

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 DESCRIPTION DU SYSTÈME ET DE LA PROBLÉMATIQUE	5
1.1 Introduction.....	5
1.2 Survol du système mesurant la consommation énergétique résidentielle.....	5
1.3 Types de capteurs.....	6
1.3.1 Description du HES	6
1.3.2 Description du CT.....	8
1.4 Implémentation à l'intérieur de capteurs de courant.....	9
1.5 Traitement et transmission des données	11
1.5.1 Traitement des données effectué par les capteurs et l'UCT	12
1.5.2 Fréquence d'enregistrement et de transmission des données par les capteurs	12
1.6 Mise en contexte de la problématique	13
1.6.1 Problème de gain des capteurs.....	13
1.6.2 Problème de diaphonie entre les capteurs.....	13
1.7 Modélisation mathématique du système.....	14
1.8 Objectifs du mémoire et caractéristiques recherchées de l'algorithme	16
1.9 Solutions proposées	17
1.9.1 Solutions algorithme LMS.....	17
1.9.2 Solution FastICA combinée à l'algorithme LMS	18
1.10 Conclusion	18
CHAPITRE 2 REVUE DE LA LITTÉRATURE ET THÉORIE	21
2.1 Introduction.....	21
2.2 Revue de la littérature dans le domaine de la calibration de WSN.....	21
2.3 Théorie du LMS.....	22
2.3.1 Avantages et inconvénients de LMS.....	22
2.3.2 Algorithme du gradient.....	22
2.3.3 L'algorithme LMS	23
2.3.4 Choix de la grandeur du pas d'adaptation et convergence de LMS	24
2.4 Théorie de l'ICA.....	24
2.4.1 L'effet cocktail party.....	25
2.4.2 Modèle de base de l'ICA.....	26
2.4.3 Le blanchiment résout la moitié du problème d'ICA	27
2.4.4 Restrictions et ambiguïtés d'ICA.....	30
2.4.5 Indépendance, non-gaussianité et l'entropie négative	33
2.4.6 Avantages de l'algorithme FastICA.....	36
2.4.7 L'algorithme FastICA pour une composante.....	37
2.4.8 Estimation de plusieurs composantes indépendantes avec FastICA.....	39
2.5 Conclusion	41

CHAPITRE 3	SOLUTION UTILISANT L'ALGORITHME LMS POUR LA CALIBRATION DES CAPTEURS	43
3.1	Introduction.....	43
3.2	Retour sur la théorie de l'algorithme LMS.....	43
3.3	Résultats de simulation avec l'algorithme LMS.....	45
3.4	Résultats de simulation avec LMS sans diaphonie.....	46
3.4.1	Calibration de 3 capteurs avec LMS.....	46
3.4.2	Calibration de 30 capteurs avec LMS.....	49
3.5	Résultats de simulation avec LMS et diaphonie.....	54
3.5.1	Calibration de 3 capteurs avec LMS et diaphonie	55
3.6	Conclusion	57
CHAPITRE 4	SOLUTION UTILISANT L'ALGORITHME FASTICA POUR LA CALIBRATION DES CAPTEURS	59
4.1	Introduction.....	59
4.2	Modélisation du système en vue de l'utilisation de FastICA	59
4.3	Solutions proposées aux problèmes d'ambiguïté.....	60
4.4	Résultats de simulation avec l'algorithme FastICA.....	61
4.4.1	Calibration de 3 capteurs avec FastICA.....	62
4.4.2	Calibration de 30 capteurs avec FastICA.....	64
4.5	Conclusion	67
CHAPITRE 5	RÉSULTATS EXPÉRIMENTAUX.....	69
5.1	Introduction.....	69
5.2	Présentation du système pour les tests expérimentaux	69
5.2.1	Présentation des types d'installations	69
5.2.2	Précision des CTs et consommation des appareils	72
5.2.3	Tests pour mesurer le plancher de bruit des capteurs	73
5.3	Résultats des tests expérimentaux avec LMS	74
5.3.1	Résultats des tests avec l'installation 1 et LMS.....	75
5.3.2	Résultats des tests avec l'installation 2 et LMS.....	79
5.4	Résultats des tests expérimentaux avec FastICA.....	83
5.4.1	Résultats des tests avec l'installation 1 et FastICA	85
5.4.2	Résultats des tests avec l'installation 2 et FastICA	90
5.5	Conclusion	96
CONCLUSION.....	99
RECOMMANDATIONS	103
LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES.....	106

LISTE DES TABLEAUX

		Page
Tableau 4.1	Erreur (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 100 simulations	63
Tableau 4.2	Erreur (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0,5$ pour 100 simulations	64
Tableau 4.3	Erreur moyenne (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0$ pour 100 simulations	65
Tableau 4.4	Erreur moyenne (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0,5$ pour 100 simulations	66
Tableau 5.1	Types d'appareils pour chaque installation	71
Tableau 5.2	Précision des CTs	72
Tableau 5.3	Consommation des appareils	73
Tableau 5.4	Valeurs (A_{RMS}) du plancher de bruit pour chaque capteur	74
Tableau 5.5	Erreurs (A_{RMS}) de l'installation 1 pour 5 tests sans LMS	79
Tableau 5.6	Erreurs (A_{RMS}) de l'installation 1 pour 5 tests avec LMS	79
Tableau 5.7	Erreurs (A_{RMS}) de l'installation 2 pour 5 tests sans LMS	83
Tableau 5.8	Erreurs (A_{RMS}) de l'installation 2 pour 5 tests avec LMS	83
Tableau 5.9	Erreurs (A_{RMS}) de l'installation 1 pour 5 tests sans FastICA+LMS	90
Tableau 5.10	Erreurs (A_{RMS}) de l'installation 1 pour 5 tests avec FastICA+LMS	90
Tableau 5.11	Erreurs (A_{RMS}) de l'installation 2 pour 5 tests sans FastICA+LMS	95
Tableau 5.12	Erreurs (A_{RMS}) de l'installation 2 pour 5 tests avec FastICA+LMS	95

LISTE DES FIGURES

	Page
Figure 1.1	Système de mesure de la consommation électrique résidentielle6
Figure 1.2	Fonctionnement d'un capteur à effet Hall7
Figure 1.3	Fonctionnement d'un transformateur de courant8
Figure 1.4	Capteur transformateur de courant.....9
Figure 1.5	Circuit imprimé du capteur de courant10
Figure 1.6	Le capteur de courant dans son boîtier10
Figure 1.7	Capteurs de courant installés dans un panneau de disjoncteurs.....11
Figure 1.8	Modèle du système15
Figure 2.1	Représentation visuelle de « l'effet cocktail party ».....26
Figure 3.1	Schéma de calibration avec l'algorithme LMS44
Figure 3.2	Exemple de 3 signaux réels ayant une distribution uniforme45
Figure 3.3	Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation47
Figure 3.4	Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 1000 simulations.....47
Figure 3.5	Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0,5$ pour 1 simulation48
Figure 3.6	Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations.....49
Figure 3.7	Évolution des coefficients du filtre pour 30 capteurs avec $\sigma_n = 0$ pour 1 simulation50
Figure 3.8	Erreur du courant total (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0$ pour 1000 simulations.....51
Figure 3.9	Évolution des coefficients du filtre pour 30 capteurs avec $\sigma_n = 0,5$ pour 1 simulation52

Figure 3.10	Erreur du courant total (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations.....	52
Figure 3.11	Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec $\sigma_n = 0$ pour 1000 simulations pour chaque cas.....	53
Figure 3.12	Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations pour chaque cas.....	54
Figure 3.13	Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation incluant de la diaphonie.....	56
Figure 3.14	Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation incluant de la diaphonie.....	57
Figure 4.1	Schéma des étapes à suivre pour FastICA.....	61
Figure 4.2	Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec diaphonie et avec $\sigma_n = 0$ et $\sigma_n = 0,5$ pour 100 simulations pour chaque cas.....	67
Figure 5.1	Schéma d'une installation typique.....	70
Figure 5.2	Installation 1.....	71
Figure 5.3	Installation 2.....	72
Figure 5.4	Pince multimètre TRMS AC Fluke 365.....	73
Figure 5.5	Exemple des signaux pour l'installation 1 avec LMS.....	77
Figure 5.6	Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS de l'installation 1 pour 5 tests.....	78
Figure 5.7	Exemple des signaux pour l'installation 2 avec LMS.....	81
Figure 5.8	Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS de l'installation 2 pour 5 tests.....	82
Figure 5.9	Exemple de la distribution des données provenant des capteurs HES pour l'installation 1.....	86
Figure 5.10	Exemple des signaux pour l'installation 1 avec FastICA et LMS incluant l'effet de la diaphonie.....	88
Figure 5.11	Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS après FastICA de l'installation 1 pour 5 tests.....	89

Figure 5.12	Exemple de la distribution des données provenant des capteurs HES pour l'installation 2	91
Figure 5.13	Exemple des signaux pour l'installation 2 avec FastICA et LMS incluant l'effet de la diaphonie	93
Figure 5.14	Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS après FastICA de l'installation 2 pour 5 tests.....	94

Rapport-Gratuit.com

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ADC	Convertisseur analogique numérique (<i>Analog to digital converter</i>)
AMUSE	Algorithme pour l'extraction de multiples signaux inconnus (<i>algorithm for multiple unknown signal extraction</i>)
BSS	Séparation aveugle de sources (<i>blind source separation</i>)
CLT	Théorème de la limite centrale (<i>central limit theorem</i>)
CT	Transformateur de courant (<i>current transformer</i>)
EVD	Décomposition en valeurs singulières (<i>eigenvalue decomposition</i>)
FastICA	Analyse en composantes indépendantes rapide (<i>fast independent component analysis</i>)
HES	Capteur à effet Hall (<i>hall effect sensor</i>)
HOS	Statistiques d'ordre supérieur (<i>higher order statistics</i>)
IC	Composantes indépendantes (<i>independent component</i>)
ICA	Analyse en composantes indépendantes (<i>independent component analysis</i>)
IEEE	Institute of Electrical and Electronics Engineers
LMS	Algorithme des moindres carrés (<i>least mean square</i>)
NLMS	Algorithme des moindres carrés normalisé (<i>normalized least mean square</i>)
PDF	Fonction de densité de probabilité (<i>probability density function</i>)
PWM	Modulation de largeur d'impulsion (<i>pulse width modulation</i>)
RMS	Valeur efficace (<i>root mean square</i>)
SOBI	Algorithme pour l'identification aveugle de deuxième ordre (<i>second order blind identification</i>)
SOS	Statistiques de deuxième ordre (<i>second order statistics</i>)
UCT	Unité centrale de traitement

XX

WSN Réseaux de capteurs sans fil (*wireless sensor network*)

LISTE DES SYMBOLES ET UNITÉS DE MESURE

α	Vecteur des gains inconnus des capteurs
α_{nn}	Coefficients principaux des capteurs pour la matrice de mélange
α_{ij}	Coefficients de diaphonie pour la matrice de mélange
A	Ampère
A	Matrice de mélange
\tilde{A}	Nouvelle matrice de mélange
A_{RMS}	Ampères RMS
cosh()	Cosinus hyperbolique
$\nabla(n)$	Vecteur gradient
$\hat{\nabla}(n)$	Estimé instantané du vecteur gradient
$d(n)$	Signal désiré
D	Matrice diagonale des vecteurs propres
det()	Le déterminant
$\eta(n)$	Vecteur de bruit
$\eta_{\text{CT}}(n)$	Vecteur de bruit du CT
$e(n)$	Signal d'erreur
E	Matrice orthogonale des vecteurs propres
$E\{ \}$	Espérance mathématique
exp()	Fonction exponentielle
$G_n()$	Fonctions pour l'entropie négative
$g_n()$	Dérivées des fonctions G pour l'entropie négative

Hz	Hertz
I	Matrice identité
$I(n)$	Vecteur des courants réels
$\hat{I}(n)$	Vecteur des courants estimés
$I_m(n)$	Vecteur des courants mesurés
I_{m_cross}	Vecteur des courants mesurés par les capteurs avec l'ajout de diaphonie
$I_T(n)$	Courant total précis
K	Nombre de capteurs
λ_{max}	Plus grande valeur propre de la matrice de corrélation
log()	Logarithme
μ	Pas d'adaptation de l'algorithme LMS
\hat{N}	Le nombre d'échantillons utilisés pour trouver les \hat{w} finaux moyens
P	Matrice de permutation
σ_g	Écart-type du gain
σ_n	Écart-type du bruit
σ_{gp}	Écart-type des gains principaux pour la matrice de mélange
s	Vecteur des ICs (aussi appelé vecteur des sources)
s_i	ICs (aussi appelés sources)
T	Nombre d'échantillons
tanh()	Tangente hyperbolique
tma_{ratio}	Grandeur de la fenêtre utilisée pour la moyenne glissante triangulaire
V	Voltage
V	Matrice de blanchiment

\mathbf{w}	Vecteur de séparation
W	Watt
\mathbf{W}	Matrice de séparation
$\hat{\mathbf{w}}(n)$	Vecteur des coefficients du filtre
$\mathbf{x}(n)$	Vecteur d'entrée du filtre
x_i	Variabes d'observations
$y(n)$	Signal de sortie du filtre
\mathbf{z}	Vecteur des données d'observations blanchies

INTRODUCTION

La popularité grandissante des réseaux de capteurs sans fil (WSN, *wireless sensor network*) contribue à leur utilisation dans plusieurs domaines tels que les télécommunications, la santé, l'environnement, le *smart grid*, etc. Un système utilisant un WSN afin de mesurer la consommation énergétique résidentielle a été caractérisé et amélioré dans Levasseur (2013) et a aussi fait l'objet d'un article dans Levasseur et al. (2012). Ce système est composé de plusieurs capteurs à effet Hall (HES, *hall effect sensor*) à faible consommation d'énergie servant à mesurer le courant traversant chacun des disjoncteurs dans le panneau de distribution principale d'une résidence. La différence principale de ce système par rapport à d'autres tel que Powerhouse Dynamics (2014) et Current Cost (2014) concerne l'utilisation de HES au lieu des transformateurs traditionnels de courants (CT, *current transformer*). Les HES ont été choisis pour leur coût d'achat moins élevé que les CTs et aussi pour leur taille très petite qui facilite grandement l'installation dans un panneau de disjoncteurs. Par contre, les HES n'ont pas le même niveau de précision que les CTs puisque ce capteur est sensible à la distance entre celui-ci et le fil qu'il mesure, ceci cause des variations inconnues fixes du gain entraînant une mise à l'échelle du courant mesuré. Contrairement aux CTs, les HES sont aussi sujets aux problèmes de diaphonie (*crosstalk*) entre les capteurs. Pour que ce système reste compétitif sur le marché, il est primordial qu'il offre un niveau de précision adéquat afin de fournir des mesures de consommation d'énergie représentatives de la réalité. Puisque la calibration manuelle de chacun des capteurs prend beaucoup de temps, qu'il y a souvent de nombreux capteurs pour chaque système et que les capteurs ne restent pas calibrés indéfiniment, une méthode automatique de calibration en continu serait avantageuse.

Dans le cadre de ce projet, deux algorithmes numériques sont testés pour faire la calibration du WSN afin d'améliorer la précision des mesures. Pour ce faire, un CT de très grande précision est ajouté au système afin de mesurer le courant total du panneau de disjoncteurs. Une mesure de ce courant total précis est utilisée comme signal de référence pour la calibration des capteurs. Le premier algorithme numérique est l'algorithme des moindres carrés (LMS, *least mean square*) qui est un algorithme simple utilisé afin de faire la

calibration des capteurs sans présence de diaphonie. Le second algorithme, l'analyse en composantes indépendantes rapide (FastICA, *fast independent component analysis*), est plus complexe, mais en mesure de réduire le problème de diaphonie. Les deux algorithmes sont évalués en simulation et en milieu expérimental.

Le premier chapitre montre une vue d'ensemble du système de mesure de la consommation d'énergie résidentielle, présente la problématique de ce projet ainsi que les solutions proposées et les objectifs à atteindre. Le système présenté est principalement constitué de HES, d'un CT très précis pour la mesure du courant total et d'une unité centrale de traitement (UCT) utilisé pour le traitement et la calibration des mesures. La problématique principale est reliée aux problèmes de gain et de diaphonie des capteurs dans le système. Les solutions proposées sont la calibration avec LMS sans diaphonie et avec la combinaison de FastICA et LMS lorsqu'il y a présence de diaphonie.

Avant la présentation des résultats, le chapitre 2 présente une brève revue de la littérature dans le domaine des WSNs ainsi que la théorie reliée aux deux algorithmes numériques.

Le chapitre 3 fait un court retour sur la théorie de l'algorithme LMS en l'adaptant au cas présent et présente ensuite les résultats de simulations obtenus avec cet algorithme. La convergence des coefficients du filtre LMS ainsi que de l'erreur totale pour 3 et 30 capteurs est présentée en ajoutant ou non du bruit aux mesures. Ensuite, l'erreur moyenne par capteur est présentée à la fin de la section des résultats pour 30 capteurs. Ce chapitre est conclu en testant l'algorithme LMS en présence de diaphonie afin de caractériser le niveau d'erreur.

Le quatrième chapitre fait un bref retour sur la théorie de FastICA appliquée au cas présent et les résultats de simulation avec diaphonie en utilisant cet algorithme sont présentés. Un retour sur la théorie explique entre autres pourquoi la combinaison de FastICA et LMS est nécessaire afin d'extraire la diaphonie et de bien calibrer l'amplitude des signaux. Les résultats des erreurs obtenues sont présentés pour 3 et 30 capteurs avec et sans ajout de bruit

aux mesures. L'erreur moyenne par capteur en utilisant la combinaison de FastICA et LMS est présentée à la fin de la section.

Le chapitre 5 présente les deux installations qui sont mises en place afin d'effectuer les tests expérimentaux ainsi que les résultats des tests expérimentaux en utilisant chacun des algorithmes. La première installation est constituée d'une cafetière, d'un fusil à air chaud et d'un purificateur d'air. La seconde est composée d'une cafetière, d'un fusil à air chaud et de 6 plinthes chauffantes branchées en parallèle et contrôlées par un thermostat électronique. Afin de pouvoir calculer l'erreur, un CT précis ainsi qu'un HES sont utilisés pour la mesure de chaque appareil et le total des courants mesurés par les CTs est utilisé comme référence pour l'algorithme LMS. Les résultats des tests avec l'algorithme LMS pour l'installation 1 et 2 en ayant les capteurs éloignés les uns des autres, soit sans diaphonie, sont présentés et les résultats avec diaphonie pour l'installation 1 et 2 avec la combinaison de FastICA et LMS sont ensuite montrés. Puisqu'aucune diaphonie n'a été détectée en rapprochant les capteurs en laboratoire, de la diaphonie a dû être ajoutée en simulation aux mesures expérimentales. Par contre, de la diaphonie a été détectée lors d'installations à l'intérieur de résidences dans le passé.

La conclusion de ce projet met en évidence les particularités de chaque algorithme et fait un retour sur les objectifs de départ. La section des recommandations discute des points importants à prendre en compte avant l'utilisation de chacun des algorithmes en pratique et de proposer des algorithmes alternatifs à ceux présentés qui pourraient être testés dans des travaux futurs.

Contributions

Les contributions suivantes ont été apportées à ce projet :

- utilisation innovatrice de l'algorithme LMS (autre que pour le filtrage) afin de faire la calibration des mesures d'un système de capteurs sans fil mesurant la consommation énergétique résidentielle sans présence de diaphonie;
- utilisation de la combinaison de FastICA et de LMS afin de résoudre le problème de diaphonie et de faire la calibration des mesures;
- calibration des capteurs en simulation avec ces deux algorithmes;
- calibration expérimentale des capteurs avec ces deux algorithmes;
- publication d'un article intitulé « Auto-calibration of Hall effect sensors for home energy consumption monitoring » dans le journal Electronics Letters en 2014 (Beaufort Samson et al., 2014);
- présentation d'une affiche sur la calibration des capteurs avec LMS pour le concours d'affiches portant sur les projets de recherche des étudiants aux cycles supérieurs de l'ÉTS;
- débogage de l'UCT pour la prise de mesures.

CHAPITRE 1

DESCRIPTION DU SYSTÈME ET DE LA PROBLÉMATIQUE

1.1 Introduction

Ce mémoire traite principalement de la calibration de capteurs sans fil mesurant la consommation électrique. Avant d'aller plus en profondeur dans le sujet principal du mémoire, ce chapitre présente sommairement le système étudié, plus de détails étant disponibles dans Levasseur (2013) et Levasseur et al. (2012). Une mise en contexte de la problématique, le modèle mathématique du système, les objectifs à atteindre dans ce mémoire, les spécifications désirées de l'algorithme ainsi qu'une brève introduction des solutions proposées afin de résoudre cette problématique sont également présentés à la fin de ce chapitre.

1.2 Survol du système mesurant la consommation énergétique résidentielle

Le système étudié, mesurant la consommation électrique résidentielle, est composé principalement de divers capteurs de courant et d'une unité centrale de traitement (UCT) tel que montré à la figure 1.1. Un capteur à effet Hall (HES, *hall effect sensor*) (détaillé à la section 1.3.1) est installé sur chaque fil relié à un disjoncteur dans le panneau de distribution électrique. Un capteur de courant de type transformateur de courant (CT, *current transformer*) (voir section 1.3.2) sert à mesurer précisément la consommation totale de courant de tous les disjoncteurs dans le panneau. Les HES transmettent leurs mesures à l'UCT par radiofréquence selon la norme IEEE 802.15.4 (IEEE STANDARDS ASSOCIATION, 2003). Celle-ci se charge de calculer la valeur RMS du courant pour chaque capteur et d'améliorer la précision des mesures en utilisant les algorithmes présentés aux chapitres 3 et 4.

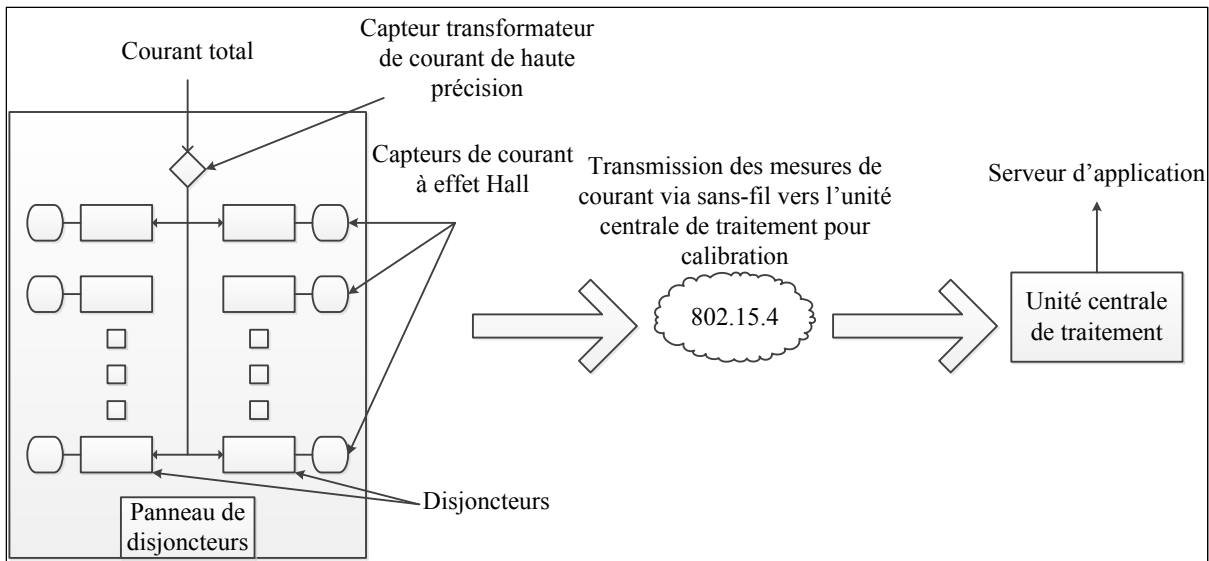


Figure 1.1 Système de mesure de la consommation électrique résidentielle
Adaptée de Levasseur et al. (2012)

1.3 Types de capteurs

Les capteurs de courant constituent les éléments de base du système mesurant la consommation électrique. Le type de capteur qui a été choisi afin de mesurer le courant passant dans chaque fil relié à un disjoncteur dans ce système est un HES. Les raisons majeures reliées à ce choix sont la taille, le coût et la facilité d'installation de ce type de capteur. Un capteur de type CT est utilisé pour chaque panneau de disjoncteurs afin de calculer précisément le courant total circulant dans celui-ci.

1.3.1 Description du HES

Comme le nom l'indique, ce type de capteur est basé sur l'effet Hall qui génère une tension proportionnelle au champ magnétique le traversant. Plus précisément, si un courant circule dans une plaque conductrice perpendiculairement à un champ magnétique qui est appliqué au travers de cette plaque, une différence de potentiel appelée la tension de Hall est générée. Cette différence de potentiel est perpendiculaire au champ magnétique ainsi qu'au courant circulant dans la plaque. La figure 1.2 illustre ce phénomène. Pour plus de détails sur les HES, voir Ramsden (2006).

