epsilon$ , signal d’alarme de sapeurs-pompiers,  $\mu=3$ .

# Conclusion

Nous avons proposé au chapitre II un algorithme de détection des signaux d'alarmes fondé sur la recherche de la signature temporelle et fréquentielle caractérisée par les durées des deux tonales et les fréquences fondamentales correspondantes. Cet algorithme donne des résultats satisfaisants dans le cadre de la détection de la présence d'un signal d'alarme. Par contre celui-ci n'est pas adapté pour l'identification : l'identification comme la détection sont fondées sur les paramètres temps-fréquence de l'alarme, or ces paramètres peuvent être proches les uns des autres ce qui peut facilement entraîner des erreurs sur l'identification.

Ensuite au chapitre III nous avons proposé une méthode de quantification fondée sur le codage correcteur d'erreurs permettant de rendre les signaux robustes au bruit de transmission. Toutefois, une telle protection se paye en termes d'erreur de quantification.

Afin de réduire l'erreur de quantification, nous avons proposé au chapitre IV une quantification correctrice d'erreur fondée sur le principe du matching pursuit utilisant des dictionnaires de mots de codes générés par des codes correcteurs d'erreurs. Contrairement à la quantification proposée au chapitre III, cette quantification utilise la distance euclidienne comme métrique pour effectuer le codage au lieu de la distance de Hamming utilisée dans le chapitre III.

Cette dernière quantification conduit à une erreur de quantification plus faible que la précédente. En plus de cela cette quantification peut jouer le rôle d'un algorithme de décodage robuste quand elle est utilisée au niveau du récepteur. Il a été montré (par des

simulations) dans le chapitre IV que cette quantification possède de meilleures performances que celles des méthodes classiques de décodage des codeurs utilisés pour générer les dictionnaires de mots de code (codes BCH, codes convolutifs et turbo codes). Toutefois, la complexité de cet algorithme peut être plus grande que celle des décodeurs classiques à cause de la recherche dans le dictionnaire des mots de codes minimisant la métrique (distance euclidienne) du décodage effectué par cette quantification.

Appliquée aux signaux d'alarme, cette quantification améliore les performances de l'algorithme de détection des signaux d'alarme introduit au chapitre II.

En exploitant les propriétés données aux signaux par la quantification (protection et la forme spécifique), nous avons proposé au chapitre V deux algorithmes de détection et d'identification des signaux fondés sur cette quantification. Ces algorithmes sont capables de détecter et d'identifier dans les deux domaines temporel et fréquentiel des signaux quantifiés avant transmission et des signaux non quantifiés avant transmission. Pour effectuer la détection et l'identification d'un signal, ces algorithmes ont besoin uniquement d'une petite portion du signal. En effet, ces algorithmes sont capables de détecter et d'identifier un signal sur une petite fenêtre temporelle du signal et une représentation sur un nombre de bits limité (3, 4 bits). Les performances de ces algorithmes dépassent celles de la méthode de corrélation, pour une détection et une identification effectuées dans le domaine fréquentiel avec une représentation sur un nombre de bits limité à 3, 4.