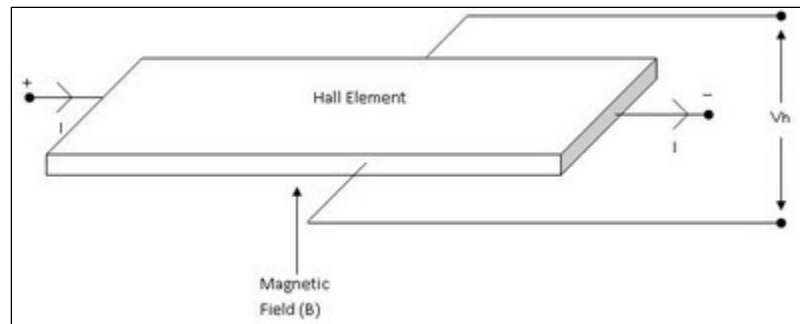


Figure 1.2 Fonctionnement d'un capteur à effet Hall
Tirée de Sensors and Actuators (2013)

Contrairement aux CT, les HES peuvent être fabriqués en version miniaturisée et faire partie de circuits intégrés. Ce type de capteur est donc souvent vendu en faisant partie d'un circuit intégré qui est fabriqué à très grande échelle, ce qui permet de réduire le coût de production. Un autre avantage du HES, qui est une conséquence directe de sa taille très petite, est la facilité d'installation de ce capteur dans un panneau de disjoncteurs, ceux-ci étant disposés très près les uns des autres. La très petite taille des HES, son coût d'achat ainsi que sa facilité d'installation ont fait opter pour ce type de capteur par rapport à un CT qui est beaucoup plus gros et plus coûteux.

Le principal inconvénient du HES est qu'il capte tous les champs magnétiques perpendiculaires à sa plaque métallique, ce qui vient donc fausser les mesures prises par les HES lorsque plusieurs capteurs sont très près les uns des autres; comme c'est le cas dans un panneau de disjoncteurs. Ce problème, appelé diaphonie, est expliqué plus précisément à la section 1.6.2. Le second inconvénient du HES est que sa sensibilité dépend de la proximité de celui-ci au fil, ce qui cause des incertitudes sur le gain du capteur lorsque cette distance est variable.

1.3.2 Description du CT

Un CT est composé d'un enroulement autour d'un noyau magnétique. Le courant circulant dans le fil crée un champ magnétique dans le noyau, ce qui génère un courant dans l'enroulement où N est le nombre de spires, comme le démontre la figure 1.3.

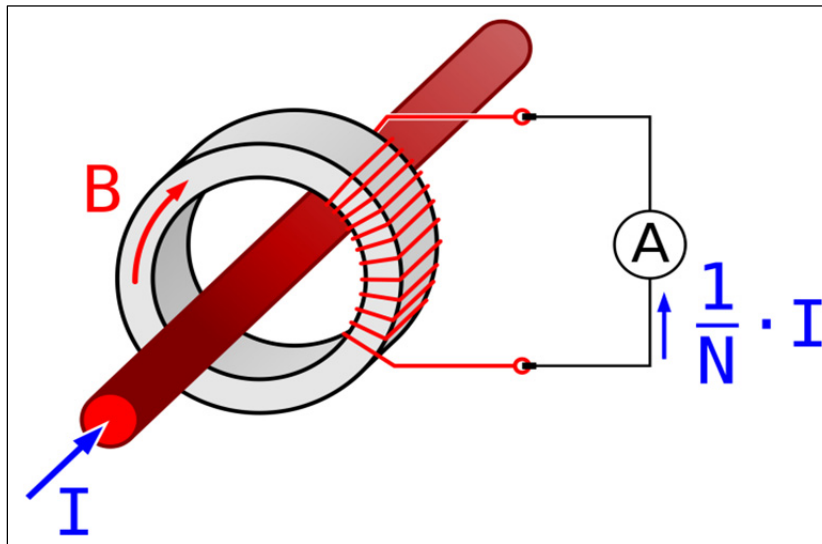


Figure 1.3 Fonctionnement d'un transformateur de courant
Tirée de Biezl (2009)

En comparant le CT au HES, on peut constater que les qualités du HES sont les défauts du CT et vice-versa. La qualité principale du CT est sa précision lors de la prise de mesure. Puisque le CT fait le tour complet du fil, il capture la totalité du champ magnétique et fournit donc une mesure bien plus précise qu'un HES. Une des raisons majeures de cette précision est que la mesure prise par le CT n'est pas influencée par les champs magnétiques avoisinants puisqu'il capture seulement ceux générés par le fil à l'intérieur de son noyau magnétique, tel que représenté à la figure 1.3. Ce capteur a cependant une taille typiquement trop grande pour être installé à la sortie de chaque disjoncteur, ainsi qu'un coût d'achat élevé par rapport à un HES.

Dans le système de mesure de la consommation électrique, un CT est utilisé afin de mesurer précisément le courant total circulant à l'intérieur du panneau de disjoncteurs. Cette information sert pour la calibration des capteurs de courant avec l'algorithme LMS qui est expliqué au chapitre 2. Le modèle de CT utilisé dans ce système est le SCT-023-400UL (EKM Metering Inc. (2013c)), montré à la figure 1.4. La précision de ce type de capteur est discutée à la section 5.2.2.

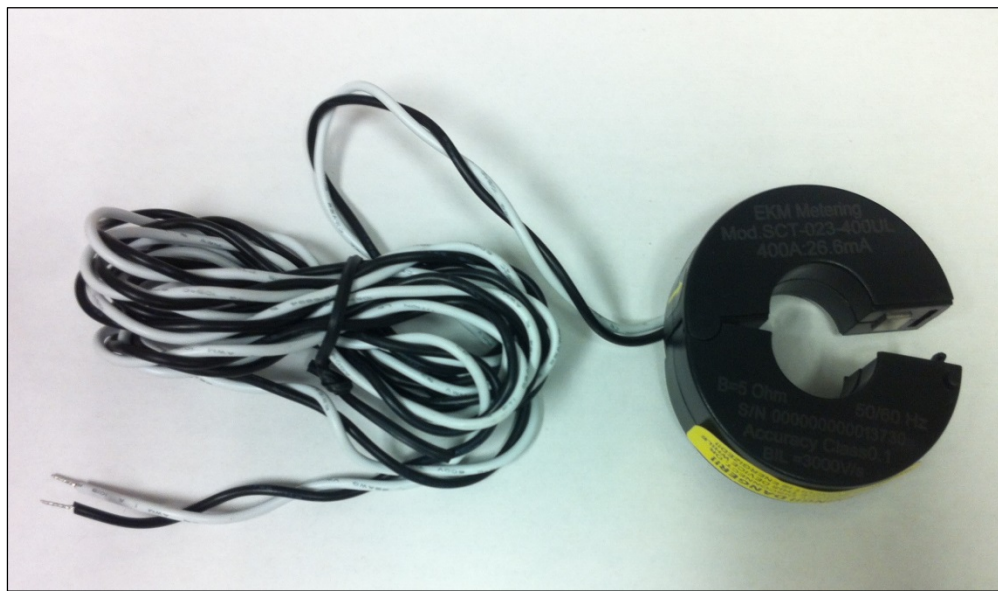


Figure 1.4 Capteur transformateur de courant

1.4 Implémentation à l'intérieur de capteurs de courant

Les HES font partie des capteurs de courant sans fil dans le système étudié de mesure de la consommation électrique. Les capteurs de courant sont donc munis d'un HES, d'une pile AAA, d'un transmetteur radiofréquence et d'un microcontrôleur. Ce dernier sert à faire la gestion des données, l'économie d'énergie et possède aussi un convertisseur analogique numérique (ADC, *analog to digital converter*) permettant de convertir la sortie analogique du HES en un résultat numérique. Le modèle de capteur à effet Hall utilisé dans ce système est le A1393SEHLT-T (Allegro Microsystems Inc. (2011)). Le circuit imprimé du capteur de courant est introduit à l'intérieur d'un boîtier de plastique et un blindage magnétique est utilisé afin de réduire la diaphonie. La figure 1.5 montre le circuit imprimé ainsi que les

composants du capteur de courant tandis que la figure 1.6 montre le capteur de courant inséré dans son boîtier de plastique avec le blindage magnétique. L'installation de capteurs de courants dans un panneau de disjoncteurs est montrée à la figure 1.7.

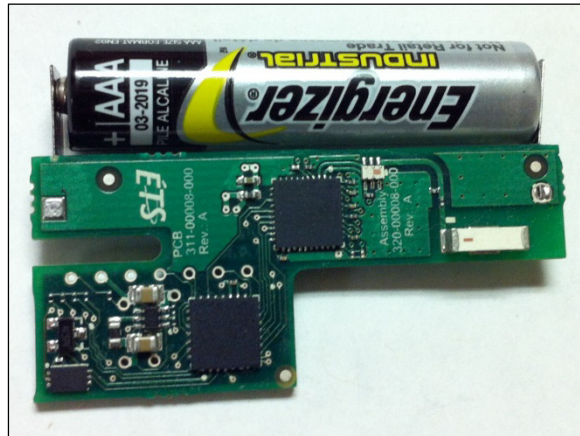


Figure 1.5 Circuit imprimé du capteur de courant



Figure 1.6 Le capteur de courant dans son boîtier



Figure 1.7 Capteurs de courant installés dans un panneau de disjoncteurs

1.5 Traitement et transmission des données

Une autre partie importante du système est le traitement et la transmission des mesures prises par les capteurs de courant. Cette section explique brièvement le traitement mathématique relié aux mesures autant par le capteur de courant que par l'UCT, ainsi que la fréquence à laquelle les mesures sont enregistrées et transmises vers l'UCT.

1.5.1 Traitement des données effectué par les capteurs et l'UCT

Le microcontrôleur, situé sur le circuit imprimé de chacun des capteurs de courant, convertit les valeurs analogiques fournies par le HES (une tension proportionnelle au champ magnétique mesuré) en valeurs numériques grâce à l'ADC à l'intérieur du microcontrôleur. Afin d'économiser l'énergie de la batterie du capteur de courant, seules les opérations mathématiques les plus simples sont effectuées par le microcontrôleur tandis que l'UCT se charge des opérations plus complexes puisque celui-ci n'est pas alimenté par une batterie. Le courant efficace (RMS, *root mean square*) est calculé selon :

$$I_{RMS} = \sqrt{\frac{1}{N} \left(\sum_{n=1}^N I_n^2 \right)} \quad (1.1)$$

où I_n est le courant à l'échantillon n et N est le nombre d'échantillons utilisés pour le calcul du courant RMS pour chaque transmission.

Par contre, puisque la division ainsi que la racine carrée dans ce calcul sont des opérations mathématiques qui consomment beaucoup d'énergie, elles sont effectuées par l'UCT. Le microcontrôleur du capteur de courant s'occupe donc seulement de calculer la somme des carrés des mesures sur N échantillons et transmet cette valeur par radiofréquence à l'UCT.

1.5.2 Fréquence d'enregistrement et de transmission des données par les capteurs

Plusieurs scénarios possibles de prises de mesures ont été considérés avant le choix final. Plus de détails sur chacun de ces scénarios sont disponibles dans Levasseur (2013). Le scénario final qui a été retenu est de sous échantillonner le signal à une fréquence de 3,08 Hz afin d'avoir un compromis entre l'économie de la pile et la détection des variations dans le courant. Le capteur de courant effectue une transmission des mesures par radiofréquence dès que 18 échantillons sont recueillis. L'UCT effectue l'équation (1.1) sur les 18 échantillons reçus et enregistre la valeur obtenue. Puisque la fréquence d'échantillonnage est de 3,08 Hz

et qu'une mesure est enregistrée à tous les 18 échantillons, une mesure de courant RMS est obtenue toutes les 5,844 secondes. Afin de maximiser la durée de vie de la batterie de chaque capteur, la norme 802.15.4 (IEEE STANDARDS ASSOCIATION, 2003) est utilisée due à sa faible consommation d'énergie lors de transmissions sans fil.

1.6 Mise en contexte de la problématique

Plusieurs facteurs viennent affecter la précision des mesures des HES dans ce système, les plus importants sont le gain inconnu de chaque capteur ainsi que la diaphonie présente entre les HES.

1.6.1 Problème de gain des capteurs

Puisque la sensibilité des HES dépend de leur proximité par rapport au fil, ceci vient affecter le gain du capteur. Ce gain produit une différence entre la mesure réelle du courant et la mesure fournie par le capteur. La relation entre la mesure réelle et la mesure enregistrée par le capteur est donc représentée comme une relation entrée-sortie variant d'un capteur à un autre. Cette relation peut être linéaire ou non linéaire (Wang, 2009).

Comme mentionné dans Wang (2009), il existe trois raisons principales pour lesquelles les données mesurées ne correspondent pas toujours aux données réelles. Ces trois raisons sont la faible qualité des capteurs, la manipulation des capteurs lors de l'installation et le vieillissement des composants à l'intérieur de ceux-ci. Dans le cas du présent système, la raison principale est que les HES ont une précision limitée de par leur nature, car ils dépendent de la position du capteur par rapport au fil.

1.6.2 Problème de diaphonie entre les capteurs

Le second problème majeur affectant le système étudié est la diaphonie. Étant donné qu'un HES mesure tous les champs magnétiques traversant sa plaque conductrice, et puisque les HES sont très près les uns des autres lorsqu'ils sont installés dans un panneau de disjoncteurs,

chacun des HES va aussi mesurer une partie du courant passant dans les fils voisins. Le blindage magnétique installé sur le boîtier de chaque capteur de courant réduit l'effet des champs magnétiques externes, mais ne les annule pas complètement.

1.7 Modélisation mathématique du système

Un modèle mathématique du système de mesure de la consommation électrique est proposé à la figure 1.8. La position du capteur par rapport au fil vient affecter la précision des mesures prises par le HES et se traduit donc en un gain inconnu. On considère que ces gains inconnus ne varient pas dans le temps, car ils sont principalement basés sur des paramètres fixes comme l'état du capteur et sa position par rapport au fil lors de l'installation. Le vieillissement ainsi que les variations de température des capteurs ne sont pas considérés directement, mais les algorithmes numériques peuvent s'adapter à ces changements. Sans considérer (pour l'instant) la diaphonie, le système est modélisé selon l'équation suivante :

$$\mathbf{I}_m(n) = \mathbf{I}(n) .* \boldsymbol{\alpha} + \boldsymbol{\eta}(n) \quad (1.2)$$

où $\mathbf{I}_m(n)$ est le vecteur des courants mesurés par les HES, $\mathbf{I}(n)$ est le vecteur des courants réels (sources), $\boldsymbol{\alpha}$ est le vecteur des gains inconnus des capteurs, $\boldsymbol{\eta}(n)$ est le vecteur de bruit et n représente le numéro de l'échantillon où $n = 0, 1, 2, \dots, N-1$. L'opérateur $.*$ est le multiplicateur élément par élément. Chaque vecteur contient K éléments, où K est le nombre de capteurs à l'intérieur du système.

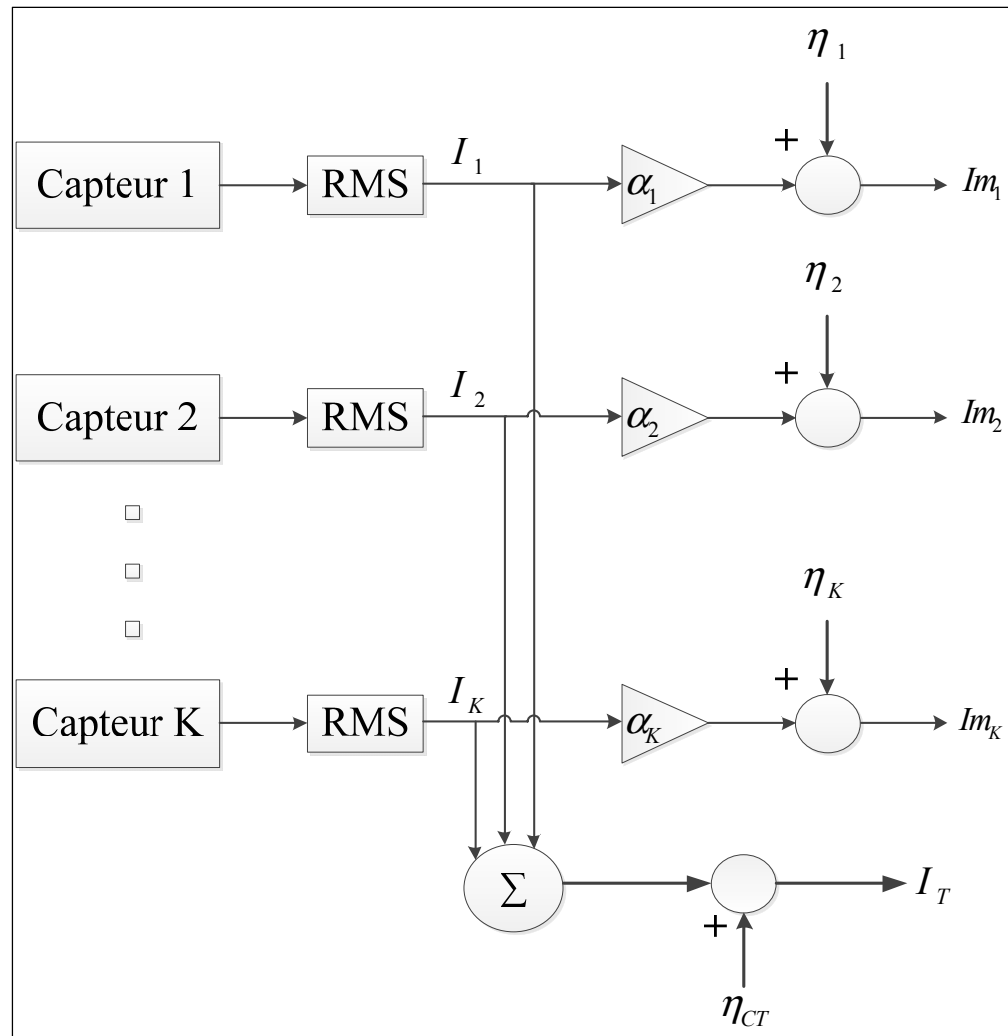


Figure 1.8 Modèle du système

Le rôle du CT dans ce système est de mesurer très précisément la somme totale du courant (la somme des courants passant dans chaque disjoncteur) afin d'être utilisé comme signal de référence. On peut modéliser le signal à la sortie du CT selon :

$$I_T(n) = \sum_{k=1}^K I_k(n) + \eta_{CT}(n) \quad (1.3)$$

où $I_T(n)$ est le courant total précis mesuré par le CT, $I_k(n)$ est le courant précis traversant chaque disjoncteur et $\eta_{CT}(n)$ est le bruit affectant le CT.

Puisque le capteur CT a une plus grande sensibilité que les capteurs HES, le rapport signal à bruit d'un CT est par conséquent beaucoup plus élevé.

1.8 Objectifs du mémoire et caractéristiques recherchées de l'algorithme

Plusieurs objectifs doivent être atteints dans ce mémoire afin de répondre à la problématique expliquée à la section 1.6. Par la suite, les spécifications attendues de l'algorithme sont mentionnées.

Les objectifs du mémoire sont :

- faire la conception d'un algorithme de calibration automatique des capteurs afin de réduire l'imprécision des mesures ainsi que l'effet de la diaphonie;
- valider cet algorithme avec des signaux simulés dans Matlab[®];
- valider l'algorithme expérimentalement à l'aide de vraies mesures.

Le premier objectif consiste à concevoir un algorithme qui est en mesure de trouver le vecteur de gains inconnus des capteurs α de l'équation (1.2) afin de réduire l'erreur entre les valeurs réelles des courants et les valeurs mesurées par les capteurs. L'algorithme doit aussi être apte à réduire le problème de diaphonie. L'erreur à atteindre doit être $< 1 A_{RMS}$, ce qui correspond à environ 120W pour une alimentation de 120V RMS. Cette précision est suffisante, considérant que le système vise avant tout à donner un portrait de la consommation afin d'aider l'utilisateur à améliorer ses habitudes de consommation énergétique. Ce niveau de précision est suffisant afin d'identifier ce qui consomme le plus dans une maison.

Le second objectif consiste à faire la validation de cet algorithme dans l'environnement logiciel Matlab[®]. La simulation de l'algorithme LMS afin de déterminer les gains des capteurs est effectuée au chapitre 3 tandis que la simulation de la combinaison de FastICA et LMS afin de s'attaquer au problème de diaphonie est effectuée au chapitre 4.

Pour le dernier objectif, l'algorithme doit être validé expérimentalement afin de calibrer les mesures de vrais capteurs. La calibration des mesures expérimentales avec et sans diaphonie est effectuée au chapitre 5 grâce à LMS ainsi qu'à la combinaison de FastICA et LMS respectivement.

De plus, il est important que l'algorithme soit adaptatif et puisse corriger l'erreur en continu puisque le système de mesure de la consommation électrique restera en fonction pendant une longue période de temps.

L'algorithme développé doit être relativement simple, car il sera mis en œuvre dans un système embarqué (l'UCT) qui effectue déjà plusieurs tâches et ne doit pas être surchargé.

La convergence de l'algorithme n'a pas à être très rapide, car elle se fera au moment de l'installation du système et celui-ci sera en place pendant plusieurs années. Une convergence plus courte que 5 minutes est acceptable.

1.9 Solutions proposées

Parmi plusieurs algorithmes existants, deux algorithmes ont été retenus puisqu'ils répondaient le mieux aux spécifications présentées à la section précédente. Individuellement, ces algorithmes ne répondent pas à toutes les attentes énumérées précédemment, mais chacun d'eux couvre une partie des caractéristiques recherchées.

1.9.1 Solutions algorithme LMS

L'algorithme LMS (Haykin, 1984) peut être utilisé dans cette application afin de compenser l'incertitude des valeurs de gains α des capteurs. En effet, LMS vise à réduire l'erreur résiduelle en se basant sur la mesure totale fournie par le CT comme signal de référence. L'algorithme doit ajuster les coefficients d'un filtre adaptatif afin de converger vers une erreur résiduelle minimale.

Le principal avantage de l'algorithme LMS est sa simplicité. Il est aussi adaptatif, permettant ainsi de faire la poursuite des variations dues au vieillissement, la température ou à des modifications de l'installation dans le panneau de distribution. Par contre, l'algorithme LMS ne converge pas aussi rapidement que d'autres algorithmes plus récents et ne règle pas le problème de diaphonie. Le principe de fonctionnement du LMS est présenté plus en détail au chapitre 2.

1.9.2 Solution FastICA combinée à l'algorithme LMS

Le FastICA (Hyvärinen, Karhunen et Oja, 2001) est un algorithme de séparation aveugle de sources (BSS, *blind source separation*) utilisant l'analyse en composantes indépendantes (ICA, *independent component analysis*), par exemple dans des applications telles que l'identification de locuteurs différents captés par plusieurs micros (Hyvärinen et Oja, 2000).

FastICA utilise des statistiques d'ordre supérieur (HOS, *higher order statistics*), ce qui signifie que cet algorithme est relativement complexe. De plus, l'algorithme n'est pas adaptatif et requiert une certaine quantité d'échantillons pour être utilisé. FastICA réduit la diaphonie, mais ne garantit pas de conserver le gain des signaux (Hyvärinen, Karhunen et Oja, 2001), c'est pourquoi l'algorithme LMS est nécessaire afin de trouver le gain de ceux-ci. Par conséquent, nous proposons l'utilisation de FastICA en complément à l'algorithme LMS afin de réduire l'effet de la diaphonie entre des capteurs voisins.

1.10 Conclusion

Pour débiter, le système de mesure de la consommation électrique résidentielle a été introduit dans ce chapitre. Le système de mesure est muni principalement de HES, d'un capteur CT mesurant le courant total très précisément et d'une UCT faisant le traitement et la calibration des données.

Les deux problèmes majeurs de ce système sont la précision des capteurs HES qui crée un gain inconnu pour chacun d'eux ainsi que la diaphonie entre les capteurs.

Enfin, les objectifs du mémoire et les attentes par rapport à l'algorithme ainsi que les solutions proposées sont présentés. L'algorithme LMS est simple, adaptatif et peut réduire l'erreur résiduelle efficacement, mais n'est pas en mesure de réduire le problème de diaphonie dans le système. L'algorithme FastICA pour sa part diminue le problème de diaphonie, mais ne garantit pas de conserver le gain des signaux. C'est pour cette dernière raison que l'algorithme LMS est combiné à FastICA afin de retrouver l'amplitude des signaux une fois que FastICA a corrigé le problème de diaphonie.

CHAPITRE 2

REVUE DE LA LITTÉRATURE ET THÉORIE

2.1 Introduction

Le chapitre précédent a introduit le système de mesure de la consommation électrique résidentiel ainsi que les algorithmes proposés. Par conséquent, une brève revue de la littérature dans le domaine des WSN et la théorie justifiant les algorithmes proposés peuvent maintenant être exposées. Une plus grande place est accordée à la description de FastICA en raison de sa complexité.

2.2 Revue de la littérature dans le domaine de la calibration de WSN

Il existe plusieurs solutions au problème de calibration des WSN, par exemple, Rui et al. (2010); Takruri, Challa et Yunis (2009); Takruri et al. (2011); Tourancheau et al. (2009); Wang (2009), mais aucune ne s'applique à notre système pour plusieurs raisons.

Les capteurs HES voisins dans notre système n'observent pas toujours des données corrélées comme c'est le cas pour les solutions présentées dans la littérature qui sont basées sur ce principe (Rui et al., 2010; Takruri, Challa et Yunis, 2009; Takruri et al., 2011). En plus d'utiliser la corrélation entre les données, les différentes approches dans Wang (2009) se servent de la mobilité du WSN pour améliorer la calibration, ce qui est impossible dans notre cas puisque le WSN est fixe. L'approche dans Tourancheau et al. (2009) est différente des autres puisqu'une compensation logicielle est effectuée afin d'améliorer la précision des capteurs. Les capteurs dans leur système sont tous calibrés manuellement grâce à une source précise et une régression linéaire est ensuite effectuée à partir des données recueillies afin de compenser l'erreur des capteurs périodiquement. Cette approche n'est pas applicable à notre cas puisque plusieurs capteurs seront installés dans chaque panneau de disjoncteurs (20-30) et une calibration manuelle de chacun d'eux demanderait beaucoup trop de temps. Puisque ces solutions ne répondent pas à nos besoins, de nouvelles doivent être envisagées.

2.3 Théorie du LMS

La théorie de base de l'algorithme LMS tel que présenté dans Haykin (1984) est montrée dans cette section. Les avantages et inconvénients de LMS, l'algorithme du gradient ainsi que l'algorithme LMS sont tout d'abord présentés tandis que l'impact du choix de la grandeur du pas d'adaptation et la convergence de l'algorithme sont présentés par la suite.

2.3.1 Avantages et inconvénients de LMS

L'algorithme LMS minimisant l'erreur quadratique moyenne a été inventé par Bernard Widrow à l'université de Stanford en 1960 et est grandement utilisé dans la littérature depuis ce temps (Haykin, 2003). C'est un algorithme robuste et stable. Comme mentionné dans Haykin (1984), c'est un algorithme simple puisqu'il ne requiert pas le calcul de la fonction de corrélation ni aucune inversion de matrice. Par contre, la vitesse de convergence de LMS est plus lente que les algorithmes adaptatifs plus récents. Dans le cas de notre système, comme mentionné à la section 1.8, le temps de convergence de l'algorithme n'est pas un facteur primordial puisque la convergence des coefficients se fera lors de l'installation du système et que celui-ci sera en place pendant plusieurs années.

2.3.2 Algorithme du gradient

Avant d'introduire le fonctionnement de l'algorithme LMS, l'algorithme du gradient est présenté puisque LMS découle directement de cet algorithme.

L'algorithme du gradient (aussi appelé l'algorithme de la plus forte pente) est un algorithme itératif d'optimisation. Le but de l'algorithme est de trouver le minimum local d'une fonction de coût selon un pas basé sur le gradient négatif de celle-ci. La fonction doit donc avoir seulement un minimum local afin que l'algorithme fonctionne correctement. Afin de trouver l'erreur minimale, l'algorithme apportera des corrections aux coefficients d'un filtre selon la fonction d'erreur suivante :

$$e(n) = d(n) - y(n) = d(n) - \hat{\mathbf{w}}^T(n)\mathbf{x}(n) \quad (2.1)$$

où $e(n)$ est le signal d'erreur, $d(n)$ est le signal désiré, $y(n)$ est le signal de sortie du filtre, $\hat{\mathbf{w}}(n)$ est le vecteur des coefficients du filtre et $\mathbf{x}(n)$ le vecteur d'entrée du filtre.

L'équation régissant l'adaptation des coefficients du filtre selon l'algorithme du gradient ainsi que le vecteur gradient sont représentés comme suit :

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{1}{2}\mu[-\nabla(n)] \quad (2.2)$$

$$\nabla(n) = -2E[e(n)\mathbf{x}(n)] \quad (2.3)$$

où μ est une valeur scalaire qui sert de pas d'adaptation, $\nabla(n)$ est le vecteur gradient et E l'espérance mathématique.

En remplaçant (2.3) dans (2.2), on obtient :

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu E[e(n)\mathbf{x}(n)] \quad (2.4)$$

2.3.3 L'algorithme LMS

La raison principale limitant l'utilisation de l'algorithme du gradient est qu'il requiert la mesure exacte du vecteur gradient à chaque itération. Le problème est qu'il est difficile en pratique d'obtenir la mesure « exacte » du vecteur gradient. L'algorithme LMS nécessite seulement l'utilisation d'estimations instantanées du vecteur gradient en se basant sur les valeurs fournies à l'entrée et sur le signal d'erreur.

L'estimé instantané $\hat{\nabla}(n)$ du vecteur gradient (Haykin, 1984) est donné par :

$$\hat{\mathbf{V}}(n) = -2e(n)\mathbf{x}(n) \quad (2.5)$$

L'équation d'adaptation des coefficients du filtre devient donc :

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{1}{2}\mu[-\hat{\mathbf{V}}(n)] = \hat{\mathbf{w}}(n) + \mu e(n)\mathbf{x}(n) \quad (2.6)$$

Le signal d'erreur ne change pas et reste le même qu'à l'équation (2.1).

2.3.4 Choix de la grandeur du pas d'adaptation et convergence de LMS

Le pas d'adaptation μ est une valeur scalaire et est le seul paramètre ajustable de l'algorithme LMS. Ce paramètre est très important puisqu'il affecte directement la correction apportée aux coefficients du filtre et donc la vitesse de convergence de l'algorithme. Si la valeur choisie de μ est grande, le temps de convergence sera plus rapide, mais l'erreur résiduelle sera plus élevée et instable. Par contre, si la valeur de μ choisie est très petite, la convergence sera lente, mais l'erreur résiduelle sera plus petite.

Afin que l'algorithme reste stable et converge, la valeur du pas d'adaptation doit être régie selon :

$$0 < \mu < \frac{2}{\lambda_{max}} \quad (2.7)$$

où λ_{max} est la plus grande valeur propre de la matrice de corrélation $R = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$.

2.4 Théorie de l'ICA

La séparation aveugle de sources (BSS, *blind source separation*) (Cichocki, 2004) consiste à séparer des signaux sources à partir de signaux mélangés. C'est un concept plus large qui englobe l'ICA sans toutefois se limiter seulement à cette catégorie d'algorithmes. Plusieurs

algorithmes comme l'algorithme pour l'extraction de multiples signaux inconnus (AMUSE, *algorithm for multiple unknown signal extraction*) et l'algorithme pour l'identification aveugle de deuxième ordre (SOBI, *second order blind identification*) font partie du BSS sans être de l'ICA (Cichocki, 2004). Ces algorithmes utilisent des statistiques de deuxième ordre (SOS, *second order statistics*) et n'ont pas de restrictions au niveau de l'indépendance des composantes et de la distribution des composantes indépendantes (ICs, *independent component*) (aussi appelées sources). Pour sa part, l'ICA utilise des HOS et impose que les ICs aient une distribution non gaussienne et qu'elles soient statistiquement indépendantes les unes des autres. Par contre, les composantes estimées par AMUSE et SOBI ne sont pas nécessairement indépendantes contrairement à ICA, mais seulement spatio temporellement décorrélés.

La théorie de l'ICA est tout d'abord présentée suivie par la présentation de l'algorithme FastICA. Toute la théorie et les équations sont tirées de Hyvärinen (1999); Hyvärinen et Oja (2000); Hyvärinen, Karhunen et Oja (2001).

2.4.1 L'effet cocktail party

Un des exemples simples représentant une des applications de l'ICA est « l'effet cocktail party » (Hyvärinen, Karhunen et Oja, 2001). Plusieurs personnes discutent en même temps dans une pièce tandis qu'on cherche à extraire les ICs, qui sont la voix de chaque personne, parmi tous ces signaux mélangés. S'il y a plus d'ICs que d'observations, le modèle est sous-déterminé tandis que si c'est l'inverse le modèle est surdéterminé. La figure 2.1 illustre ce problème.

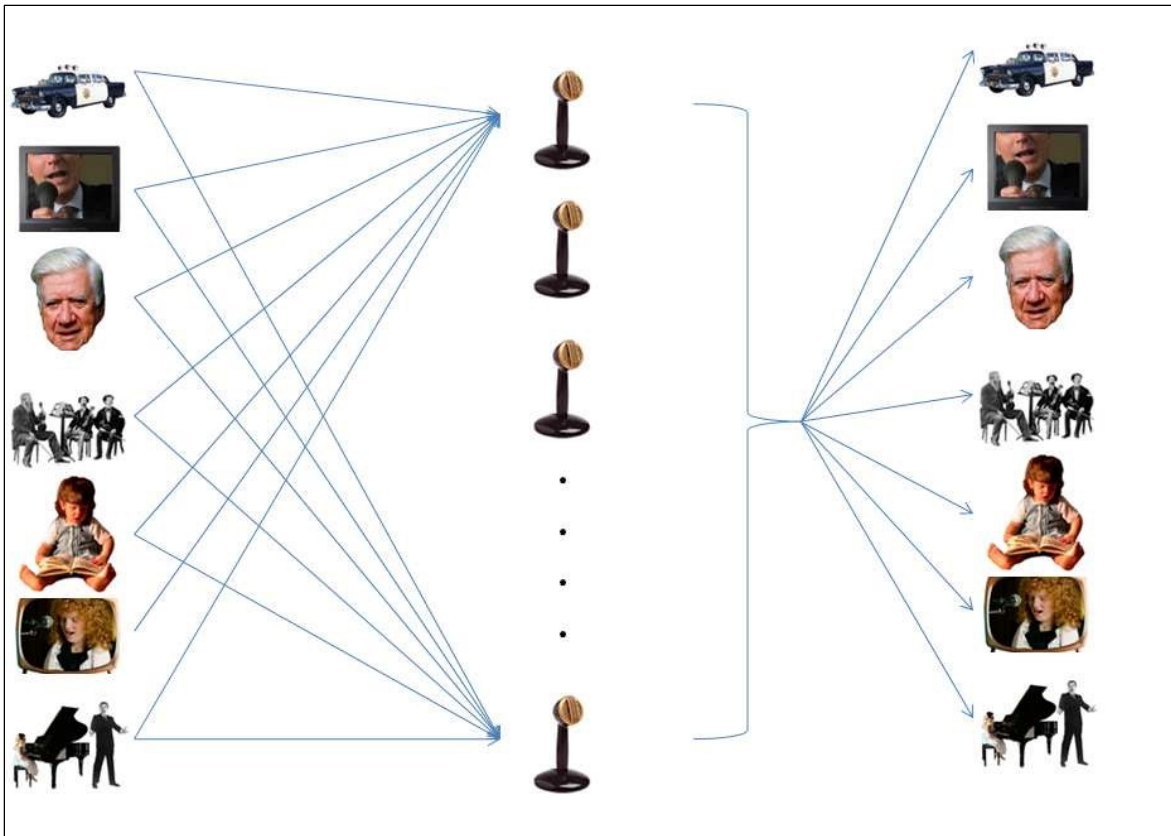


Figure 2.1 Représentation visuelle de « l'effet cocktail party »
Tirée de Trivedi (2010)

2.4.2 Modèle de base de l'ICA

Le modèle d'ICA est principalement constitué d'observations, d'ICs et de coefficients inconnus.

Le modèle de base de l'ICA peut être écrit de la façon suivante :

$$x_i = a_{i1}s_1 + a_{i2}s_2 + \dots + a_{in}s_n \quad (2.8)$$

où $i = 1, \dots, n$ et n est le nombre de variables, x_1, \dots, x_n sont les variables observées, s_1, \dots, s_n sont les ICs et a_{ij} , $i, j = 1, \dots, n$ sont les coefficients de mélange.

Le problème d'ICA est difficile à résoudre puisque seulement les variables d'observations x_i sont connues tandis que les coefficients de mélange a_{ij} et les ICs s_i sont inconnus.

La forme matricielle du modèle d'ICA est souvent plus facile d'utilisation pour plusieurs variables :

$$\mathbf{x} = \mathbf{A}\mathbf{s} \quad (2.9)$$

où \mathbf{x} est le vecteur d'observations, \mathbf{A} est la matrice de mélange contenant les coefficients a_{ij} et \mathbf{s} est le vecteur des ICs.

Ceci est seulement le modèle de base d'ICA et n'inclut aucun bruit dans le modèle qui serait normalement présent lors des mesures. Ce modèle est aussi considéré comme déterminé puisque le nombre d'ICs est égal au nombre d'observations.

2.4.3 Le blanchiment résout la moitié du problème d'ICA

Le blanchiment est une étape importante à effectuer sur les signaux mélangés puisque ce prétraitement des données réduit de moitié le problème d'ICA (Hyvärinen, Karhunen et Oja, 2001). Il est appliqué avant l'utilisation de l'algorithme FastICA. Le blanchiment d'un vecteur est toujours une transformation réalisable contrairement à l'ICA qui a des restrictions fixes devant être respectées avant son utilisation.

Avant le blanchiment d'un vecteur, ses données doivent être centrées afin qu'il ait une moyenne nulle. Ceci est effectué de la façon suivante :

$$\mathbf{x} \leftarrow \mathbf{x} - E\{\mathbf{x}\} \quad (2.10)$$

En centrant les données, la matrice de corrélation sera égale à la matrice de covariance, ce qui simplifie les calculs par la suite.

Une des méthodes afin de blanchir les données est la décomposition en valeurs singulières (EVD, *eigenvalue decomposition*) de la matrice de covariance :

$$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{E}\mathbf{D}\mathbf{E}^T \quad (2.11)$$

où \mathbf{E} est la matrice orthogonale des vecteurs propres de $E\{\mathbf{x}\mathbf{x}^T\}$ et \mathbf{D} est la matrice diagonale de ses valeurs propres.

Une fois EVD effectuée, le vecteur \mathbf{x} peut être blanchi en utilisant la transformation linéaire suivante :

$$\mathbf{z} = \mathbf{E}\mathbf{D}^{-\frac{1}{2}}\mathbf{E}^T \mathbf{x} \quad (2.12)$$

où \mathbf{z} est le vecteur des données d'observations blanchies.

Le blanchiment du vecteur \mathbf{x} rendra ses éléments non corrélés et avec une variance unitaire. Le vecteur \mathbf{x} ayant été blanchi, la matrice de covariance (ainsi que la matrice de corrélation) associée sera donc équivalente à la matrice identité \mathbf{I} :

$$E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{I} \quad (2.13)$$

Si l'on considère le blanchiment qui a été effectué à l'équation (2.12) sur le vecteur d'observations \mathbf{x} comme une matrice de blanchiment :

$$\mathbf{V} = \mathbf{E}\mathbf{D}^{-\frac{1}{2}}\mathbf{E}^T \quad (2.14)$$

où \mathbf{V} est la matrice de blanchiment.

En combinant l'équation (2.12) et (2.14), le vecteur blanchi \mathbf{z} peut être vu comme :

$$\mathbf{z} = \mathbf{V}\mathbf{x} \quad (2.15)$$

Selon le modèle de base d'ICA présenté à l'équation (2.9) et la transformation linéaire par blanchiment de \mathbf{x} à l'équation (2.15), on peut voir que la matrice de mélange \mathbf{A} est transformée en une nouvelle matrice de mélange par le blanchiment :

$$\mathbf{z} = \mathbf{V}\mathbf{A}\mathbf{s} = \tilde{\mathbf{A}}\mathbf{s} \quad (2.16)$$

où $\tilde{\mathbf{A}}$ est la nouvelle matrice de mélange.

Par contre, même si le blanchiment est un prétraitement qui rend les données non corrélées, ce n'est pas suffisant afin d'extraire les ICs. Le principe d'indépendance est beaucoup plus fort que seulement la non-corrélation entre les données (Hyvärinen, Karhunen et Oja, 2001). Les équations suivantes illustrent pourquoi on ne peut pas extraire les ICs seulement grâce à la propriété du blanchiment (Hyvärinen, Karhunen et Oja, 2001, p. 160).

Si l'on prend pour exemple une transformation orthogonale \mathbf{U} de \mathbf{z} :

$$\mathbf{y} = \mathbf{U}\mathbf{z} \quad (2.17)$$

Puisque \mathbf{U} est orthogonale :

$$\mathbf{E}\{\mathbf{y}\mathbf{y}^T\} = \mathbf{E}\{\mathbf{U}\mathbf{y}\mathbf{y}^T\mathbf{U}^T\} = \mathbf{U}\mathbf{I}\mathbf{U}^T = \mathbf{I} \quad (2.18)$$

Ceci démontre que le vecteur \mathbf{y} , qui est en fait une transformation orthogonale du vecteur \mathbf{z} , est aussi considéré comme blanchi. En se basant seulement sur la propriété du blanchiment, on ne peut pas dire si les ICs sont données grâce à \mathbf{z} ou \mathbf{y} . Le blanchiment permet donc seulement de retrouver les ICs à une transformation orthogonale près. Par contre, il ne faut pas négliger l'utilité du blanchiment, car celui-ci rend la nouvelle matrice de mélange $\tilde{\mathbf{A}}$ orthogonale, ce qui simplifie le problème d'ICA.

L'orthogonalité de la nouvelle matrice $\tilde{\mathbf{A}}$ est démontrée selon :

$$E\{\mathbf{z}\mathbf{z}^T\} = \tilde{\mathbf{A}}E\{\mathbf{s}\mathbf{s}^T\}\tilde{\mathbf{A}}^T = \tilde{\mathbf{A}}\tilde{\mathbf{A}}^T = \mathbf{I} \quad (2.19)$$

Au lieu de devoir trouver la matrice de mélange original \mathbf{A} qui contient n^2 paramètres, la recherche est réduite à une matrice orthogonale $\tilde{\mathbf{A}}$ contenant seulement $n(n-1)/2$ paramètres. Puisque le blanchiment est une étape facile à effectuer sur les données et réduit la complexité du problème d'ICA de moitié, il est important d'appliquer ce prétraitement aux données avant l'utilisation de FastICA.

2.4.4 Restrictions et ambiguïtés d'ICA

Afin que le modèle ICA puisse être estimé, deux hypothèses ainsi qu'une restriction doivent être posées. La deuxième hypothèse n'est pas obligatoire, mais celle-ci est posée afin de simplifier le problème.

La première hypothèse est la base d'ICA et repose sur le fait que les ICs sont présumées « statistiquement indépendantes ». Si l'on considère des variables aléatoires telles que y_1, y_2, \dots, y_n pour tout $i \neq j$, celles-ci sont considérées comme indépendante si aucune information n'est donnée à y_j à partir de la valeur de y_i .

Pour ce qui est de la deuxième hypothèse, la matrice de mélange est présumée « carrée », ce qui signifie que le nombre d'ICs est égal aux nombres d'observations. Il est aussi présumé que la matrice de mélange \mathbf{A} est inversible afin de pouvoir facilement retrouver les ICs à partir de l'inverse de cette matrice, soit :

$$\mathbf{s} = \mathbf{A}^{-1}\mathbf{x} \quad (2.20)$$

où \mathbf{A}^{-1} est la matrice inverse de \mathbf{A} .

La restriction finale nécessaire à l'application d'ICA requiert que les ICs aient une « distribution non gaussienne ». Les équations suivantes illustrent pourquoi cette restriction est nécessaire (Hyvärinen, Karhunen et Oja, 2001, p. 161).

En supposant que la distribution conjointe de deux ICs soit gaussienne, leur fonction de densité de probabilité (PDF, *probability density function*) conjointe s'écrit comme suit :

$$p(s_1, s_2) = \frac{1}{2\pi} \exp\left(-\frac{s_1^2 + s_2^2}{2}\right) = \frac{1}{2\pi} \exp\left(-\frac{\|\mathbf{s}\|^2}{2}\right) \quad (2.21)$$

où exp est la fonction exponentielle.

Si l'on présume que les données ont été blanchies, la matrice de mélange \mathbf{A} de base deviendra orthogonale. En utilisant l'équation de transformation des PDFs (voir l'équation (2.82) dans Hyvärinen, Karhunen et Oja (2001, p. 35)) et en sachant que $\mathbf{A}^{-1} = \mathbf{A}^T$ pour une matrice orthogonale, on peut obtenir la densité conjointe des mélanges x_1 et x_2 selon :

$$p(x_1, x_2) = \frac{1}{2\pi} \exp\left(-\frac{\|\mathbf{A}^T \mathbf{x}\|^2}{2}\right) |\det \mathbf{A}^T| \quad (2.22)$$

où det est le déterminant.

Puisque \mathbf{A} est orthogonale, \mathbf{A}^T est aussi orthogonale alors $\|\mathbf{A}^T \mathbf{x}\|^2 = \|\mathbf{x}\|^2$ et $|\det \mathbf{A}^T| = 1$, ce qui donne :

$$p(x_1, x_2) = \frac{1}{2\pi} \exp\left(-\frac{\|\mathbf{x}\|^2}{2}\right) \quad (2.23)$$

On peut donc constater que la matrice de mélange orthogonale ne change pas la PDF, car elle n'est pas présente dans l'équation (2.23). Il est donc impossible de retrouver la matrice de

mélange initiale à partir des signaux mélangés puisque la distribution originale est identique à la distribution mélangée.

Par contre, il est quand même possible de retrouver les ICs si seulement une composante a une distribution gaussienne, car elle ne sera pas mélangée avec une autre composante gaussienne, donc elle pourra être retrouvée.

Il y a deux ambiguïtés du modèle ICA basées sur l'équation (2.9).

La première ambiguïté est basée sur le fait qu'on ne peut pas déterminer les variances des ICs. Puisque \mathbf{s} et \mathbf{A} sont tous deux inconnus dans l'équation (2.9), n'importe quel scalaire multiplié à une des sources s_i pourrait toujours être annulé par l'inverse de ce scalaire multiplié à la colonne \mathbf{a}_i correspondante de la matrice de mélange \mathbf{A} :

$$\mathbf{x} = \sum_i \left(\frac{1}{\alpha_i} \mathbf{a}_i \right) (s_i \alpha_i) \quad (2.24)$$

où α_i est un scalaire.

Cela signifie que l'amplitude des composantes ne peut être estimée (Hyvärinen, Karhunen et Oja, 2001). L'amplitude de chaque composante est donc fixée en présumant une variance unitaire pour toutes les ICs, ce qui se traduit selon :

$$E\{s_i^2\} = 1 \quad (2.25)$$

La solution à ce problème est présentée au chapitre 4.

La deuxième ambiguïté est liée au fait qu'on ne peut pas déterminer l'ordre initial des ICs (Hyvärinen, Karhunen et Oja, 2001). Il est important de retrouver l'ordre initial des ICs afin de savoir quel signal source est associé à quel signal observé. Puisque \mathbf{s} et \mathbf{A} sont inconnus,

l'ordre des composantes peut être changé et n'importe laquelle des ICs peut donc être vue comme la première. Si l'on considère une matrice de permutation \mathbf{P} et son inverse \mathbf{P}^{-1} , comme dans le cas du scalaire \mathbf{a}_i pour la première indétermination, ils peuvent être introduits dans le modèle et s'annuler :

$$\mathbf{x} = \mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{s} \quad (2.26)$$

où \mathbf{P} est une matrice de permutation quelconque.

Ce qui signifie alors que $\mathbf{P}\mathbf{s}$ sont les ICs de base et $\mathbf{A}\mathbf{P}^{-1}$ la matrice de mélange de base, mais tous deux dans des ordres différents. La solution à ce deuxième problème est aussi présentée au chapitre 4.

2.4.5 Indépendance, non-gaussianité et l'entropie négative

Comme mentionnée à la section précédente, l'hypothèse de non-gaussianité des ICs est très importante dans ICA, car sans celle-ci on ne peut utiliser ICA pour extraire les ICs des signaux mélangés. La maximisation de la non-gaussianité selon le théorème limite central (CLT, *central limit theorem*) est utilisée par l'algorithme FastICA afin de retrouver les ICs. Cette section explique pourquoi la non-gaussianité est synonyme d'indépendance selon le CLT et pourquoi il est préférable d'utiliser l'entropie négative plutôt que le kurtosis afin de mesurer la non-gaussianité.

Selon le CLT (voir la section 2.5.2 dans Hyvärinen, Karhunen et Oja (2001, p. 34) pour plus de détails), la distribution associée à la somme de plusieurs variables indépendantes aléatoires sera gaussienne pour un grand nombre de variables. Si, par exemple, deux variables indépendantes aléatoires ont chacune une distribution uniforme, suite à un mélange linéaire de ces deux variables et à un blanchiment de ces mélanges, les mélanges résultants auront une distribution beaucoup plus proche d'une distribution gaussienne.

Sachant qu'on peut retrouver les ICs initiales s_i en inversant la matrice de mélange A et en la multipliant au vecteur d'observation \mathbf{x} selon :

$$\mathbf{s} = \mathbf{A}^{-1}\mathbf{x} \quad (2.27)$$

Alors pour retrouver seulement une composante on obtient :

$$y = \mathbf{b}^T \mathbf{x} = \mathbf{q}^T \mathbf{s} = \sum_i q_i s_i \quad (2.28)$$

où y est une combinaison linéaire du vecteur source \mathbf{s} , \mathbf{b} est un vecteur que l'on doit trouver et \mathbf{q} est le vecteur des coefficients à trouver puisque $y = \mathbf{b}^T \mathbf{A} \mathbf{s} = \mathbf{q}^T \mathbf{s}$.

Alors selon le CLT, on doit trouver le vecteur \mathbf{b} qui maximisera la non-gaussianité de $\mathbf{b}^T \mathbf{x}$ qui est en fait y , une des ICs à trouver.