# Bibliographie

- [1] L. Besacier Alain Dufaux, M. Ansorge and F.Pellandini. Automatic detection and classification of wide-band acoustic signals. *137th Meeting of the Acoustical Society of America and Forum Acusticum*, march 1999.
- [2] Claude Berrou Alain Glavieux and Punya Thitimajshima. Near Shannon limit error - correcting coding and decoding : Turbo-codes. *IEEE*, pages 1064–1070, February 1993.
- [3] Toby Berger. Minimum entropy quantizers and permutation codes. *IEEE Transactions on Information Theory*, pages 150–157, 1982.
- [4] F. Beritelli, S. Casale, A. Russo, and S. Serrano. An automatic emergency signal recognition system for the hearing impaired. *The 12th Digital Signal Processing Workshop*, pages 179–182, september 2006.
- [5] E. Berlekamp. On decoding binary Bose-Chaudhuri-Hoc-Quenghem codes. *IEEE Transactions on Information Theory*, 11 :577–579, October 1965.
- [6] R.C. Bose and D.K. Ray-Chaudhuri. On a class of error correcting binary group codes. *Information and Control*, 3 :68 – 79, 1960.
- [7] Marc-André Carbonneau, Narimene Lezzoum, Jérémie Voix, and Ghyslain Gagnon. Detection of alarms and warning signals on an digital in-ear devise. *International Journal of Industrial Ergonomics*, 2012.
- [8] Dan Censor, Ben Gurion, and Beer Sheva. Theory of the Doppler effect : Fact, fiction and approximation. *Radio Science*, 19 :1027–1040, July-Aug 1984.
- [9] Shaobing Chen and David Donoho. *Basis Pursuit*. Standford University, Standford, CA 94305.
- [10] A. D. Cheveigne and H. Kawahara Yin. A fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 2002.
- [11] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1) :1–38, 1977.
- [12] Geoffrey Dvis, Stéphane Mallat, and Zhifeng Zhang. *Adaptive Time-Frequency Approximations with Matching Pursuits*. Courant Istitut of Mathematical Sciences, Computer Science Departmemnt, 251 Mercer Street, New york, NY 10012, USA.
- [13] Robert J. Elliott, Lakhdar Aggoun, and John B. Moore. *Hidden Markov Models Hidden Markov Models : Estimation and Control*. Springer, 1995.



- [14] Y. Ephraim and D. Malah. Speech enhancement using a minimum mean square error log-spectral amplitude estimator. *IEEE Trans. on Acoustics, Speech and Signal Processing*, (2), 1985.
- [15] G. D. Forney, JR. Member, and IEEE. Convolutional codes : Algebraic structure. *Transaction on Information Theory*, IT-16(6), November 1970.
- [16] R. G. Gallager. Low-density parity-check codes. *IRE Transactions on Information Theory*, 1962.
- [17] Stéphane G.Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41 :3397–3415, December 1993.
- [18] D. W. Hagelbarger. Recurrent codes : Easily mechanized, burst-correcting, binary codes. *The Bell System Technical Journal*, July 1959.
- [19] A. Hocquenghem. Codes correcteurs d’erreurs. *Chiffres (in French)*, 2 :147–156, sept. 1959.
- [20] Haibin Huang, Susanto Rahardja, Rongshan Yu, and Xiao Lin. Integer MDCT with enhanced approximation of the DCT-IV. *IEEE Transactions on Signal Processing*, 54(3), 2006.
- [21] H. Imai and S. Hirakawa. A new multilevel coding method using error-correcting codes. *IEEE Trans. on Inf. Theory*, 23 :371–377, 1977.
- [22] A.K. Jain, Jianchang Mao, and K.M. Mohiuddin. Artificial neural networks : a tutorial. *Computer*, 29(3) :31–44, March 1996.
- [23] Éric Tannier Jean-Guillaume Dumas, Jean-Louis Roch and Sébastien Varrette. *Théorie des Codes : Compression, Cryptage, Correction*. Dunod, Paris, 2007.
- [24] J.D. Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on selected areas in communications*, (2) :314–329, february 1988.
- [25] B. H. Juang. On the hidden markov model and dynamic time warping for speech recognition-a unified view. *AT&T Bell Laboratories Technical Journal*, 63(7), September 1984.
- [26] James F Kaiser. Nonrecursive digital filter design using the i0-sinh window function. *IEEE International Symposium on Circuits and Systems*, pages 20–23, April 1974.
- [27] Yoseph Linde, Andres Buzo, and Robert M.Gray. An algorithm for vector quantizer design. *IEEE Transactions on communications*, (1) :84–95, 1980.
- [28] Robert A. Lutfi and Inseok Heo. Automated detection of alarm sounds. *Acoustical Society of America*, 2012.
- [29] A. Dufaux M. Ansorge, L. Besacier and F. Pellandini. Automatic sound recognition relying on statistical methods, with application to telesurveillance. *International Workshop on Intelligent Communication Technologies and Applications*, pages 116–120, May 1999.
- [30] L. Besacier M. Ansorge, A. Dufaux and F. Pellandini. Automatic sound detection and recognition for noisy environment. *European Signal Processing Conference*, pages 1033–1036, september 2000.