Deux méthodes grandement utilisées afin de mesurer la non-gaussianité sont le kurtosis et l'entropie négative. Le kurtosis mesure l'aplatissement de la distribution d'une variable aléatoire tandis que l'entropie mesure à quel point une variable est aléatoire. Une version de FastICA utilisant le kurtosis existe, mais celle-ci est moins robuste que celle utilisant l'entropie négative (voir la section 8.2.3 dans Hyvärinen, Karhunen et Oja (2001, p. 178) pour cette version de l'algorithme). Le problème de la mesure de la non-gaussianité avec le kurtosis est sa très grande sensibilité aux mesures aberrantes. Ceci signifie que seulement quelques échantillons erronés peuvent grandement affecter la valeur du kurtosis. L'entropie négative est pour sa part une mesure robuste de la non-gaussianité, mais demande des calculs plus complexes que le kurtosis. Une approximation de l'entropie négative est effectuée afin que les calculs soient simplifiés.

L'entropie est liée à l'information donnée par l'observation d'une variable aléatoire. Plus une variable est imprévisible et non-structurée (donc aléatoire) plus son entropie sera élevée. L'entropie différentielle H (Hyvärinen, Karhunen et Oja, 2001, p. 182) est donné selon :

$$H(\mathbf{y}) = - \int p_{\mathbf{y}}(\boldsymbol{\eta}) \log(p_{\mathbf{y}}(\boldsymbol{\eta})) d\boldsymbol{\eta} \quad (2.29)$$

où \mathbf{y} est un vecteur aléatoire quelconque et $p_{\mathbf{y}}(\boldsymbol{\eta})$ est la densité de probabilité du vecteur \mathbf{y} .

Un principe important en lien avec la théorie de l'information est que pour des variables aléatoires ayant la même variance, la variable gaussienne aura l'entropie la plus élevée. Cette information permet donc d'utiliser l'entropie afin de pouvoir mesurer la non-gaussianité d'une variable. L'entropie négative est une version normalisée de l'entropie différentielle puisqu'elle donne zéro pour une variable gaussienne et est toujours non-négative. L'entropie négative J s'écrit comme suit :

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \quad (2.30)$$

où \mathbf{y}_{gauss} est une variable aléatoire gaussienne ayant la même matrice de corrélation et de covariance que la variable \mathbf{y} .

Le calcul standard de l'entropie négative est complexe puisqu'il requiert un estimé de la PDF, c'est pourquoi l'utilisation de l'approximation de l'entropie négative afin de mesurer la non-gaussianité est effectuée (voir la section 8.3.3 dans Hyvärinen, Karhunen et Oja (2001, p. 183)). L'approximation s'écrit sous la forme :

$$J(\mathbf{y}) \propto [E\{G(\mathbf{y})\} - E\{G(\mathbf{v})\}]^2 \quad (2.31)$$

où y est une variable aléatoire ayant une moyenne nulle et une variance unitaire, v est une variable gaussienne aléatoire ayant aussi une moyenne nulle et une variance unitaire et G est une fonction non quadratique donnant une bonne approximation de l'entropie négative.

Selon Hyvärinen, Karhunen et Oja (2001, p. 184), les fonctions G suivantes donnent de bonnes approximations de l'entropie négative :

$$G_1(y) = \frac{1}{a_1} \log(\cosh(a_1 y)) \quad (2.32)$$

$$G_2(y) = -\exp(-y^2/2) \quad (2.33)$$

où $1 \leq a_1 \leq 2$ est une constante; en général $a_1 = 1$.

La non-gaussianité de $\mathbf{b}^T \mathbf{x}$ présentée à l'équation (2.28) peut être estimée en utilisant l'équation (2.31). Les solutions sont finalement obtenues en maximisant le vecteur \mathbf{b} grâce à l'algorithme FastICA afin que la non-gaussianité de $\mathbf{b}^T \mathbf{x}$ soit maximale.

2.4.6 Avantages de l'algorithme FastICA

FastICA est un algorithme qui a été inventé par Aapo Hyvärinen à l'université de technologie Helsinki en Finlande. Cet algorithme est basé sur une méthode itérative utilisant le point fixe et cherche à maximiser la non-gaussianité. FastICA sert à extraire les ICs des signaux mélangés (Hyvärinen, 1999; Hyvärinen et Oja, 2000; Hyvärinen, Karhunen et Oja, 2001).

Plusieurs propriétés intéressantes sont associées à FastICA (Hyvärinen et Oja, 2000, p. 16). Les propriétés principales de l'algorithme sont les suivantes :

- la convergence de l'algorithme est cubique, donc très rapide comparé aux algorithmes ICA standard qui sont basés sur la descente du gradient où la convergence est seulement linéaire;

- aucune grandeur de pas d'adaptation à choisir, donc facile d'utilisation contrairement aux algorithmes basés sur la descente du gradient où ce paramètre doit toujours être ajusté;
- n'exige aucune estimation de la fonction de distribution de probabilité et les ICs peuvent être trouvées avec n'importe quelles non-linéarités g (les non-linéarités g étant la dérivée des fonctions G estimant l'entropie négative) pourvu que ceux-ci n'aient pas une distribution gaussienne. La section 2.4.7 explique la signification de la non-linéarité g et les fonctions non quadratiques g recommandées;
- grâce au choix de la non-linéarité g , l'algorithme FastICA peut être optimisé aux besoins de l'utilisateur (ex. : algorithme robuste, variance minimale, etc.);
- FastICA possède plusieurs avantages des algorithmes neuronaux : parallèle, distribué, calculs simples et requiert peu d'espace mémoire.

2.4.7 L'algorithme FastICA pour une composante

Cette section présente le cœur de l'algorithme FastICA pour extraire une IC tandis que la section suivante montre comment utiliser cet algorithme de manière itérative afin d'en extraire plusieurs.

L'algorithme FastICA utilise la méthode du point fixe et cherche à trouver la direction du vecteur unitaire \mathbf{w} qui maximisera la non-gaussianité de la projection $\mathbf{w}^T \mathbf{z}$. Une fois le blanchiment effectué, $\mathbf{b}^T \mathbf{x}$ présenté à l'équation (2.28) devient $\mathbf{w}^T \mathbf{z}$. L'approximation de l'entropie négative est utilisée afin de mesurer la non-gaussianité grâce à l'équation (2.31). Comme mentionné à la section 2.4.3, le blanchiment des données est utilisé comme prétraitement avant l'application de l'algorithme FastICA. L'équation principale de FastICA est la suivante (plus de détails sont disponibles à la section 8.3.5 de Hyvärinen, Karhunen et Oja (2001, p. 188)) :

$$\mathbf{w} \leftarrow E\{\mathbf{z}g(\mathbf{w}^T \mathbf{z}) - E\{g'(\mathbf{w}^T \mathbf{z})\}\mathbf{w}\} \quad (2.34)$$

où \mathbf{w} est un des vecteurs de la matrice de séparation \mathbf{W} et g est une fonction non quadratique donnant une bonne approximation de l'entropie négative.

Les fonctions g_1 et g_2 sont les dérivées des fonctions G_1 et G_2 . Les fonctions principales g à utiliser pour faire l'approximation de l'entropie négative selon Hyvärinen, Karhunen et Oja (2001, p. 185) sont les suivantes :

$$g_1(y) = \tanh(a_1 y) \quad (2.35)$$

$$g_2(y) = y \exp(-y^2/2) \quad (2.36)$$

$$g_3(y) = y^3 \quad (2.37)$$

où $1 \leq a_1 \leq 2$ est une constante; généralement $a_1 = 1$.

Les dérivées g' de ces fonctions (Hyvärinen, Karhunen et Oja, 2001, p. 190) sont :

$$g'_1(y) = a_1(1 - \tanh^2(a_1 y)) \quad (2.38)$$

$$g'_2(y) = (1 - y^2) \exp(-y^2/2) \quad (2.39)$$

$$g'_3(y) = 3y^2 \quad (2.40)$$

Les étapes à suivre afin d'estimer une des ICs sont les suivantes :

1. Centrer les données afin d'avoir une moyenne nulle;
2. Blanchir les données afin d'obtenir le vecteur \mathbf{z} ;
3. Choisir un vecteur initial \mathbf{w} (ex. au hasard) ayant une norme unitaire;
4. Effectuer $\mathbf{w} \leftarrow E\{\mathbf{z}g(\mathbf{w}^T \mathbf{z})\} - E\{g'(\mathbf{w}^T \mathbf{z})\}\mathbf{w}$;
5. Effectuer $\mathbf{w} \leftarrow \mathbf{w}/\|\mathbf{w}\|$;
6. Retourner à l'étape 4 si \mathbf{w} n'a pas convergé.

La convergence de \mathbf{w} signifie que le nouveau vecteur \mathbf{w} pointe dans la même direction que l'ancien ou plus techniquement que la valeur absolue du produit scalaire de ces deux vecteurs est près de 1.

2.4.8 Estimation de plusieurs composantes indépendantes avec FastICA

En utilisant l'équation (2.34) de manière itérative, l'estimation de plusieurs ICs est possible.

Par contre, avant de pouvoir utiliser l'algorithme FastICA de manière itérative, l'orthogonalité des vecteurs w_i doit être respectée. L'orthogonalité de la nouvelle matrice de mélange $\tilde{\mathbf{A}}$ est démontrée grâce à l'équation (2.19). Puisque la matrice de mélange est présumée inversible, les ICs sont obtenues selon :

$$\mathbf{s} = \tilde{\mathbf{A}}^{-1} \mathbf{z} = \mathbf{W} \mathbf{z} \quad (2.41)$$

où \mathbf{W} est la matrice de séparation.

À partir de l'équation (2.16), pour seulement une composante :

$$s_i = \mathbf{w}_i^T \mathbf{z} \quad (2.42)$$

où s_i est un des ICs.

Afin que les composantes soient toujours considérées comme indépendantes, elles doivent rester décorrélées, car selon le blanchiment on a $E\{(\mathbf{w}_i^T \mathbf{z})(\mathbf{w}_j^T \mathbf{z})\} = \mathbf{w}_i^T \mathbf{w}_j$, ce qui signifie que la décorrélation est donc équivalente à l'orthogonalité.

En résumé, pour estimer plusieurs composantes, l'équation (2.34) doit être utilisée itérativement pour les vecteurs $\mathbf{w}_1, \dots, \mathbf{w}_n$ et afin d'empêcher ceux-ci de converger à des maximums identiques, l'orthogonalisation des vecteurs $\mathbf{w}_1, \dots, \mathbf{w}_n$ doit être effectuée après

chaque itération. L'orthogonalisation par déflation et l'orthogonalisation symétrique sont deux méthodes de décorrélation possibles.

La première méthode d'orthogonalisation est « l'orthogonalisation par déflation » en utilisant la méthode de Gram-Schmidt (Hyvärinen, Karhunen et Oja, 2001). Cette méthode estime les ICs une à la suite de l'autre. Les étapes suivantes montrent le fonctionnement de cette méthode :

1. Centrer les données afin d'avoir une moyenne nulle;
2. Blanchir les données afin d'obtenir le vecteur \mathbf{z} ;
3. Choisir le nombre d'ICs m à trouver et fixer le compteur $p \leftarrow 1$;
4. Choisir une valeur initiale pour \mathbf{w}_p ayant une norme unitaire (ex. au hasard);
5. Effectuer $\mathbf{w}_p \leftarrow \mathbf{E}\{\mathbf{z}g(\mathbf{w}_p^T\mathbf{z})\} - \mathbf{E}\{g'(\mathbf{w}_p^T\mathbf{z})\}\mathbf{w}$, où g est défini à la section 2.4.7;
6. Procéder à l'orthogonalisation suivante :

$$\mathbf{w}_p \leftarrow \mathbf{w}_p - \sum_{j=1}^{p-1} (\mathbf{w}_p^T \mathbf{w}_j) \mathbf{w}_j \quad (2.43)$$

7. Effectuer $\mathbf{w}_p \leftarrow \mathbf{w}_p / \|\mathbf{w}_p\|$;
8. Si \mathbf{w}_p n'a pas convergé, retourner à l'étape 5;
9. Fixer $p \leftarrow p + 1$. Si $p \leq m$, retourner à l'étape 4.

La deuxième méthode d'orthogonalisation est « l'orthogonalisation symétrique » (Hyvärinen, Karhunen et Oja, 2001) qui ne privilégie aucun vecteur par rapport à un autre. Les vecteurs \mathbf{w}_i sont donc estimés parallèlement plutôt qu'individuellement. L'orthogonalisation symétrique possède deux avantages principaux par rapport à l'orthogonalisation par déflation. Le premier avantage est qu'il n'y a pas d'erreurs d'estimation dans les premiers vecteurs qui sont cumulés dans les vecteurs suivants lors de l'orthogonalisation puisque tous les vecteurs sont estimés en même temps. Le deuxième avantage est que les ICs peuvent être calculées par des processeurs parallèles. Les étapes suivantes montrent le fonctionnement de cette méthode :

1. Centrer les données afin d'avoir une moyenne nulle;
2. Blanchir les données afin d'obtenir le vecteur \mathbf{z} ;
3. Choisir le nombre d'ICs m à trouver;
4. Choisir les valeurs initiales (ex. au hasard) pour les \mathbf{w}_i , où $i = 1, \dots, m$, chacune ayant une norme unitaire. Faire l'orthogonalisation de la matrice de séparation \mathbf{W} selon l'étape 6 ci-dessous;
5. Pour tous $i = 1, \dots, m$, effectuer $\mathbf{w}_i \leftarrow \mathbf{E}\{\mathbf{z}g(\mathbf{w}_i^T \mathbf{z})\} - \mathbf{E}\{g'(\mathbf{w}_i^T \mathbf{z})\}\mathbf{w}_i$, où g est défini à la section 2.4.7;
6. Procéder à l'orthogonalisation symétrique de la matrice $\mathbf{W} = (\mathbf{w}_1, \dots, \mathbf{w}_m)^T$ selon :

$$\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^T)^{-1/2}\mathbf{W} \quad (2.44)$$
7. S'il n'y a pas convergence, retourner à l'étape 5.

La moyenne est rajoutée aux données une fois la convergence atteinte et ce pour les deux méthodes d'orthogonalisation, car les données ont été centrées avant le blanchiment afin d'avoir une moyenne nulle.

2.5 Conclusion

Comme expliquée dans la première section sur la revue de la littérature dans le domaine de la calibration des WSNs, aucune des solutions mentionnées ne s'applique à notre cas. Les différents capteurs sont fixes et mesurent des signaux non corrélés. Toutefois, un capteur précis mesure la somme des courants. Ceci nous permet de proposer de nouvelles solutions, telles que l'algorithme LMS et FastICA.

La théorie de l'algorithme LMS est introduite dans la deuxième partie de ce chapitre. LMS est un algorithme simple, robuste et adaptatif basé sur une approximation instantanée du gradient. Bien que la vitesse de convergence de cet algorithme soit plus lente comparée à la vitesse d'algorithmes plus récents, la vitesse de convergence n'est pas un paramètre critique dans cette application. L'utilisation adaptée de cet algorithme à notre système est effectuée au début du chapitre 3.

Ensuite, la théorie de l'ICA basée sur le principe d'indépendance et les statistiques est introduite. Afin de pouvoir appliquer ICA pour extraire les ICs des signaux mélangés, il est essentiel que celles-ci soient indépendantes et que leur distribution ne soit pas gaussienne. Avant d'appliquer l'algorithme FastICA sur les signaux mélangés, le prétraitement de ces données grâce au blanchiment est une étape simple à effectuer et réduit de moitié le problème d'ICA. Avant l'introduction de l'algorithme FastICA maximisant la non-gaussianité, il est expliqué que l'entropie négative est une méthode plus robuste que le kurtosis afin de mesurer la non-gaussianité.

Finalement, l'algorithme FastICA est présenté pour l'estimation d'une composante et ensuite pour l'estimation de plusieurs composantes. Deux méthodes d'orthogonalisation sont présentées pour l'estimation de plusieurs ICs. L'orthogonalisation par déflation estime les ICs une à la suite de l'autre tandis que l'orthogonalisation symétrique les estime parallèlement. L'utilisation de l'algorithme FastICA adaptée à notre système est présentée au début du chapitre 4.

CHAPITRE 3

SOLUTION UTILISANT L'ALGORITHME LMS POUR LA CALIBRATION DES CAPTEURS

3.1 Introduction

La théorie du LMS ayant été présentée au chapitre précédent, les résultats de calibration obtenus en simulation avec cet algorithme peuvent maintenant être présentés. Un court retour sur la théorie est fait au début de ce chapitre en l'adaptant à notre système. Ensuite, les résultats de simulations obtenus pour 5 et 30 capteurs sont présentés. La dernière section montre l'inefficacité de l'algorithme LMS à faire la calibration des capteurs lorsque de la diaphonie est ajoutée aux mesures.

3.2 Retour sur la théorie de l'algorithme LMS

Les deux équations principales à la base de LMS sont les équations (2.1) et (2.6). Dans le cas du système de capteurs à l'étude, au lieu d'estimer les coefficients d'un filtre sur les données entrantes en observant la sortie du filtre, nous proposons d'estimer les coefficients de gain des capteurs en observant les mesures des capteurs et la somme précise des mesures, comme le montre la figure 3.1. L'équation d'adaptation des coefficients et du signal d'erreur devient donc :

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu e(n) \mathbf{I}_m(n) \quad (3.1)$$

$$e(n) = I_T(n) - \hat{\mathbf{w}}^T(n) \mathbf{I}_m(n) \quad (3.2)$$

où $\hat{\mathbf{w}}(n)$ est le vecteur des coefficients de gain des capteurs, $e(n)$ est l'erreur du courant total, μ est le pas d'adaptation, $\mathbf{I}_m(n)$ est le vecteur des courants mesurés et $I_T(n)$ est le courant total précis.

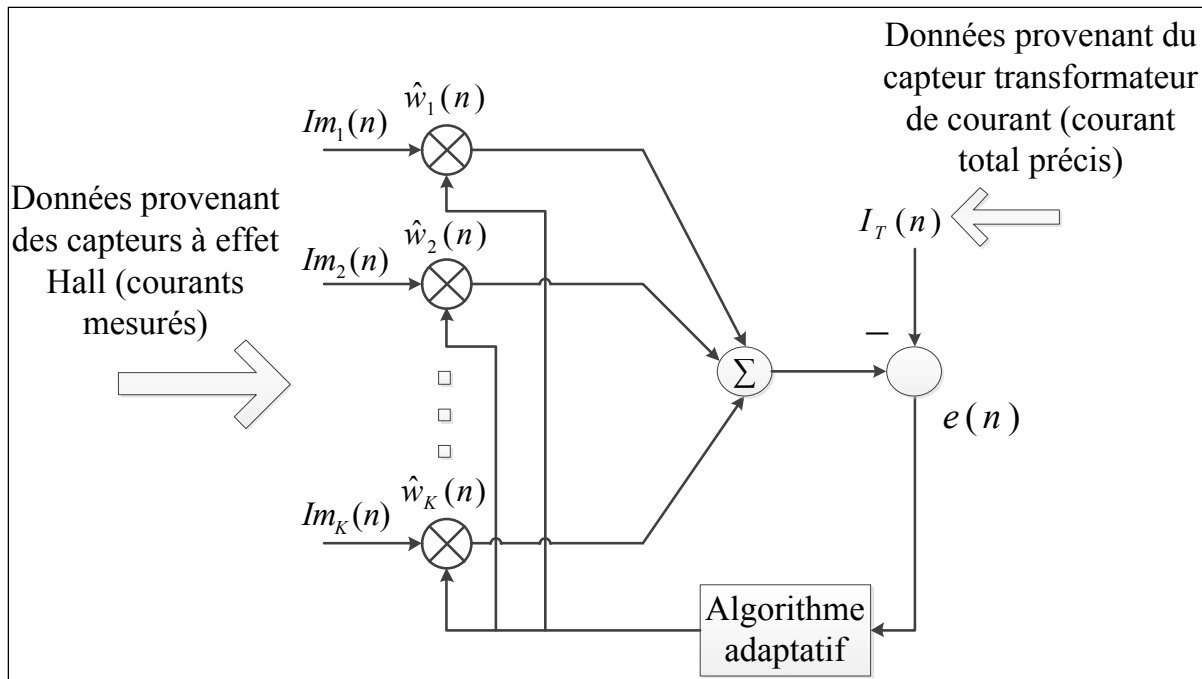


Figure 3.1 Schéma de calibration avec l'algorithme LMS

Selon le schéma de la figure 3.1, les coefficients $\hat{\mathbf{w}}(n)$ sont trouvés grâce au courant total précis $I_T(n)$ utilisé comme signal de référence afin de réduire l'erreur entre les signaux mesurés $\mathbf{I}_m(n)$ et les signaux réels $\mathbf{I}(n)$. Le courant estimé qui est corrigé grâce aux coefficients trouvés se traduit donc selon :

$$\hat{\mathbf{I}}(n) = \mathbf{I}_m(n) .* \hat{\mathbf{w}}(n) \quad (3.3)$$

où $\hat{\mathbf{I}}(n)$ est le vecteur des courants estimés.

En ajustant le pas d'adaptation μ , un bon ratio entre une erreur résiduelle faible ($< 1 A_{RMS}$) et un temps de convergence raisonnable (< 5 minutes) peut être trouvé. Dans notre cas, puisque le système n'a pas besoin de converger très rapidement, la réduction de l'erreur résiduelle est priorisée par rapport au temps de convergence.

3.3 Résultats de simulation avec l'algorithme LMS

Plusieurs paramètres sont utilisés en simulation afin de pouvoir bien contrôler et modéliser les signaux à l'entrée afin qu'ils soient représentatifs des signaux réels. Tout le traitement numérique des signaux est effectué à l'aide du logiciel Matlab[®]. Les signaux réels $I(n)$ sont générés aléatoirement en utilisant une distribution uniforme tandis que les gains aléatoires α des capteurs sont générés en utilisant une distribution normale de moyenne unitaire. Le bruit ajouté aux signaux mesurés $\eta(n)$ est représenté comme étant du bruit blanc gaussien. Du bruit blanc gaussien est aussi ajouté au courant total précis $\eta_{CT}(n)$, mais celui-ci est considéré comme étant 10 fois moins élevé que celui ajouté aux signaux mesurés.

Puisque le courant maximal traversant la plupart des disjoncteurs résidentiels nord-américains est de 20A, l'amplitude maximale des signaux est fixée à 20. Toutes les mesures de courants sont en A_{RMS} . La figure 3.2 montre un exemple de 3 signaux réels variant aléatoirement avec une amplitude maximale de 20A suivant une distribution uniforme.

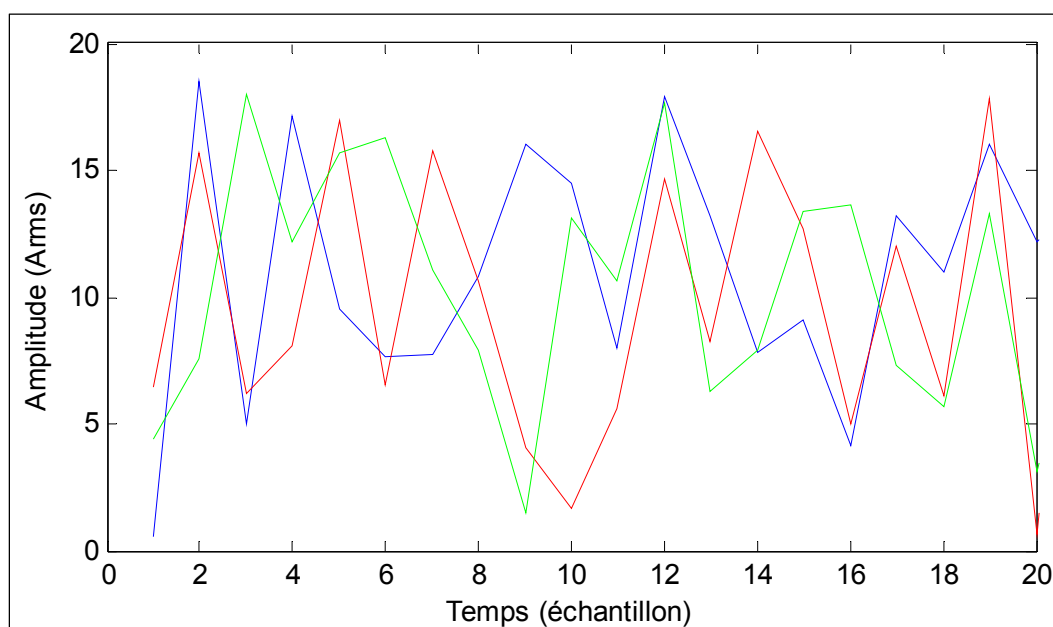


Figure 3.2 Exemple de 3 signaux réels ayant une distribution uniforme

Le nombre de capteurs K , le nombre d'échantillons T , l'écart-type des gains $\sigma_g = 0,16$ et du bruit $\sigma_n = 0,5$ ainsi que le pas d'adaptation μ sont tous des paramètres à fixer avant chaque simulation (Levasseur, 2013). L'écart-type des gains σ_g et du bruit σ_n sont basés sur les valeurs observées en milieu expérimental. Puisque les coefficients $\hat{\mathbf{w}}(n)$ trouvés par LMS changent à chaque échantillon, il faut aussi spécifier combien de coefficients doivent être utilisés vers la fin de la convergence de LMS afin de faire une moyenne des derniers coefficients trouvés pour déterminer les $\hat{\mathbf{w}}$ finaux. Ce paramètre est représenté selon \hat{N} et a seulement une utilité à la section 3.4.2 lorsque l'erreur par capteur est présentée. Le pas d'adaptation μ est trouvé par essais erreur afin que l'erreur résiduelle soit $< 1 A_{\text{RMS}}$ ou bien que le plancher de bruit soit atteint.

Afin que les simulations soient les plus représentatives possible, la méthode Monte-Carlo qui consiste à faire la moyenne des résultats sur plusieurs simulations est utilisée.

3.4 Résultats de simulation avec LMS sans diaphonie

Cette section est divisée en sous-section selon le nombre de capteurs utilisés lors des simulations soit 3 et 30. Chaque section présente l'évolution des coefficients de l'algorithme LMS ainsi que la convergence de l'erreur totale avec et sans ajout de bruit. L'erreur RMS moyenne par capteur est présentée à la dernière section.

3.4.1 Calibration de 3 capteurs avec LMS

Avant de présenter les résultats pour des cas plus complexes, la calibration de seulement 3 capteurs est effectuée comme preuve de concept. Les tests sont réalisés avec et sans l'ajout de bruit aux mesures afin d'évaluer les performances de LMS.

Les paramètres sont fixés comme suit : $K = 3$, $T = 1000$, $\sigma_g = 0,16$, $\sigma_n = 0$ et $\mu = 0,0005$.

La figure 3.3 représente la convergence des coefficients pour une simulation sans ajout de bruit. On peut donc constater que LMS converge à des coefficients stables sans problème. Les lignes droites représentent les coefficients inconnus que l'algorithme LMS tente de trouver. La figure 3.4 montre l'erreur totale du courant pour 1000 simulations Monte-Carlo convergeant à $0,066 A_{RMS}$ après 300 échantillons.

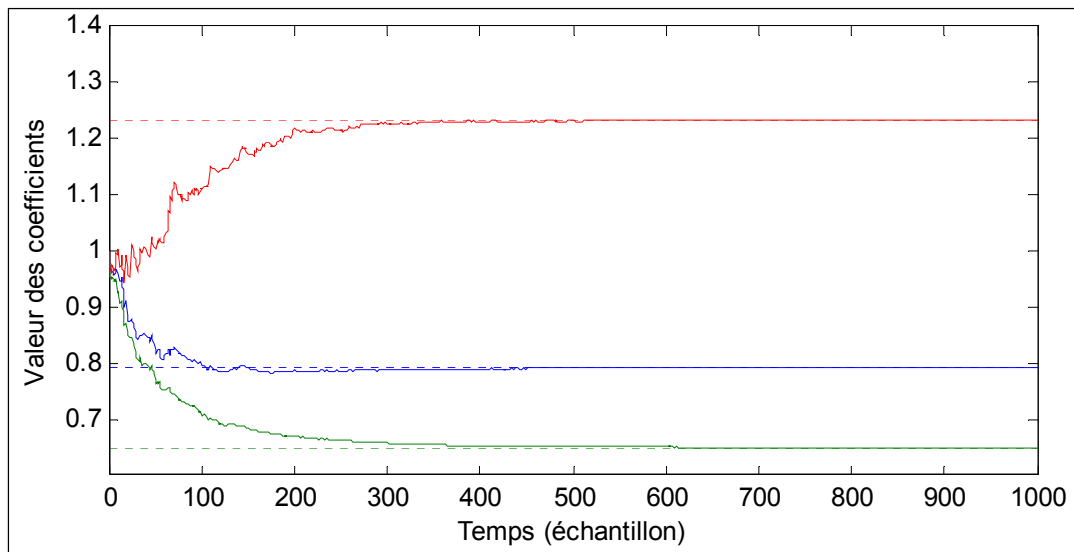


Figure 3.3 Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation

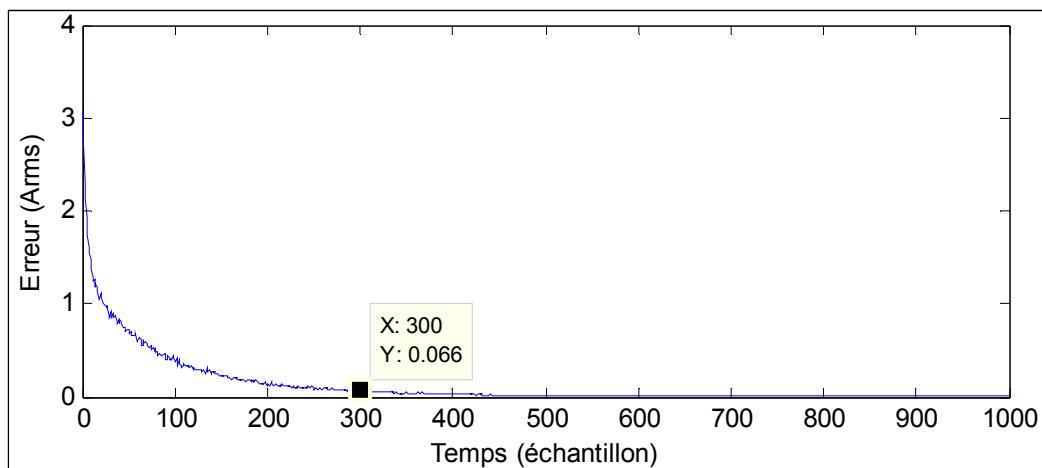


Figure 3.4 Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 1000 simulations

L'ajout de bruit au système est obtenu en posant $\sigma_n = 0,5$. Puisque le niveau de bruit ajouté est élevé, le nombre d'échantillons et le pas d'adaptation sont modifiés aux valeurs suivantes : $T = 5000$ et $\mu = 0,00005$ afin que l'algorithme LMS converge.

La figure 3.5 montre l'évolution des coefficients en présence de bruit pour une simulation et la figure 3.6 l'erreur totale du courant avec Monte-Carlo pour 1000 simulations. On peut observer à la figure 3.5 que l'algorithme LMS réussit à trouver des coefficients à moins de 1% de ceux recherchés malgré la présence de bruit. Tel qu'on peut le voir à la figure 3.6, l'erreur totale converge à $1,04 A_{RMS}$ après 1000 échantillons. L'algorithme LMS ne peut réduire davantage l'erreur puisque le plancher de bruit est atteint.

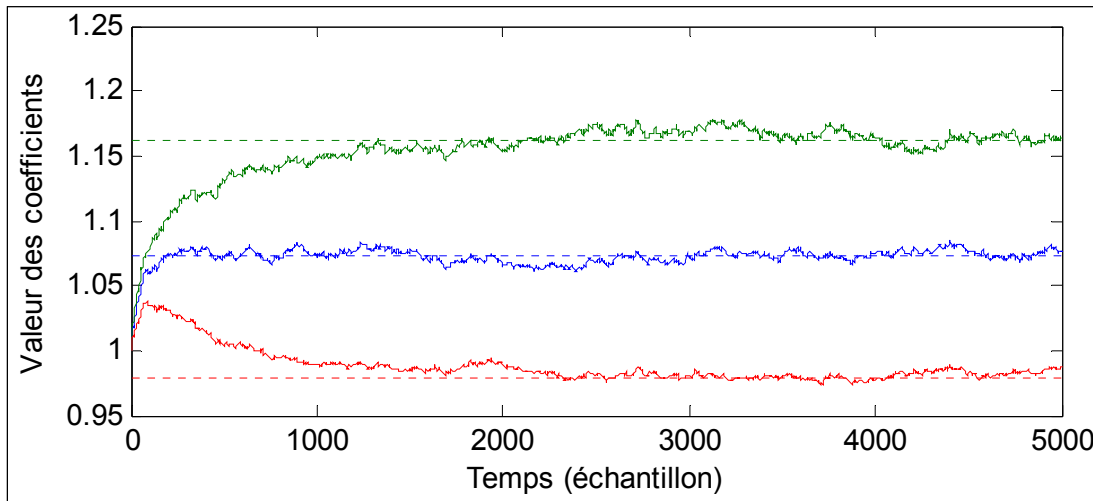


Figure 3.5 Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0,5$ pour 1 simulation

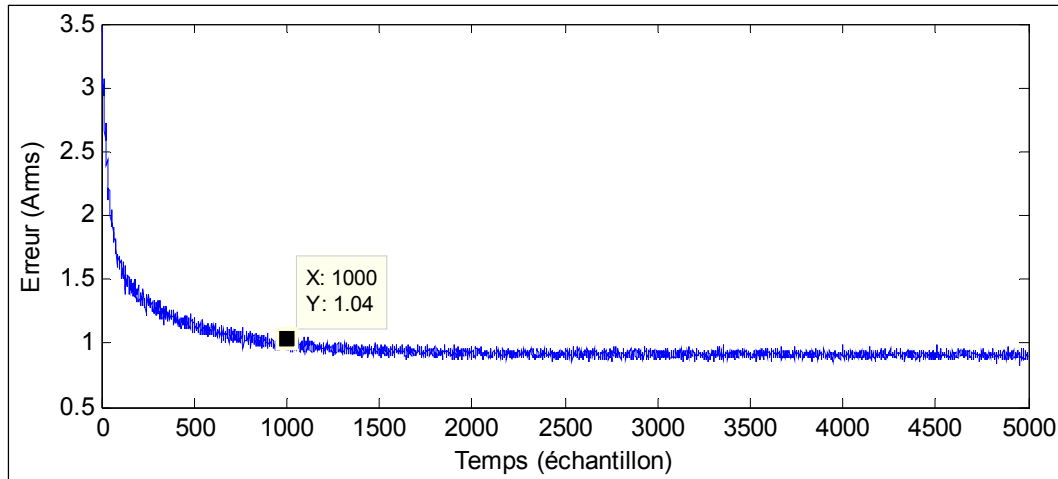


Figure 3.6 Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations

3.4.2 Calibration de 30 capteurs avec LMS

Après avoir présenté les résultats de convergence avec l'algorithme LMS pour 3 capteurs, le nombre de capteurs est augmenté à $K = 30$. Ceci représenterait une maison familiale utilisant le système pour mesurer la consommation de chacun des disjoncteurs du panneau. Les mêmes tests que dans les sections précédentes sont effectués en simulant sans bruit et avec bruit.

Puisque le nombre de capteurs est rendu plus élevé, la valeur du pas d'adaptation doit être réduite à $\mu = 0,00005$ afin que l'algorithme LMS converge et le nombre d'échantillons doit être augmenté à $T = 10000$ afin de mieux voir la convergence de LMS. Les résultats des simulations précédentes sans bruit auraient été identiques avec $\mu = 0,00005$, mais la convergence des coefficients et de l'erreur totale avec LMS auraient été plus lente.

Tout d'abord, les paramètres sont fixés comme suit : $K = 30$, $T = 10000$, $\sigma_g = 0,16$, $\sigma_n = 0$ et $\mu = 0,00005$.

La figure 3.7 montre la convergence des coefficients vers environ 2000 échantillons au lieu de 300 pour 3 capteurs. En observant la figure 3.8, on peut constater que pour 1000 simulations Monte-Carlo l'erreur totale est réduite à $0,13 A_{\text{RMS}}$ après 5000 échantillons et à $0,076 A_{\text{RMS}}$ après 7000.

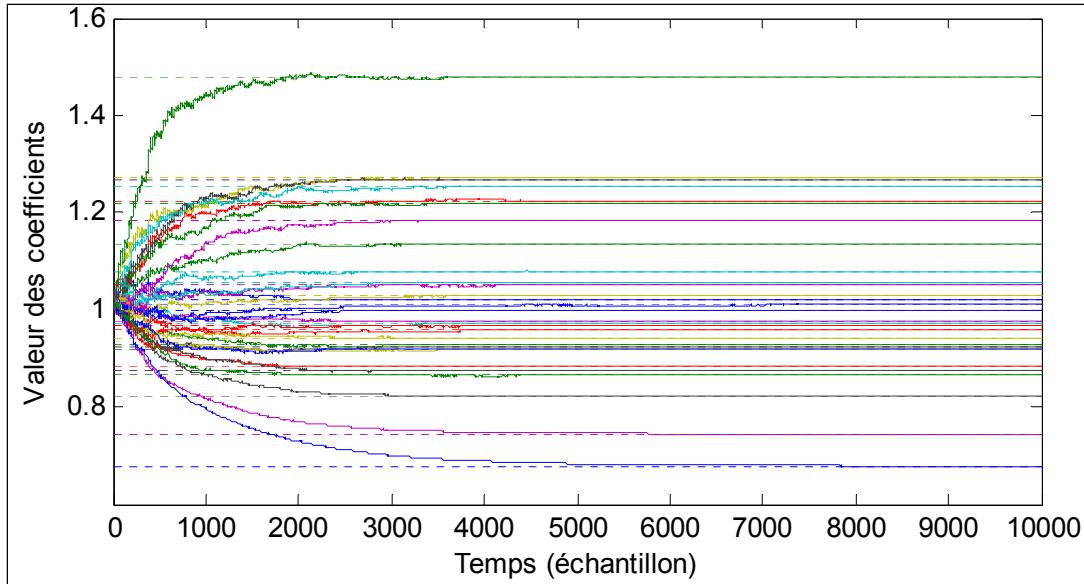


Figure 3.7 Évolution des coefficients du filtre pour 30 capteurs avec $\sigma_n = 0$ pour 1 simulation

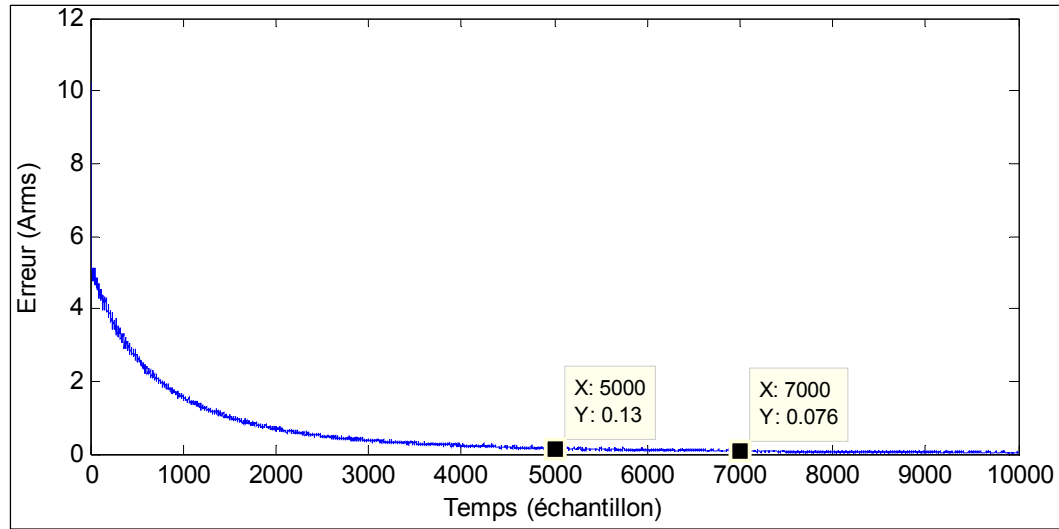


Figure 3.8 Erreur du courant total (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0$ pour 1000 simulations

Du bruit est ajouté aux mesures en fixant $\sigma_n = 0,5$ et en conservant les mêmes paramètres pour le pas d'adaptation et le nombre d'échantillons.

La figure 3.9 montre que les coefficients convergent à moins de 5% des valeurs recherchées. Tel que démontré à la figure 3.10, l'erreur totale converge à 3,19 A_{RMS} après 5000 échantillons et à 3 A_{RMS} après 7000.

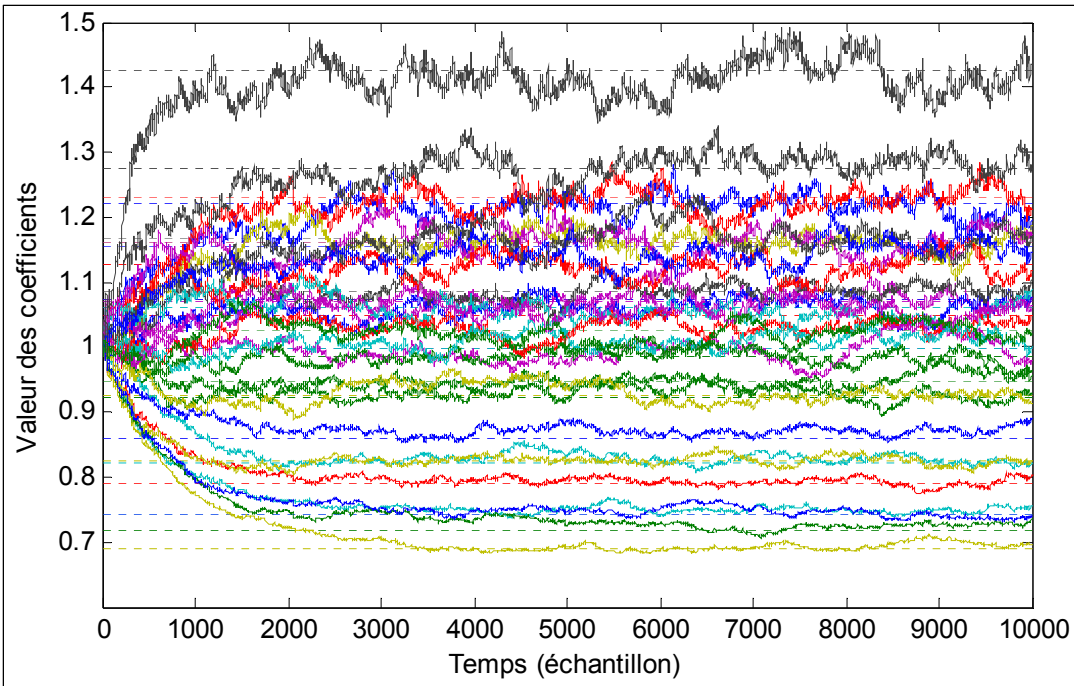


Figure 3.9 Évolution des coefficients du filtre pour 30 capteurs avec $\sigma_n = 0,5$ pour 1 simulation

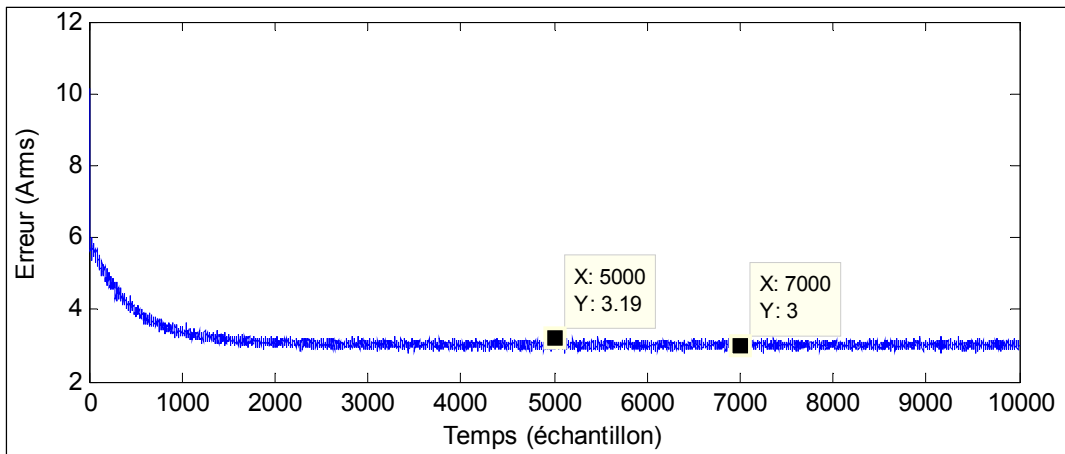


Figure 3.10 Erreur du courant total (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations

La prochaine partie présente l'erreur RMS moyenne par capteur avec et sans calibration pour 3 à 30 capteurs. La variation de l'erreur par capteur pourra donc être observée.

Le premier graphique (figure 3.11) présente l'erreur moyenne par capteur pour un nombre de capteurs variant entre 3 et 30 sans inclure de bruit tandis que le second (figure 3.12) inclut le même niveau de bruit qui a été utilisé précédemment. Les paramètres à fixer pour la figure 3.11 sont : $K = 3$ à 30, $T = 10000$, $\sigma_g = 0,16$, $\sigma_n = 0$, $\mu = 0,00005$. Le nombre d'échantillons utilisés en fin de simulation pour calculer l'erreur résiduelle est $\hat{N} = 500$. Un total de 1000 simulations Monte-Carlo est effectué pour chaque quantité de capteurs afin d'obtenir une valeur précise de l'erreur par capteur.

La figure 3.11 montre clairement que peu importe le nombre de capteurs, l'erreur moyenne par capteur varie très peu. La moyenne de l'erreur moyenne par capteur sans LMS est de 1,48 A_{RMS} et est réduite à 0,0008 A_{RMS} lorsqu'on applique l'algorithme LMS.

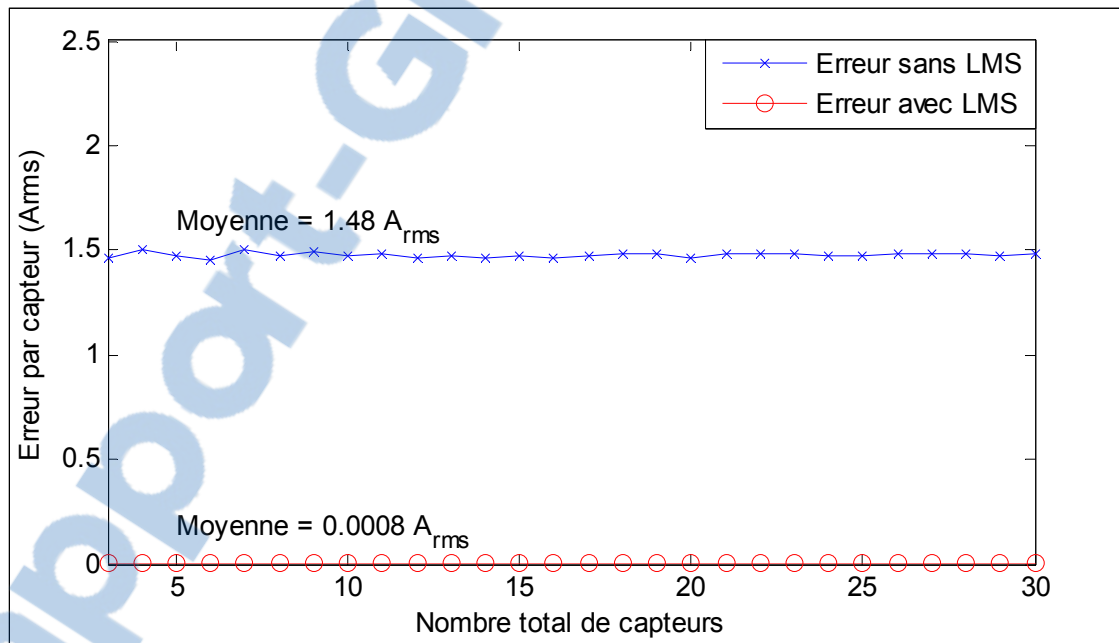


Figure 3.11 Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec $\sigma_n = 0$ pour 1000 simulations pour chaque cas

En fixant $\sigma_n = 0,5$, il est possible de voir si l'erreur par capteur varie lorsque du bruit est ajouté aux mesures. Mis à part l'ajout de bruit, les mêmes paramètres que pour la figure 3.11 sont utilisés.

La figure 3.12 montre encore une fois que la variation de l'erreur moyenne par capteur est faible. Cette fois la moyenne de l'erreur moyenne par capteur sans LMS est de $1,62 A_{RMS}$, ce qui est plus élevé qu'à la figure 3.11 puisque du bruit est ajouté aux mesures. L'erreur par capteur est réduite jusqu'à ce que le plancher de bruit soit atteint, ce qui résulte à une moyenne de l'erreur moyenne par capteur de $0,53 A_{RMS}$.

Ceci vient donc confirmer que l'erreur par capteur est stable pour un nombre de capteurs variant entre 3 et 30 avec ou sans l'ajout de bruit aux mesures et que l'algorithme LMS est apte à minimiser l'erreur jusqu'à ce que le plancher de bruit soit atteint. L'objectif initial d'une erreur $< 1 A_{RMS}$ est donc respecté avec et sans l'ajout de bruit.

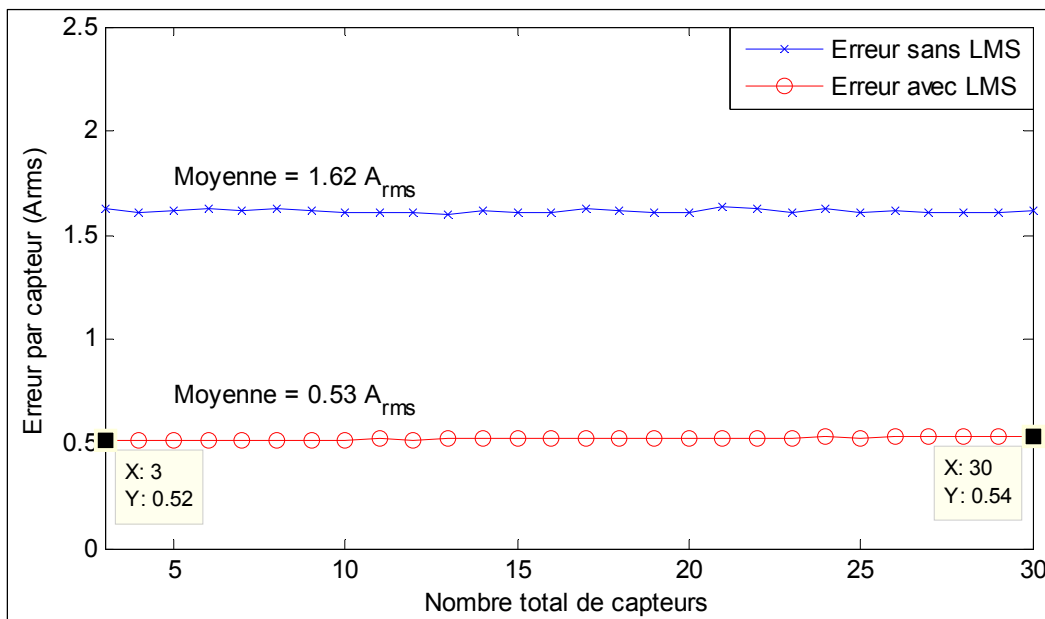


Figure 3.12 Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec $\sigma_n = 0,5$ pour 1000 simulations pour chaque cas

3.5 Résultats de simulation avec LMS et diaphonie

L'erreur par capteur en présence de diaphonie est montrée afin de voir si l'algorithme LMS est en mesure de calibrer les capteurs dans ce cas.