- [31] D.J.C. MacKay and R.M. Neal. Near Shannon limit performance of low density parity check codes. *Electronics Letter*, 33(6), march 1997.
- [32] James L. Massey. Step-by-step decoding of the Bose-Chaudhuri-Hocquenghem codes. *IEEE Transactions on Information Theory*, 4, 1965.
- [33] Joel Maxi. Quantizing for minimum distorsion. *IEEE Transactions on Information Theory*, pages 7–12, Mars 1960.
- [34] F. Meucci, L. Pierucci, E. Del Re, L. Lastrucci, and P. Desii. A real-time siren detector to improve safety of guide in traffic environment. *16th European Signal Processing Conference*, August 2008.
- [35] B. Milner and X. Shao. Clean speech reconstruction from MFCC vectors and fundamental frequency using an integrated front-end. *Speech Communication*, 48(6) :697–715, 2006.
- [36] Joao F. C. Mota, Joao M. F. Xavier, Pedro M. Q. Aguiar, and Markus Püschel. Distributed basis pursuit. *math.OC*, march 2012.
- [37] W. W. Peterson. *Error Correcting Codes*. IRE Transaction on Information Theory, 1961.
- [38] W. W. Peterson and D. T. Brown. Cyclic codes for error detection. *Proceedings of the IRE*, pages 228–235, january 1961.
- [39] Henry D. Pfister. *Algebraic Decoding of Reed-Solomon and BCH Codes*. Texas AM University.
- [40] Jonathan Pinel and Laurent Girin. Sparsification of audio signals using the MDCT/IntMDCT and a psychoacoustic model—Application to informed audio source separation. In *42nd Int. Conf. Semantic Audio*, July 2011.
- [41] Stuart P.Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2) :129–137, Mars 1956.
- [42] J. Princen and A. Bradley. Analysis/synthesis filter bank design based on time domain aliasing cancellation. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 34 :1153–1161, 1986.
- [43] J. Princen, A. Johnson, and A. Bradley. Subband/transform coding using filter bank designs based on time domain aliasing cancellation. In *ICASSP '87*, volume 12, pages 2161–2164, Apr 1987.
- [44] L. R. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.
- [45] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recohntion. *Proceedings of the IEEE*, 77(2), February 1989.
- [46] IEEE Rainer Martin, Senior Member. Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Transactions on Speech and Audio Processing*, 9, July 2001.
- [47] Douglas Reynolds. *Gaussian Mixture Models*. MIT Lincoln Laboratory.
- [48] Douglas Reynolds. *Gaussian Mixture Models*. Springer US, 2015.

- [49] Ens Schröder, Benjamin Cauchi, Marc René Schädler, Niko Moritz, Kamil Adiloglu, Jörn Anemüller, Simon Doclo, Birger Kollmeier, and Stefan Goetze. Acoustic event detection using signal enhancement and spectro-temporal feature extraction. *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, October 2013.
- [50] Claude Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27 :379–423, 623–656, July, October 1948.
- [51] Claude Shannon. Coding theorems for a discrete source with a fidelity criterion. In *IRE National Convention Record*, pages 142–163, 1959.
- [52] Gilbert Strang. Karmarkar’s algorithm and its place in applied mathematics. *The Mathematical Intelligencer*, 9(2), 1987.
- [53] Jurij F. Tasic, Mohamed Najim, and Michael Ansorge. *Intelligent Integrated Media Communication Techniques*. Luwer Academic Publishers, 2003.
- [54] Punya Thitimajshima. Recursive systematic convolutional codes and application to parallel concatenation. *Proceedings of GLOBECOM ’95*, November 1995.
- [55] Stephen B. Wicker and Vijay K. Bhargava. *Reed-Solomon codes and their applications*. IEEE press, 1994.