Si l'on considère par exemple 3 capteurs, qu'il n'y a pas de bruit ajouté aux mesures et que de la diaphonie est créée entre chaque capteur, les équations suivantes sont obtenues :

$$I_{m1}(n) = \alpha_{11} I_1(n) + \alpha_{12} I_2(n) + \alpha_{13} I_3(n) \quad (3.4)$$

$$I_{m2}(n) = \alpha_{21} I_1(n) + \alpha_{22} I_2(n) + \alpha_{23} I_3(n) \quad (3.5)$$

$$I_{m3}(n) = \alpha_{31} I_1(n) + \alpha_{32} I_2(n) + \alpha_{33} I_3(n) \quad (3.6)$$

où α_{11} , α_{22} , et α_{33} sont les gains principaux des capteurs tandis que α_{12} , α_{13} , α_{21} , α_{23} , α_{31} et α_{32} sont les gains de diaphonie.

Ce qui signifie qu'au lieu d'avoir n inconnus, où n est le nombre de capteurs total dans le système, on obtient n^2 inconnus. Si l'on représente les équations (3.4), (3.5) et (3.6) sous forme matricielle on obtient :

$$\begin{bmatrix} I_{m1} \\ I_{m2} \\ I_{m3} \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} \quad (3.7)$$

Le modèle général utilisé pour la diaphonie est expliqué plus précisément à la section 4.2 du chapitre 4.

3.5.1 Calibration de 3 capteurs avec LMS et diaphonie

Les résultats de convergence de 3 capteurs pour une simulation sans ajout de bruit sont présentés en utilisant une matrice de coefficients a_{ij} aléatoire.

Les paramètres à fixer sont : $K = 3$, $T = 1000$, $\sigma_g = 0,16$, $\sigma_n = 0$ et $\mu = 0,0005$. Afin de représenter l'erreur RMS moyenne pour chaque capteur, le paramètre \hat{N} est fixé comme suit : $\hat{N} = 500$. Les coefficients aléatoires de diaphonie a_{ij} , où $i \neq j$, suivent une distribution uniforme dont la borne inférieure $a = 0$ et la borne supérieure $b = 0,2$, alors que les

coefficients aléatoires de gains des capteurs a_{ij} , où $i = j$, suivent une distribution normale de moyenne unitaire. Un exemple de matrice de coefficients devient :

$$a = \begin{bmatrix} 1,0234 & 0,0773 & 0,0139 \\ 0,0194 & 0,9163 & 0,0925 \\ 0,1961 & 0,0959 & 0,8345 \end{bmatrix} \quad (3.8)$$

La figure 3.13 montre clairement que les coefficients trouvés par LMS ne sont pas les mêmes que les coefficients principaux inconnus situés sur la diagonale principale de la matrice des coefficients a_{ij} de l'équation (3.8). Par contre, en observant la figure 3.14, l'algorithme LMS semble avoir convergé normalement vers une erreur de 0,036 A_{RMS} après 300 échantillons comme à la figure 3.4 pour les mêmes paramètres sans l'ajout de diaphonie. L'erreur totale est peut-être faible, mais l'erreur RMS moyenne par capteur ne l'est pas. En effet, on trouve une erreur de 2,14 A_{RMS} , 1,21 A_{RMS} et 2,67 A_{RMS} pour les 3 capteurs, soit une erreur moyenne de 2,01 A_{RMS}

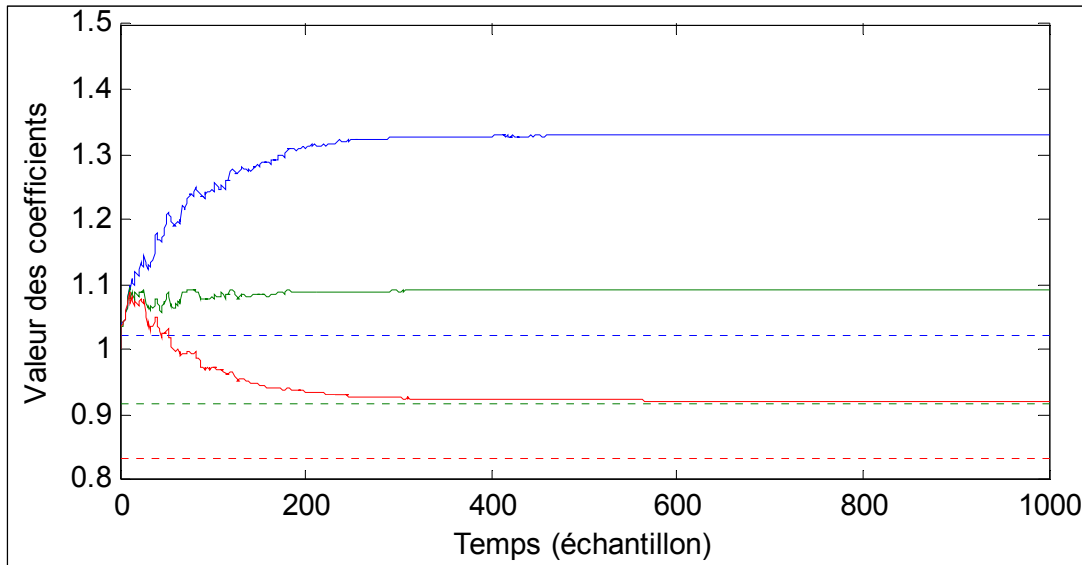


Figure 3.13 Évolution des coefficients du filtre pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation incluant de la diaphonie

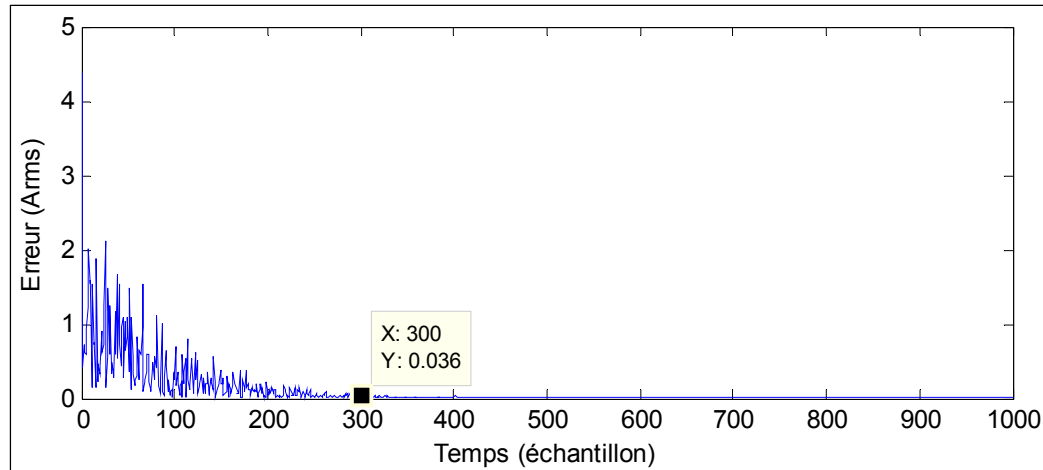


Figure 3.14 Erreur du courant total (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 1 simulation incluant de la diaphonie

L'algorithme LMS n'est donc pas en mesure de corriger le problème de diaphonie puisque seulement 1 coefficient est trouvé pour ajuster les mesures de chaque capteur tandis que selon les équations (3.4), (3.5) et (3.6), 3 coefficients par capteur seraient nécessaires.

3.6 Conclusion

Un retour sur la théorie de l'algorithme LMS est effectué au début du chapitre et un schéma de la calibration des mesures avec LMS est présenté.

Pour les résultats de simulations avec 3 capteurs, l'algorithme LMS a convergé à une erreur de 0,066 A_{RMS} après 300 échantillons. Suite à l'ajout de bruit aux mesures, LMS a convergé à une erreur totale de 1,04 A_{RMS} après 1000 échantillons.

Pour 30 capteurs, l'erreur totale est diminuée à 0,076 A_{RMS} après 7000 échantillons. Suite à l'ajout de bruit aux mesures, l'erreur totale converge à 3 A_{RMS} après 7000 échantillons.

Puisque l'erreur totale augmente selon le nombre de capteurs, l'erreur RMS moyenne par capteur pour 3 à 30 capteurs est présentée par la suite afin de vérifier si celle-ci reste stable. Les résultats sans ajout de bruit montrent que la moyenne de l'erreur moyenne par capteur

pour 3 à 30 capteurs sans LMS est de $1,48 A_{RMS}$ et de $0,0008 A_{RMS}$ avec l'utilisation de LMS. Les résultats montrent aussi que l'erreur par capteur ne varie presque pas, peu importe le nombre de capteurs. Pour ce qui est des résultats avec l'ajout de bruit pour 3 à 30 capteurs, la moyenne de l'erreur moyenne par capteur est de $1,62 A_{RMS}$ sans LMS et $0,53 A_{RMS}$ avec LMS. Encore une fois, l'erreur reste stable, peu importe le nombre de capteurs.

Les performances de LMS en présence de diaphonie ont aussi été vérifiées. La simulation effectuée montre que bien que l'erreur totale converge à $0,036 A_{RMS}$ après 300 échantillons, les coefficients trouvés ne sont pas les mêmes que les originaux. La moyenne de l'erreur moyenne par capteur obtenue est de $2,01 A_{RMS}$.

L'algorithme LMS est donc en mesure de converger à une erreur presque nulle pour un nombre de capteurs variant entre 3 et 30 lorsqu'il n'y a pas de bruit ajouté aux mesures. Par contre, si du bruit est ajouté aux mesures, l'algorithme LMS ne converge pas à 0 et atteint le plancher de bruit. L'objectif initial d'une erreur moyenne par capteur $< 1 A_{RMS}$ devient donc tributaire de la conception des capteurs et de leur niveau de bruit intrinsèque. La dernière section de ce chapitre démontre que l'algorithme LMS n'est pas en mesure de corriger le problème de diaphonie même dans le cas le plus simple.

CHAPITRE 4

SOLUTION UTILISANT L'ALGORITHME FASTICA POUR LA CALIBRATION DES CAPTEURS

4.1 Introduction

Les résultats de simulation avec l'algorithme LMS ont montré de bonnes performances pour retrouver le gain des capteurs, mais ces performances chutent en présence de diaphonie. Ce chapitre présente les résultats de simulation avec l'algorithme FastICA, qui, de par sa nature, contribue à améliorer les performances lorsqu'il y a mélange des sources (diaphonie).

Un retour sur le modèle de base d'ICA et sur les équations principales de l'algorithme FastICA est effectué au début de ce chapitre. Les solutions utilisées afin de résoudre les 2 ambiguïtés reliées à ICA sont présentées par la suite. Pour continuer, les résultats de simulation avec diaphonie pour 3 et 30 capteurs sont présentés. L'erreur moyenne par capteur pour 3 à 30 capteurs en incluant de la diaphonie est aussi présentée à la fin de la section 4.4.2.

L'algorithme FastICA utilisé lors des simulations est un logiciel pour l'environnement Matlab[®] fourni par l'inventeur de FastICA, Aapo Hyvärinen, ainsi que son groupe de recherche (Hugo Gävert, 2005).

4.2 Modélisation du système en vue de l'utilisation de FastICA

Le modèle de base d'ICA, qui est aussi le modèle de base du problème de diaphonie, est représenté selon :

$$\mathbf{x} = \mathbf{A}\mathbf{s} \quad (4.1)$$

où \mathbf{x} est le vecteur d'observations, \mathbf{A} est la matrice de mélange et \mathbf{s} le vecteur des ICs.

En se basant sur l'équation matricielle (3.7), le modèle général de cette équation matricielle peut être vu comme étant :

$$\begin{bmatrix} I_{m1} \\ \vdots \\ I_{mK} \end{bmatrix} = \begin{bmatrix} \alpha_{nn} & \cdots & \alpha_{iK} \\ \vdots & \ddots & \vdots \\ \alpha_{Kj} & \cdots & \alpha_{KK} \end{bmatrix} \begin{bmatrix} I_1 \\ \vdots \\ I_K \end{bmatrix} \quad (4.2)$$

où n représente le numéro du capteur, K est le nombre de capteurs total dans le système, $i = 1, 2, \dots, K$ représentant le numéro de la ligne actuelle de la matrice et $j = 1, 2, \dots, K$ représentant le numéro de la colonne actuelle de la matrice, I_{mn} est le courant mesuré du capteur n , I_n est le courant réel du capteur n , α_{nn} sont les coefficients principaux des capteurs situés sur la diagonale principale de la matrice tandis que les coefficients α_{ij} , où $i \neq j$, sont les coefficients de diaphonie.

4.3 Solutions proposées aux problèmes d'ambiguïté

Tel que présenté au chapitre 2, FastICA permet d'extraire la diaphonie des signaux mélangés, mais pose certains problèmes, car il ne permet pas de retrouver l'amplitude et l'ordre initial des signaux. Nous proposons d'utiliser l'algorithme LMS afin de redimensionner correctement l'amplitude des signaux en utilisant l'amplitude du courant total précis comme référence.

Ensuite, puisque dans notre cas les coefficients principaux des capteurs α_{nn} sont plus élevés que les coefficients de diaphonie α_{ij} , ils doivent nécessairement être situés sur la diagonale principale de la matrice de séparation. Les lignes de la matrice de séparation doivent donc être permutées jusqu'à ce que tous les coefficients principaux forment la diagonale principale de celle-ci. Les courants réels vont donc être trouvés en 4 étapes telles que montrées à la figure 4.1.

La moyenne est rajoutée aux données une fois la convergence atteinte avec FastICA puisque les données ont été centrées afin d'avoir une moyenne nulle avant le blanchiment.

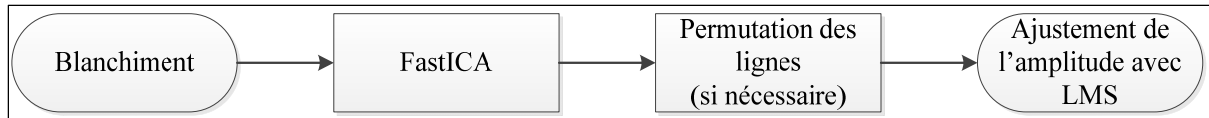


Figure 4.1 Schéma des étapes à suivre pour FastICA

4.4 Résultats de simulation avec l'algorithme FastICA

Avant la présentation des résultats de simulation avec FastICA, les paramètres utilisés pour le blanchiment, FastICA et LMS doivent être mentionnés. Les données sont tout d'abord blanchies avec la méthode EVD. La fonction non quadratique g utilisée est $g_3(y) = y^3$ puisqu'elle donne une bonne approximation de l'entropie négative selon Hyvärinen, Karhunen et Oja (2001) et que la dérivée de cette fonction est simple à calculer. La méthode d'orthogonalisation symétrique est utilisée puisqu'elle donne des résultats plus précis que celle par déflation (Hyvärinen, Karhunen et Oja, 2001).

Comme mentionné à la section précédente, les gains principaux des capteurs α_{nn} sont situés sur la diagonale principale de la matrice de mélange A tandis que les gains de diaphonies α_{ij} ($i \neq j$) sont situés dans les parties gauche et droite de la matrice. Le pire cas est supposé lors des simulations, soit que chaque capteur crée de la diaphonie à tous les autres capteurs.

Les gains principaux α_{nn} suivent une distribution normale de moyenne unitaire et leur écart-type $\sigma_{gp} = 0,16$ comme c'est le cas pour les gains des capteurs sans diaphonie. Les gains de diaphonie α_{ij} suivent une distribution uniforme dont la borne inférieure est $a = 0$ et la borne supérieure est $b = 0,2$ afin que l'effet de diaphonie d'un capteur sur un autre capteur varie uniformément entre 0 et 20%. Ces paramètres ne changent pas pour toutes les simulations qui suivront.

Les autres paramètres utilisés en simulation sont les mêmes qu'au chapitre 3, soit le nombre de capteurs K , le nombre d'échantillons T , le pas d'adaptation de l'algorithme LMS μ , l'écart-type du bruit σ_n et le nombre d'échantillons qui sont utilisés vers la fin de la

convergence de LMS afin de faire une moyenne de ceux-ci pour déterminer les coefficients $\hat{\mathbf{w}}$ finaux est représenté par $\hat{\mathbf{N}}$. Le bruit ajouté aux signaux mesurés est encore une fois représenté comme du bruit blanc gaussien. Du bruit blanc gaussien est aussi ajouté au courant total précis, mais celui-ci est considéré comme étant 10 fois moins élevé comparé à celui ajouté aux signaux mesurés. Les ICs sont donc équivalentes aux signaux réels \mathbf{I} , soit des signaux générés aléatoirement suivant une distribution uniforme avec une amplitude maximale de 20A comme au chapitre 3. Afin que FastICA fonctionne correctement, les ICs doivent avoir une distribution non gaussienne et être indépendantes les unes des autres. La méthode de Monte-Carlo est encore utilisée afin que les résultats soient le plus représentatifs possible.

L'erreur RMS entre les ICs initiales et celles estimées est présentée dans chaque sous-section sans calibration, seulement avec FastICA et avec la combinaison de FastICA et LMS. Les résultats seulement avec LMS sans diaphonie et avec diaphonie sont aussi présentés. Du bruit est ajouté aux mesures afin de voir la variation de l'erreur avec et sans l'ajout de bruit. Comme au chapitre 3, les résultats sont présentés pour 3 et 30 capteurs ainsi que l'erreur moyenne par capteur pour 3 à 30 capteurs à la fin de la section 4.4.2.

4.4.1 Calibration de 3 capteurs avec FastICA

Les résultats de la calibration de 3 capteurs avec l'algorithme FastICA sont présentés comme preuve de concept afin de tester FastICA dans un cas simple.

Les paramètres pour les premiers résultats sans ajout de bruit sont fixés comme suit : $K = 3$, $T = 10000$, $\sigma_n = 0$, $\hat{\mathbf{N}} = 500$ et $\mu = 0,001$.

Le tableau 4.1 montre l'erreur par capteur sans ajout de bruit pour 100 simulations Monte-Carlo. L'erreur par capteur pour la calibration seulement avec FastICA au tableau 4.1 est très élevée. On peut effectivement voir qu'une erreur moyenne de 9,55 A_{RMS} est obtenue, ce qui est plus élevé que l'erreur moyenne de base sans calibration qui est 2,60 A_{RMS} . Ce problème

d'amplitude est dû au premier problème d'ambiguïté. Par contre, si l'on redimensionne l'amplitude des signaux avec l'algorithme LMS en se basant sur l'amplitude du courant total précis après avoir enlevé la diaphonie avec FastICA, on obtient une erreur moyenne de 0,13 A_{RMS} . La combinaison de l'algorithme FastICA et de LMS enlève donc la diaphonie et redimensionne les signaux correctement tandis que FastICA seul enlève uniquement la diaphonie. L'algorithme FastICA a convergé en moyenne en 3 itérations en utilisant 10000 échantillons par itération et l'utilisation de LMS pour redimensionner les signaux a convergé en moyenne après 1000 échantillons. Contrairement aux résultats sans diaphonie et sans bruit pour 3 capteurs obtenus au chapitre 3, l'erreur ne tend pas vers 0.

Tableau 4.1 Erreur (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0$ pour 100 simulations

Type de calibration	Capteur 1	Capteur 2	Capteur 3	Moyenne
Sans calibration	2,61	2,49	2,70	2,60
FastICA seulement	9,54	9,55	9,54	9,55
FastICA + LMS	0,12	0,13	0,13	0,13
LMS seulement (avec diaphonie)	1,70	1,63	1,70	1,68
LMS seulement (sans diaphonie)	0,00019	0,00027	0,00018	0,00021

Pour les résultats avec l'ajout de bruit, les paramètres sont fixés selon : $K = 3$, $T = 10000$, $\sigma_n = 0,5$, $\hat{N} = 500$ et $\mu = 0,001$.

Le tableau 4.2 montre l'erreur par capteur avec l'ajout de bruit pour 100 simulations Monte-Carlo. En comparant les résultats du tableau 4.2 au tableau 4.1, on peut remarquer que l'erreur moyenne avec la combinaison de FastICA et LMS a augmenté dû à l'ajout de bruit. L'erreur moyenne est passée de 0,13 A_{RMS} sans ajout de bruit à 0,55 A_{RMS} avec bruit. FastICA a encore convergé en 3 itérations pour 10000 échantillons par itération et LMS a convergé en moyenne en 500 échantillons. On obtient donc une erreur moyenne de 0,55 A_{RMS} avec diaphonie et avec bruit pour 3 capteurs avec la combinaison de FastICA et LMS

par rapport à une erreur moyenne de 0,52 A_{RMS} sans diaphonie et avec bruit pour 3 capteurs avec LMS selon les résultats du chapitre 3.

Tableau 4.2 Erreur (A_{RMS}) pour 3 capteurs avec $\sigma_n = 0,5$ pour 100 simulations

Type de calibration	Capteur 1	Capteur 2	Capteur 3	Moyenne
Sans calibration	2,63	2,70	2,82	2,72
FastICA seulement	9,55	9,55	9,55	9,55
FastICA + LMS	0,55	0,55	0,54	0,55
LMS seulement (avec diaphonie)	1,73	1,76	1,80	1,76
LMS seulement (sans diaphonie)	0,52	0,52	0,51	0,52

4.4.2 Calibration de 30 capteurs avec FastICA

Les résultats ayant été présentés pour 3 capteurs, FastICA peut maintenant être appliqué au cas le plus complexe, soit 30 capteurs.

Les paramètres pour la calibration de 30 capteurs sans ajout de bruit sont donc : $K = 30$, $T = 10000$, $\sigma_n = 0$, $\hat{N} = 500$ et $\mu = 0,001$.

Le tableau 4.3 montre l'erreur moyenne des 30 capteurs ainsi que l'écart-type sans ajout de bruit pour 100 simulations Monte-Carlo. L'erreur sans calibration de 29,16 A_{RMS} montre que le niveau de diaphonie est rendu très élevé. Il est normal que l'erreur augmente avec l'ajout de capteurs puisque de la diaphonie est créée entre chacun d'eux. Ceci signifie donc que la matrice de mélange est passée d'une matrice de 3 par 3 à une matrice de 30 par 30 et que chaque signal contient donc plus de diaphonie que précédemment. Malgré cette erreur très élevée, la combinaison de l'algorithme FastICA et LMS réduit l'erreur à 0,51 A_{RMS} . L'erreur moyenne entre 3 et 30 capteurs sans ajout de bruit est donc passée de 0,13 A_{RMS} à 0,51 A_{RMS} . FastICA a convergé en 4 itérations au lieu de 3 cette fois-ci et LMS aux alentours de 3000 échantillons. Contrairement aux résultats sans diaphonie et sans bruit obtenus pour 30 capteurs dans le chapitre 3, l'erreur n'est plus pratiquement nulle.

Tableau 4.3 Erreur moyenne (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0$ pour 100 simulations

Type de calibration	Moyenne de l'erreur des capteurs (A_{RMS})	Écart-type de l'erreur des capteurs
Sans calibration	29,16	0,38
FastICA seulement	9,55	0,0053
FastICA + LMS	0,51	0,024
LMS seulement (avec diaphonie)	5,10	0,078
LMS seulement (sans diaphonie)	0,00088	0,00028

Afin d'ajouter du bruit aux mesures avec 30 capteurs, les paramètres sont fixés aux valeurs suivantes : $K = 30$, $T = 10000$, $\sigma_n = 0,5$, $\hat{N} = 500$ et $\mu = 0,001$.

Le tableau 4.4 présente l'erreur moyenne pour 30 capteurs ainsi que l'écart-type avec l'ajout de bruit aux mesures pour 100 simulations Monte-Carlo. L'erreur après calibration avec la combinaison de FastICA et LMS est passée de 0,51 A_{RMS} sans bruit à 0,83 A_{RMS} avec bruit. L'erreur a encore augmenté avec l'ajout du bruit, mais reste $< 1 A_{RMS}$ même avec 30 capteurs. L'erreur avec bruit est donc passée de 0,55 A_{RMS} pour 3 capteurs à 0,83 A_{RMS} pour 30 capteurs. FastICA a convergé en 4 itérations et LMS aux alentours de 1500 échantillons. On obtient donc une erreur moyenne avec diaphonie et avec bruit pour 30 capteurs de 0,83 A_{RMS} par rapport à une erreur moyenne de 0,54 A_{RMS} sans diaphonie et avec bruit selon les résultats obtenus avec LMS au chapitre 3.