---

## **Représentation de signaux robuste aux bruits - Application à la détection et l'identification des signaux d'alarme**

---

Ces travaux ont pour application la détection l'identification des signaux audio et particulièrement les signaux d'alarmes de voitures prioritaires. Dans un premier temps, nous proposons une méthode de détection des signaux d'alarme dans un environnement bruité, fondée sur des techniques d'analyse temps-fréquence des signaux. Cette méthode permet de détecter et d'identifier des signaux d'alarmes noyés dans du bruit, y compris pour des rapports signal à bruit négatifs. Puis nous proposons une quantification des signaux robuste aux bruits de transmission. Il s'agit de remplacer chaque niveau de bit d'un vecteur d'échantillons temporels ou fréquentiels par un mot binaire de même longueur fourni par un codeur correcteur d'erreur. Dans une première approche, chaque niveau de bits est quantifié indépendamment des autres selon le critère de minimisation de la distance de Hamming. Dans une seconde approche, pour réduire l'erreur de quantification à robustesse égale, les différents niveaux de bits sont quantifiés successivement selon un algorithme de type matching pursuit. Cette quantification donne aux signaux une forme spécifique permettant par la suite de les reconnaître facilement parmi d'autres signaux. Nous proposons donc enfin deux méthodes de détection et d'identification des signaux fondées sur la quantification robuste, opérant dans le domaine temporel ou dans le domaine fréquentiel, par minimisation de la distance entre les signaux reçus restreints à leurs bits de poids fort et les signaux de référence. Ces méthodes permettent de détecter et d'identifier les signaux dans des environnements à rapport signal à bruit très faible et ceci grâce à la quantification. Par ailleurs, la première méthode, fondée sur la signature temps-fréquence, s'avère plus performante avec les signaux quantifiés.

---

Mots-clés : signaux audio, alarmes, détection, identification, quantification robuste, analyse temps-fréquence, quantification vectorielle, codage de canal, matching pursuit.

---

## **Signals representation robust to noise - Application to the detection and identification of alarm signals**

---

This work targets the detection and identification of audio signals and in particular alarm signals from priority cars. First, we propose a method for detecting alarm signals in a noisy environment, based on time-frequency signal analysis. This method makes it possible to detect and identify alarm signals embedded in noise, even with negative signal-to-noise ratios. Then we propose a signal quantization robust against transmission noise. This involves replacing each bit level of a vector of time or frequency samples with a binary word of the same length provided by an error-correcting encoder. In a first approach, each bit level is quantized independently of the others according to the Hamming distance minimization criterion. In a second approach, to reduce the quantization error at equal robustness, the different bit levels are quantized successively by a matching pursuit algorithm. This quantization gives the signals a specific shape that allows them to be easily recognized among other signals. Finally, we propose two methods for detecting and identifying signals based on robust quantization, operating in the time domain or in the frequency domain, by minimizing the distance between the received signals restricted to their high-weight bits and the reference signals. These methods make it possible to detect and identify signals in environments with very low signal-to-noise ratios, thanks to quantization. In addition, the first method, based on the time-frequency signature, is more efficient with quantized signals.

---

Key-word : audio signals, alarm, detection, identification, robust quantization, time-frequency analysis, vectorial quantization, channel coding, matching pursuit.

---

**Discipline : AUTOMATIQUE, SIGNAL, PRODUCTIQUE, ROBOTIQUE**

---

**Spécialité : Traitement du Signal**

Université de Reims Champagne-Ardenne

CRESTIC - EA 3804

Moulin de la Housse, 51897 Reims