Tableau 4.4 Erreur moyenne (A_{RMS}) pour 30 capteurs avec $\sigma_n = 0,5$ pour 100 simulations

Type de calibration	Moyenne de l'erreur des capteurs (A_{RMS})	Écart-type de l'erreur des capteurs
Sans calibration	29,27	0,40
FastICA seulement	9,55	0,0066
FastICA + LMS	0,83	0,020
LMS seulement (avec diaphonie)	5,14	0,089
LMS seulement (sans diaphonie)	0,54	0,0027

Les résultats pour 3 à 30 capteurs en utilisant la combinaison de FastICA et LMS sont maintenant présentés.

Pour les résultats sans bruit, les paramètres suivants sont utilisés : $K = 3$ à 30, $T = 10000$, $\sigma_n = 0$, $\hat{N} = 500$ et $\mu = 0,001$. Pour les résultats avec bruit, $\sigma_n = 0,5$.

La figure 4.2 montre l'erreur moyenne par capteur pour 3 à 30 capteurs pour 100 simulations avec et sans ajout de bruit. On remarque que, contrairement aux simulations sans diaphonie, l'erreur par capteur augmente linéairement selon le nombre de capteurs. Cette différence est normale, car plus il y a de capteurs au total, plus il y a de diaphonie. Le nombre de coefficients à trouver sans diaphonie est égal à n tandis qu'avec diaphonie il est égal à n^2 , où n est le nombre total de capteurs dans le système. La complexité du problème augmente donc plus rapidement et rend la tâche plus difficile pour FastICA d'enlever complètement la diaphonie. La combinaison de FastICA et LMS réduit l'erreur à 0,13 A_{RMS} pour 3 capteurs et à 0,51 A_{RMS} pour 30 capteurs sans ajout de bruit. Avec l'ajout de bruit, l'erreur est réduite à 0,55 A_{RMS} pour 3 capteurs et à 0,83 A_{RMS} pour 30.

Malgré l'augmentation de l'erreur moyenne par capteur suite à l'ajout du bruit, ce niveau d'erreur est très bien considérant la complexité du problème à résoudre et que chaque capteur mesure généralement une amplitude maximale de 20A. Ceci signifie donc que même dans le pire des cas, soit pour 30 capteurs avec diaphonie et avec bruit et en considérant que chaque

capteur introduit de la diaphonie sur tous les autres capteurs (hypothèse pessimiste), l'erreur moyenne par capteur est $< 1 A_{RMS}$ et atteint l'objectif initial fixé par rapport à l'erreur.

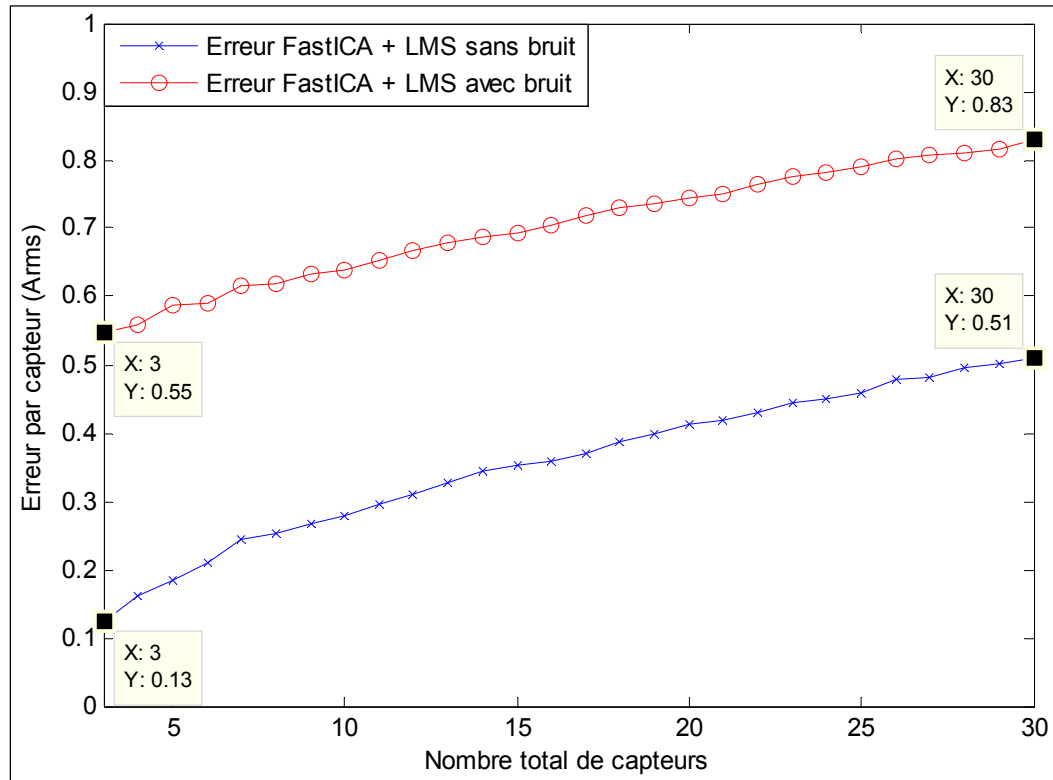


Figure 4.2 Erreur par capteur (A_{RMS}) pour 3 à 30 capteurs avec diaphonie et avec $\sigma_n = 0$ et $\sigma_n = 0,5$ pour 100 simulations pour chaque cas

4.5 Conclusion

Étant donné la proximité des capteurs dans le système, de la diaphonie vient affecter les mesures prises par ceux-ci. Puisque l'algorithme LMS n'est pas en mesure de réduire le problème de diaphonie, comme démontré au chapitre précédent, l'algorithme FastICA est utilisé afin d'extraire les courants réels des courants mesurés. Par contre, FastICA doit être utilisé en combinaison avec LMS afin de retrouver l'amplitude originale des signaux en plus d'extraire la diaphonie. Le chapitre est donc amorcé en faisant un retour sur l'algorithme FastICA en adaptant celui-ci afin qu'il soit représentatif du modèle de diaphonie propre à notre système.

Afin de tester le fonctionnement de FastICA dans le cas le plus simple, les résultats avec seulement 3 capteurs sont tout d'abord présentés. Les résultats montrent bien que la combinaison de FastICA avec l'algorithme LMS est nécessaire afin de redimensionner l'amplitude des signaux puisqu'une erreur de $9,54 A_{RMS}$ est obtenue en utilisant seulement FastICA comparé à une erreur de $0,13 A_{RMS}$ avec la combinaison des deux algorithmes sans l'ajout de bruit. L'utilisation de l'algorithme LMS sans FastICA et avec diaphonie réduit seulement l'erreur à $1,68 A_{RMS}$.

Le cas le plus complexe est ensuite testé, soit la calibration de 30 capteurs avec diaphonie. Malgré une erreur de $29,27 A_{RMS}$ sans calibration et avec bruit, la combinaison de FastICA et LMS est en mesure de réduire l'erreur à $0,83 A_{RMS}$.

Pour ce qui est de l'erreur obtenue avec la combinaison de FastICA et LMS pour un nombre total de capteurs variant entre 3 et 30, on remarque principalement que l'erreur augmente linéairement contrairement aux résultats obtenus au chapitre 3. L'erreur varie de $0,55 A_{RMS}$ pour 3 capteurs à $0,83 A_{RMS}$ pour 30 capteurs avec bruit et avec diaphonie par rapport à une erreur moyenne stable de $0,53 A_{RMS}$ sans diaphonie obtenue au chapitre 3 avec LMS.

Puisque le problème de diaphonie est difficile à résoudre et que chaque capteur mesure un signal ayant une amplitude maximale de $20A$, une erreur de $0,83 A_{RMS}$ dans le pire des cas, soit 30 capteurs avec bruit, est acceptable et l'objectif initial d'une erreur $< 1 A_{RMS}$ est respecté. La combinaison de l'algorithme FastICA et LMS est donc en mesure de diminuer le problème de diaphonie pour 3 à 30 capteurs en considérant que les ICs aient une distribution non gaussienne et soient indépendantes les unes des autres.

CHAPITRE 5

RÉSULTATS EXPÉRIMENTAUX

5.1 Introduction

Maintenant que les résultats de simulation avec LMS ont été présentés au chapitre 3 et les résultats de simulation avec FastICA au chapitre 4, les résultats expérimentaux obtenus avec ces deux algorithmes peuvent être présentés. Ce chapitre présente tout d'abord les deux installations qui sont testées expérimentalement afin de valider la calibration des mesures dans différentes situations, la précision des CTs, la consommation de chacun des appareils et le plancher de bruit de chacun des capteurs.

Par la suite, les résultats expérimentaux sans diaphonie (soit en ayant les capteurs éloignés les uns des autres) avec l'algorithme LMS sont présentés pour chacune des installations. Cinq tests d'une heure chacun sont effectués pour chaque installation.

Pour terminer, les résultats expérimentaux avec diaphonie sont présentés avec la combinaison de l'algorithme FastICA et LMS.

5.2 Présentation du système pour les tests expérimentaux

Les sous-sections qui suivent traitent des types d'installations mises en place ainsi que des appareils électriques utilisés pour chacune.

5.2.1 Présentation des types d'installations

Tout d'abord, les tests effectués sont limités à 3 capteurs puisque nous avons seulement accès à un disjoncteur de 20A pour les tests en laboratoire et que la consommation totale des 3 appareils utilisés en même temps est très près de 20A. Trois capteurs de courant à effet Hall sont utilisés afin de mesurer le courant consommé par les 3 appareils ainsi que 3 CTs afin

d'obtenir le courant réel consommé par ces appareils. Les mesures précises obtenues avec les CTs sont utilisées comme références afin de calculer l'erreur pour chacun des capteurs. La somme des 3 courants réels obtenus grâce aux CTs est utilisée comme courant total précis pour l'algorithme LMS. On utilise un CT de 200 ampères (EKM Metering Inc., 2013b). Le schéma d'une installation standard avec 3 capteurs est présenté à la figure 5.1. Le tableau 5.1 présente les appareils utilisés pour chacune des installations. Six plinthes consommant 0,7A chacune sont utilisées pour l'installation 2 et sont contrôlées grâce à un thermostat électronique utilisant un signal de modulation de largeur d'impulsion (PWM, *pulse width modulation*). La figure 5.2 et la figure 5.3 montrent ces deux installations.

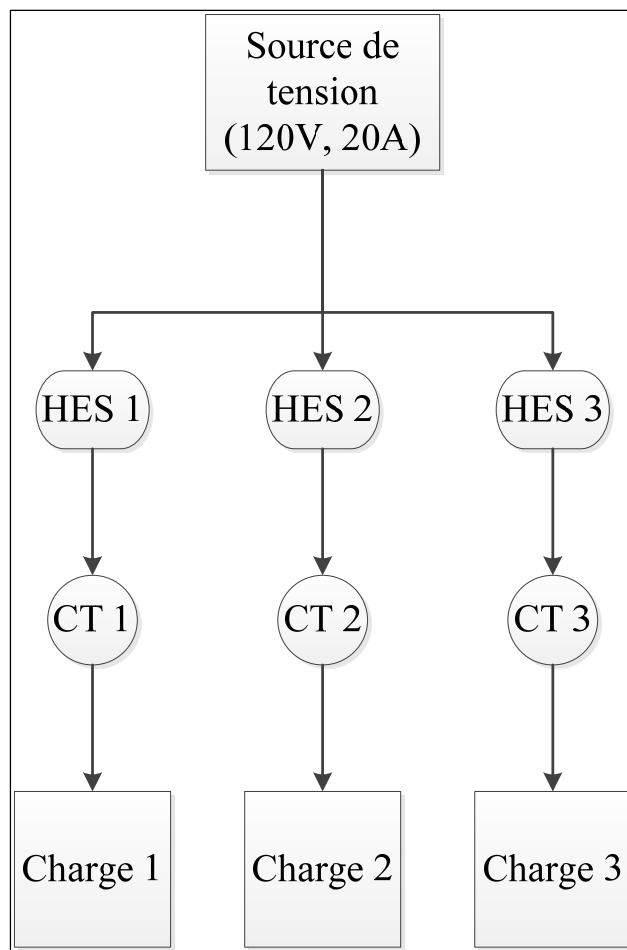


Figure 5.1 Schéma d'une installation typique

Tableau 5.1 Types d'appareils
pour chaque installation

Installation 1	Installation 2
Cafetière (975W)	Cafetière (975W)
Fusil à air chaud (1500W)	Fusil à air chaud (1500W)
Purificateur d'air (204W)	Plinthes et thermostat électronique (504W)



Figure 5.2 Installation 1



Figure 5.3 Installation 2

5.2.2 Précision des CTs et consommation des appareils

La précision des CTs de 200A selon le courant mesuré à l'entrée est présentée au tableau 5.2 selon la fiche technique du manufacturier (EKM Metering Inc., 2013b).

Tableau 5.2 Précision des CTs

Courant à l'entrée (A)	4	10	20	40	100	160	200	240
Précision (%)	0,110	0,142	0,178	0,202	0,159	0,0076	0,016	-0,054

La consommation de courant de chaque appareil est tout d'abord vérifiée avec un multimètre muni d'une pince ampèremétrique (Fluke, 2014) montré à la figure 5.4. Le tableau 5.3 montre le résultat de ces mesures, où, dans le cas des appareils pouvant être utilisés à plusieurs puissances, F1, F2 et F3 montre le niveau de puissance testé.



Figure 5.4 Pince multimètre TRMS AC Fluke 365

Tableau 5.3 Consommation des appareils

	Appareils de mesure										
	Cafetière		Fusil à air chaud			Purificateur d'air				Plinthes	
État	ON	OFF	F1	F2	OFF	F1	F2	F3	OFF	Pointe maximale	OFF
Fluke 365 (A)	7	0	5,6	10	0	0,6	0,8	1	0	4,2	0

5.2.3 Tests pour mesurer le plancher de bruit des capteurs

Afin de réduire l'impact du bruit dans le système, les mesures prises par les 3 capteurs de courant qui sont utilisés pour faire les tests sont enregistrées sans aucun appareil en marche pendant 1 heure afin de mesurer le plancher de bruit moyen de chacun.

Le tableau 5.4 montre la valeur moyenne du plancher de bruit de chaque capteur pour 1 heure de test. Les valeurs RMS du bruit pour chacun des capteurs sont utilisées afin de soustraire la puissance du plancher de bruit des mesures obtenues pour ainsi réduire l'erreur de chaque capteur.

Tableau 5.4 Valeurs (A_{RMS}) du plancher de bruit pour chaque capteur

Capteur 1	Capteur 2	Capteur 3
0,81	1,06	0,79

5.3 Résultats des tests expérimentaux avec LMS

L'algorithme LMS est utilisé afin de calibrer les mesures prises par les capteurs en se basant sur le courant total précis fourni par la somme des CTs. Cette section présente les résultats de 5 tests avec l'installation 1 ainsi que 5 tests avec l'installation 2 en ayant les 3 capteurs éloignés les uns des autres. Chacun des tests est d'une durée d'une heure et consiste à utiliser aléatoirement les 3 appareils comme c'est le cas dans une maison ou un appartement. Le schéma-bloc du système est présenté à la figure 1.1. Par contre, au lieu que ce soit l'UCT qui calibre les mesures reçues, elles sont acquises via internet et le traitement numérique ainsi que la calibration de celles-ci est effectuée avec le logiciel Matlab[®]. Pour le produit commercial en développement, ce serait l'UCT qui ferait le traitement numérique ainsi que la calibration des données. Les données prises par les CTs sont gérées grâce au EKM-OmniMeter I (EKM Metering Inc., 2013a) et sont extraites à l'aide du logiciel fourni par le fabricant.

Plusieurs traitements numériques sont effectués sur les données mesurées par les capteurs de courant avant la calibration pour différentes raisons. Premièrement, le plancher de bruit mesuré pour chacun des capteurs est soustrait du signal mesuré afin de réduire l'erreur près de 0 lorsqu'un appareil est éteint. Deuxièmement, la corrélation entre les signaux mesurés par les capteurs de courant et les CTs est effectuée afin de bien synchroniser les mesures.

Troisièmement, une moyenne glissante triangulaire (ce qui revient à filtrer les signaux) est aussi effectuée sur les données puisque cette méthode vient faire un double lissage des données. Finalement, une interpolation des données est effectuée afin que la base de temps soit la même pour les CTs et les HES et que celle-ci soit synchrone puisque les HES échantillonnent de manière asynchrone.

Pour les CTs, une moyenne glissante triangulaire ainsi qu'une interpolation est effectuée sur les signaux mesurés afin que la forme des signaux ainsi que le nombre d'échantillons concordent avec les signaux mesurés par les capteurs de courant.

Les paramètres à fixer pour la calibration avec l'algorithme LMS sont le nombre d'échantillons T désiré grâce à l'interpolation, le pas d'adaptation μ et \hat{N} le nombre d'échantillons qui sont utilisés vers la fin de la convergence de LMS afin de faire une moyenne des derniers coefficients trouvés pour déterminer les coefficients $\hat{\mathbf{w}}$ finaux. Le nouveau paramètre tma_{ratio} déterminant la grandeur de la fenêtre utilisée pour la moyenne glissante triangulaire doit aussi être fixé pour la calibration avec LMS des données expérimentales.

5.3.1 Résultats des tests avec l'installation 1 et LMS

Puisque le plancher de bruit des capteurs est d'environ un ampère et que le purificateur d'air consomme 1 ampère à vitesse maximale, les mesures prises par le HES pour cet appareil sont très bruitées. Cet appareil est choisi afin de montrer les limites de mesure des capteurs et aussi afin de montrer que l'algorithme LMS est en mesure de réduire l'erreur des autres capteurs même si les mesures prises par un en particulier sont très bruitées.

Pour les 5 tests, les paramètres sont fixés comme suit : $T = 5000$, $\mu = 0,00005$, $\hat{N} = 500$ et $tma_{ratio} = 75$.

La figure 5.5 montre un exemple des mesures des signaux pour le test 1 de l'installation 1. On peut constater que le signal mesuré par le capteur 3, soit le purificateur d'air, est très bruité et que LMS n'est pas en mesure de bien le calibrer étant donné que le bruit vient déformer la forme du signal. Par contre, les signaux de la cafetière et du fusil à air chaud mesurés par les capteurs 1 et 2 réussissent à être calibrés puisque seulement l'amplitude de ces signaux est différente par rapport aux signaux mesurés par les CTs. On peut voir à la figure 5.6 la moyenne de la convergence de l'erreur totale avec l'algorithme LMS pour les 5 tests. LMS a de la difficulté à bien converger à une erreur stable surtout due au signal mesuré du capteur 3 qui est très bruité. Puisque chaque test est d'une durée d'environ 1 heure et qu'une interpolation est effectuée afin d'obtenir 5000 échantillons, selon la figure 5.6 LMS converge aux alentours de 330 échantillons soit 3 minutes 58 secondes à une erreur totale de $0,44 A_{\text{RMS}}$.

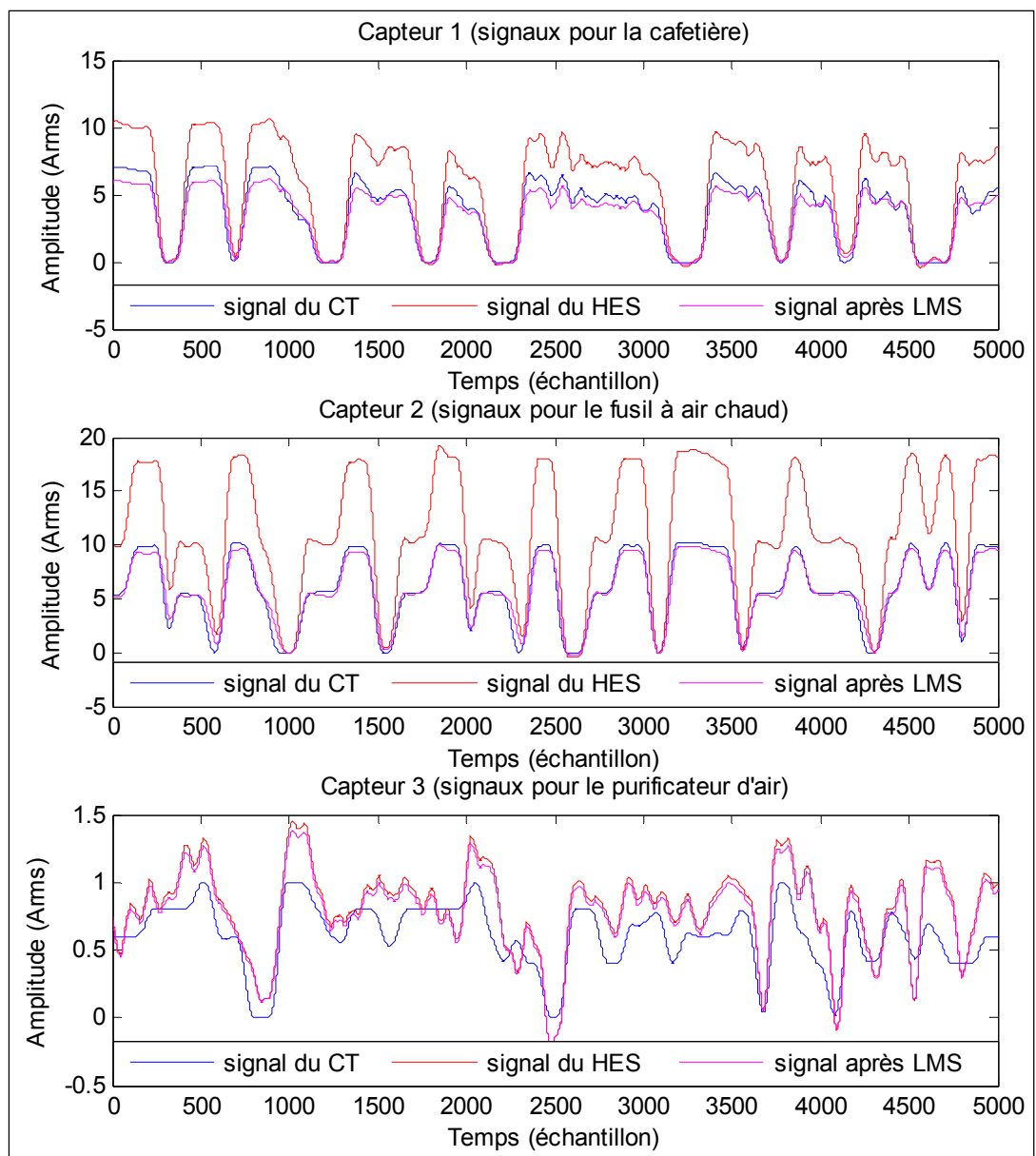


Figure 5.5 Exemple des signaux pour l'installation 1 avec LMS

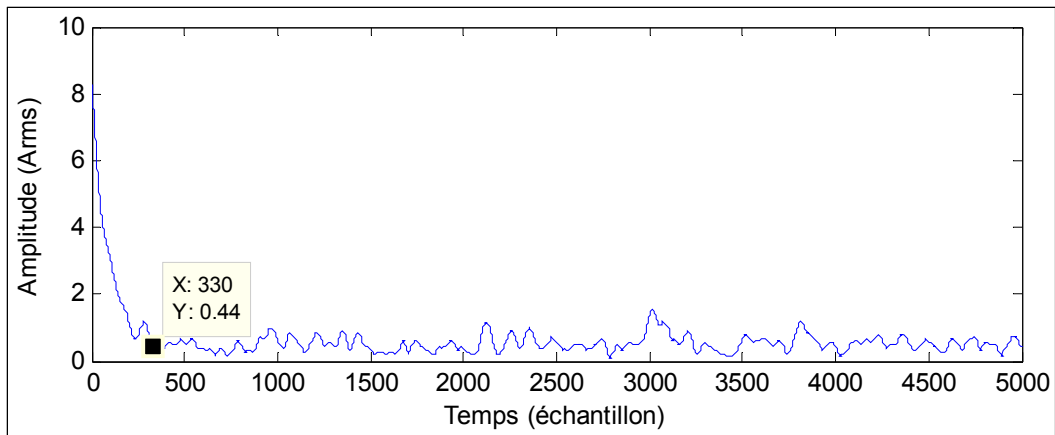


Figure 5.6 Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS de l'installation 1 pour 5 tests

Le tableau 5.5 montre l'erreur de chaque capteur pour les 5 tests avec l'installation 1 sans la calibration avec LMS tandis que le tableau 5.6 montre les mêmes résultats, mais avec calibration grâce à LMS. En comparant l'erreur moyenne par capteur des 3 capteurs après calibration de $0,47 A_{RMS}$ à l'erreur moyenne par capteur pour 3 capteurs en simulation avec bruit et après calibration de $0,52 A_{RMS}$ obtenue au chapitre 3, on peut constater que l'erreur est similaire et que le niveau de bruit ajouté en simulation correspond bien à celui en pratique. Malgré le fait que le signal du purificateur d'air est très bruité, LMS réussit tout de même à réduire considérablement l'erreur des deux autres capteurs, soit une réduction de $1,85 A_{RMS}$ pour le capteur 1 et de $5,21 A_{RMS}$ pour le capteur 2.

Tableau 5.5 Erreurs (A_{RMS}) de l'installation 1 pour 5 tests sans LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	2,43	5,88	0,26	2,86
Résultat 2	2,47	5,38	0,28	2,71
Résultat 3	2,65	5,79	0,34	2,93
Résultat 4	2,34	5,78	0,35	2,82
Résultat 5	2,50	5,76	0,34	2,87
Erreur moyenne par capteur	2,48	5,72	0,31	2,84
Écart-type de l'erreur	0,11	0,19	0,038	0,12

Tableau 5.6 Erreurs (A_{RMS}) de l'installation 1 pour 5 tests avec LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	0,60	0,53	0,23	0,46
Résultat 2	0,78	0,61	0,25	0,55
Résultat 3	0,61	0,44	0,29	0,45
Résultat 4	0,64	0,48	0,28	0,46
Résultat 5	0,49	0,48	0,28	0,41
Erreur moyenne par capteur	0,62	0,51	0,27	0,47
Écart-type de l'erreur	0,10	0,067	0,025	0,065

5.3.2 Résultats des tests avec l'installation 2 et LMS

L'installation 2 consiste à remplacer le purificateur d'air par une charge de plinthes électriques contrôlées par un thermostat électronique. Pour les 5 tests, les valeurs des paramètres à fixer sont les mêmes que pour l'installation 1, soit : $T = 5000$, $\mu = 0,00005$, $\hat{N} = 500$ et $tma_{ratio} = 75$.

La figure 5.7 montre un exemple des mesures des signaux pour le test 1 de l'installation 2. Comme mentionné précédemment, le thermostat électronique contrôlant les 6 plinthes utilise un signal PWM qui varie parfois plus rapidement que la fréquence de mesure des HES (qui est de 3,08 Hz pour optimiser la durée de vie de la pile du capteur Levasseur (2013)), ce qui crée parfois une différence entre le signal du CT et du HES pour le capteur 3 à la figure 5.7. La moyenne de convergence de l'erreur totale pour 5 tests avec l'algorithme LMS pour l'installation 2 est présentée à la figure 5.8. LMS converge aux alentours de 350 échantillons à une erreur totale de 0,49 A_{RMS} comparé à 330 échantillons et une erreur totale de 0,44 A_{RMS} pour l'installation 1, ce qui est similaire. L'algorithme converge donc en 4 minutes 12 secondes pour l'installation 2.

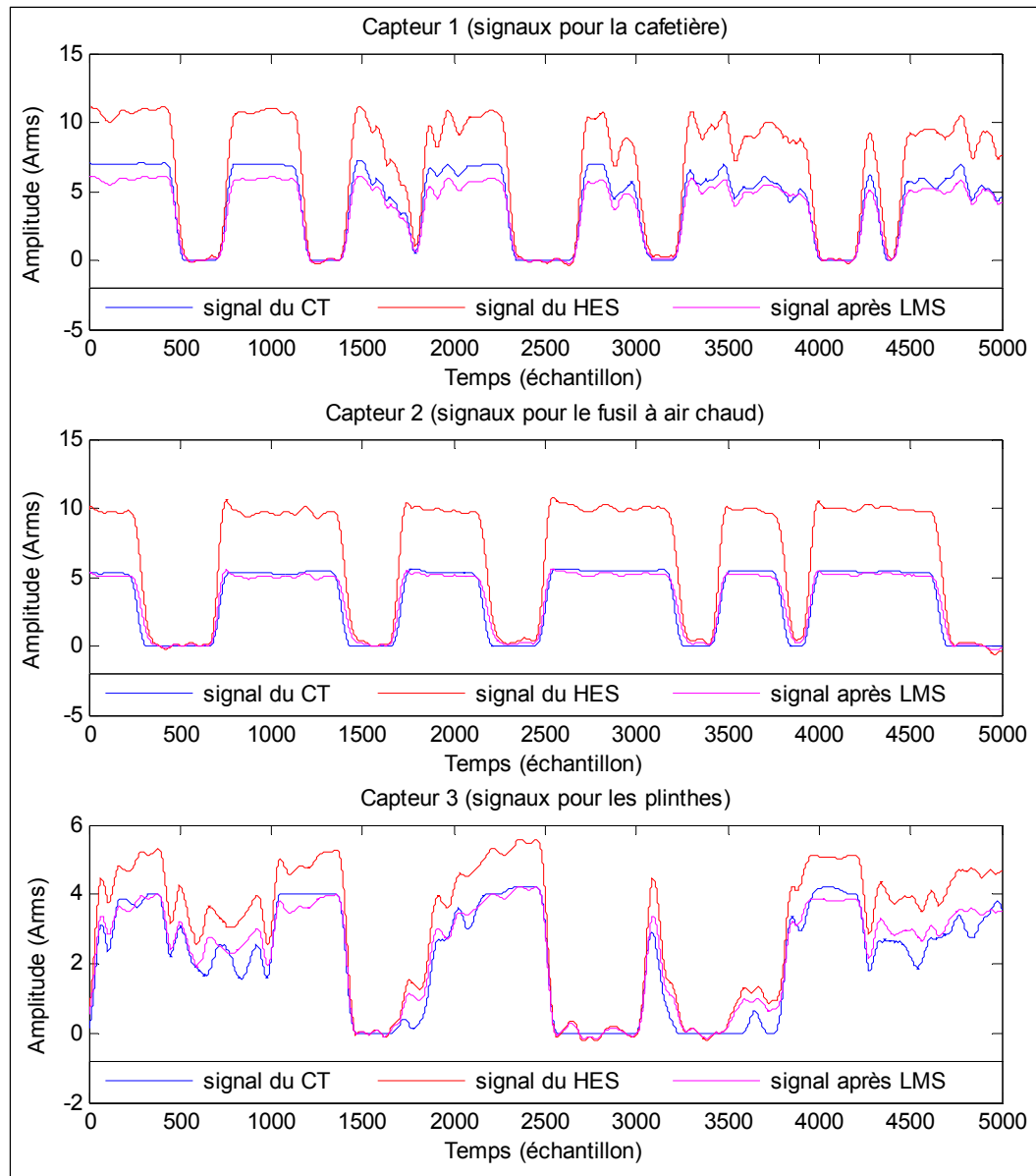


Figure 5.7 Exemple des signaux pour l'installation 2 avec LMS

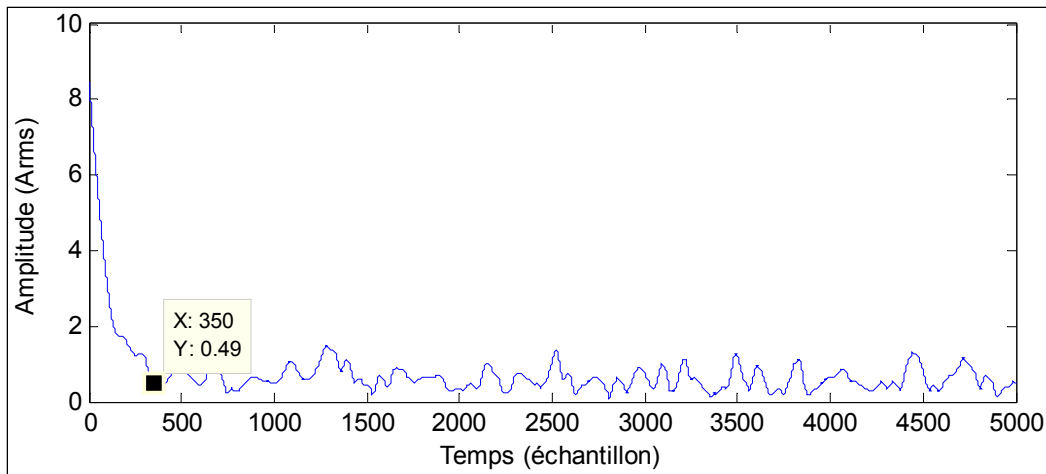


Figure 5.8 Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS de l'installation 2 pour 5 tests

Au tableau 5.7, on peut observer l'erreur de chacun des capteurs pour les 5 tests avec l'installation 2 sans calibration tandis que le tableau 5.8 montre les résultats avec calibration. Cette fois-ci l'erreur moyenne par capteur des 3 capteurs est de $0,49 A_{RMS}$ pour l'installation 2 au lieu de $0,47 A_{RMS}$ pour l'installation 1, ce qui est semblable. Ces deux erreurs moyennes expérimentales sont donc près de l'erreur moyenne par capteur de $0,52 A_{RMS}$ obtenue en simulation pour 3 capteurs avec bruit au chapitre 3. Pour les installations testées, l'algorithme LMS est donc en mesure de calibrer les capteurs du système sans diaphonie.

Tableau 5.7 Erreurs (A_{RMS}) de l'installation 2 pour 5 tests sans LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	2,91	3,78	1,04	2,58
Résultat 2	2,97	3,73	1,05	2,58
Résultat 3	2,92	3,97	1,12	2,67
Résultat 4	2,83	3,72	1,04	2,53
Résultat 5	2,68	3,76	1,04	2,49
Erreur moyenne par capteur	2,86	3,79	1,06	2,57
Écart-type de l'erreur	0,11	0,10	0,037	0,083

Tableau 5.8 Erreurs (A_{RMS}) de l'installation 2 pour 5 tests avec LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	0,73	0,49	0,37	0,53
Résultat 2	0,43	0,55	0,36	0,45
Résultat 3	0,49	0,66	0,45	0,53
Résultat 4	0,56	0,53	0,33	0,47
Résultat 5	0,49	0,45	0,44	0,46
Erreur moyenne par capteur	0,54	0,54	0,39	0,49
Écart-type de l'erreur	0,11	0,078	0,051	0,081

5.4 Résultats des tests expérimentaux avec FastICA

Lors de l'installation du système à l'intérieur de maisons dans le passé, un problème important de diaphonie avait été détecté. En laboratoire, plusieurs tests ont été effectués en rapprochant les 3 capteurs sans toutefois parvenir à recréer un niveau significatif de diaphonie entre les fils.

Ainsi, les mêmes mesures expérimentales obtenues pour les 5 tests de la section précédente sont utilisées et l'effet de la diaphonie est émulé à l'aide du logiciel Matlab[®]. La combinaison de FastICA et LMS est ensuite utilisée afin de faire la calibration des capteurs.

Les traitements numériques appliqués aux données sont les mêmes que précédemment, soit soustraction de la puissance du plancher de bruit, corrélation, moyenne glissante triangulaire et interpolation.

Afin de simuler l'effet de diaphonie, les mesures sont multipliées à une matrice de mélange. Puisque le vecteur des courants mesurés par les capteurs I_m a été affecté par du bruit, on vient aussi introduire de la diaphonie sur les bruits. Cependant, puisque les mesures ont déjà été affectées par un gain inconnu et que seulement de la diaphonie doit être ajoutée, la diagonale principale de chaque matrice de mélange générée aléatoirement pour chacun des tests est unitaire. Un exemple de l'ajout de diaphonie à l'aide d'une matrice de mélange aléatoire se traduit comme suit pour 3 capteurs :

$$\begin{bmatrix} I_{m1_cross} \\ I_{m2_cross} \\ I_{m3_cross} \end{bmatrix} = \begin{bmatrix} 1 & 0,1804 & 0,0400 \\ 0,1127 & 1 & 0,1664 \\ 0,0969 & 0,1739 & 1 \end{bmatrix} \begin{bmatrix} I_{m1} \\ I_{m2} \\ I_{m3} \end{bmatrix} \quad (5.1)$$

où I_m est le vecteur des courants mesurés par les capteurs et I_{m_cross} est le vecteur des courants mesurés par les capteurs avec l'ajout de diaphonie.

Les paramètres utilisés précédemment pour la calibration avec LMS sont encore utilisés, mais de nouveaux sont ajoutés afin de contrôler la matrice de mélange.

Afin que l'algorithme FastICA soit en mesure d'extraire les ICs des signaux mélangés, il faut que les ICs soient indépendantes les unes des autres et que leur distribution soit non gaussienne. Puisque la distribution des données est très importante, un exemple de cette distribution pour chacun des appareils de l'installation 1 et 2 avant l'ajout de diaphonie est présenté à la section suivante.

Le même niveau de diaphonie que lors des simulations effectuées au chapitre 4 est ajouté aux mesures soit en générant des coefficients de diaphonie a_{ij} selon une distribution uniforme dont la borne inférieure $a = 0$ et la borne supérieure $b = 0,2$. L'orthogonalisation symétrique ainsi que la fonction non quadratique $g = y^3$ sont encore une fois utilisées avec l'algorithme FastICA comme pour les résultats de simulation du chapitre 4.

5.4.1 Résultats des tests avec l'installation 1 et FastICA

Les valeurs des paramètres sont fixées comme suit : $T = 5000$, $\mu = 0,001$, $\hat{N} = 500$, $tma_{ratio} = 75$.

La figure 5.9 montre un exemple de la distribution des données enregistrées par les HES pour chacun des trois appareils pour le test 1 de l'installation 1. On peut constater en analysant cette figure que la distribution des données de la cafetière et du fusil à air chaud n'est pas gaussienne. Par contre, la distribution des données du purificateur d'air est beaucoup plus près d'une distribution gaussienne et cela peut nuire énormément à l'algorithme FastICA. La raison principale de la distribution plutôt gaussienne du purificateur d'air est que le signal mesuré par le HES pour cet appareil est très bruité puisque celui-ci est en dessous du plancher de bruit du HES et que le bruit a normalement une distribution gaussienne. La distribution des données pour les autres tests varie un peu, mais reste très semblable à celle observée à la figure 5.9.

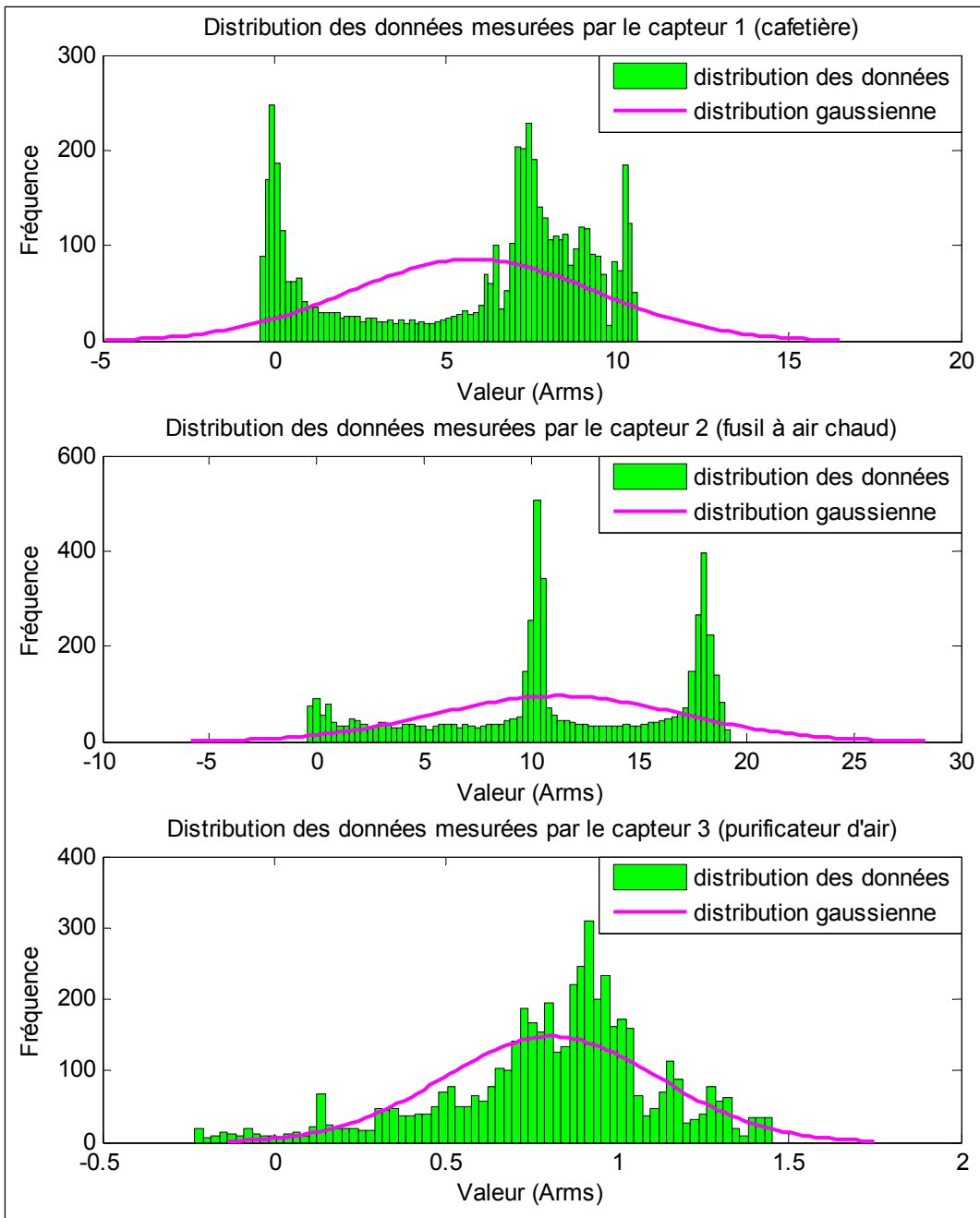


Figure 5.9 Exemple de la distribution des données provenant des capteurs HES pour l'installation 1

On peut voir à la figure 5.10 les signaux pour chaque appareil pour le test 1 de l'installation 1. Les signaux calibrés pour la cafetière et le fusil à air chaud sont semblables à ceux mesurés par les CTs, mais leur amplitude n'est pas la bonne. Pour ce qui est du signal calibré pour le

purificateur d'air, celui-ci n'est pas semblable à celui mesuré par le CT et son amplitude est beaucoup trop élevée. Le signal calibré du capteur 3 ressemble maintenant à celui du fusil à air chaud, ce qui fausse donc la calibration de l'amplitude avec LMS. FastICA n'a donc pas été en mesure d'extraire le bon signal pour le purificateur d'air dû au fait que le signal est très bruité et par conséquent a une distribution gaussienne. Ceci montre donc l'importance de la distribution des données utilisées avec FastICA.

Par contre, si seulement une des ICs a une distribution gaussienne, celle-ci peut normalement être extraite puisqu'elle n'est pas mélangée avec une autre composante gaussienne selon Hyvärinen, Karhunen et Oja (2001). Puisque le signal du purificateur d'air mesuré par le HES est très bruité, l'extraction n'est pas possible dans notre cas.

La figure 5.11 montre la convergence moyenne de l'erreur totale avec LMS une fois FastICA effectué avec l'installation 1. En observant la courbe de convergence, on peut constater que l'erreur totale est réduite, mais que l'algorithme n'est pas en mesure de converger vers des coefficients stables dus aux problèmes mentionnés précédemment. Le courant total mesuré par le CT est donc très différent du courant total des trois signaux obtenus après FastICA additionné ensemble. L'erreur totale est grandement réduite à partir de 200 échantillons à une erreur totale de $0,58 A_{RMS}$ ce qui donne une convergence en 2 minutes 24 secondes. FastICA a convergé en moyenne en 8 itérations en utilisant 5000 échantillons par itération pour les 5 tests.

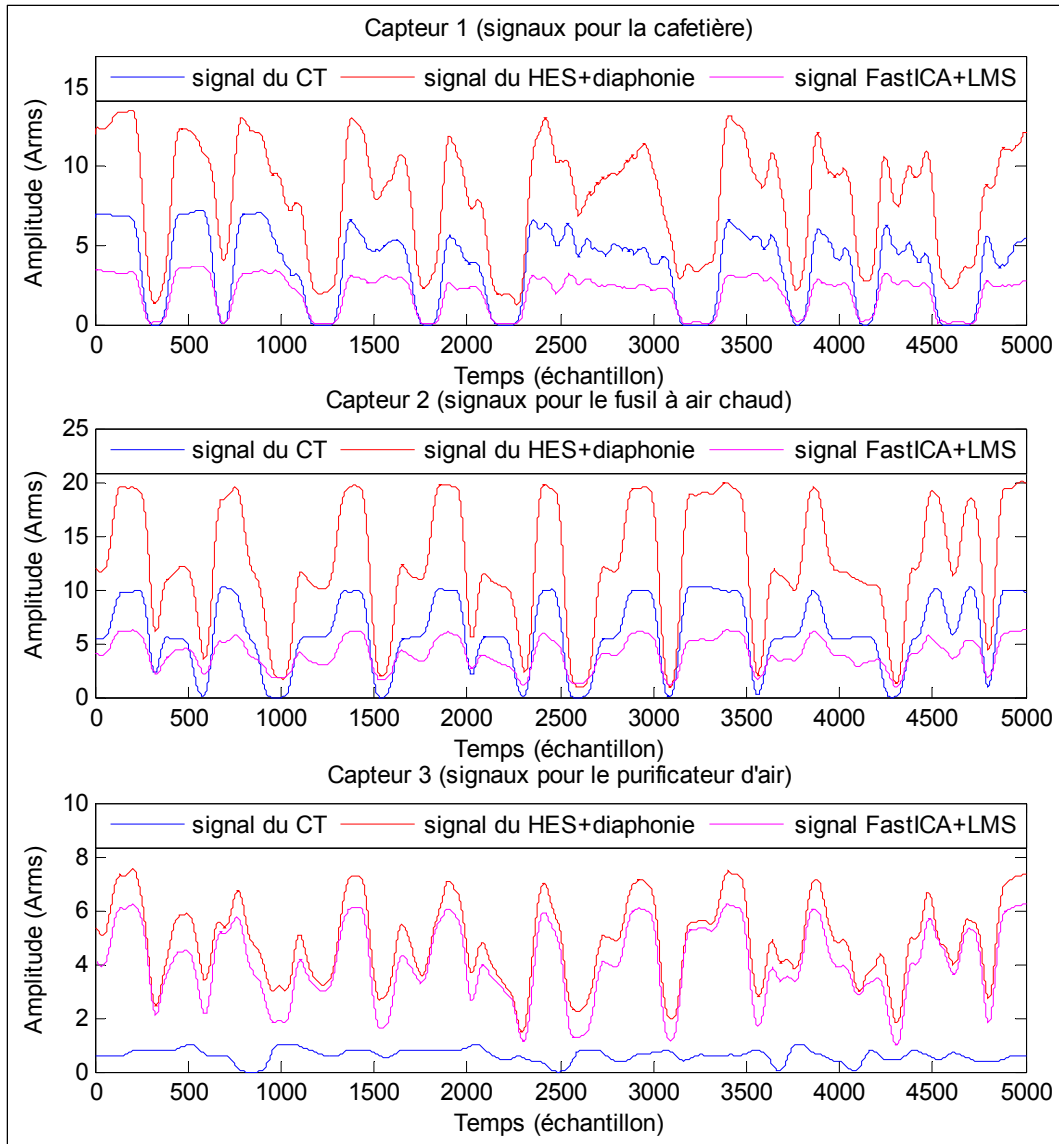


Figure 5.10 Exemple des signaux pour l'installation 1 avec FastICA et LMS incluant l'effet de la diaphonie

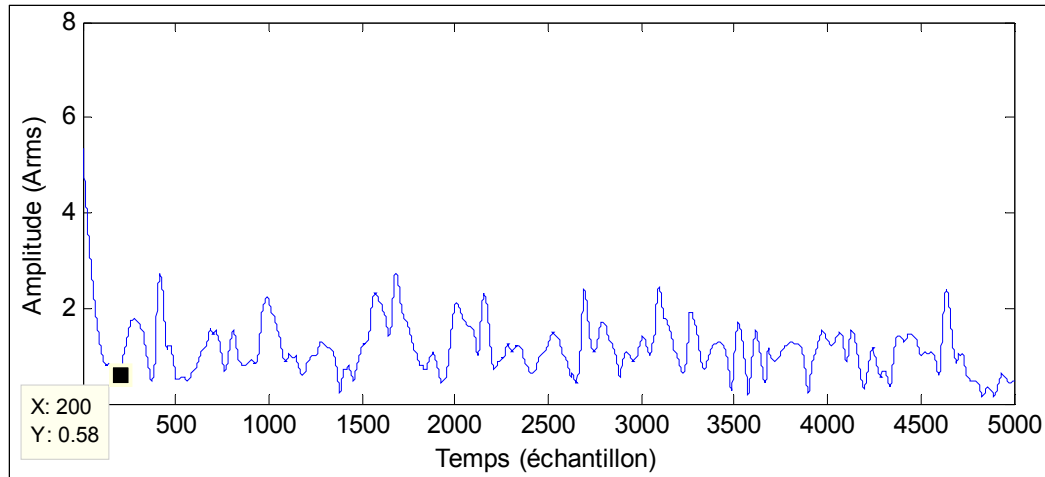


Figure 5.11 Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS après FastICA de l'installation 1 pour 5 tests

Le tableau 5.9 montre l'erreur pour chaque capteur et pour chacun des tests de l'installation 1 sans calibration tandis que le tableau 5.10 montre les mêmes résultats, mais après calibration avec FastICA et LMS. En comparant l'erreur par capteur avant calibration à celle après calibration, on peut constater que celle-ci est réduite de $3,22 A_{RMS}$ pour le capteur 1 et réduite de $2,80 A_{RMS}$ pour le capteur 2, mais augmentée de $0,12 A_{RMS}$ pour le capteur 3. Ces résultats ne sont pas surprenants considérant que le signal après calibration pour le capteur 3 extrait par FastICA n'est pas le bon, l'algorithme LMS n'est donc pas capable de converger à des coefficients stables comme on peut le voir à la figure 5.11. Une erreur moyenne par capteur pour les 3 capteurs de $3,16 A_{RMS}$ est obtenue après calibration avec FastICA et LMS pour l'installation 1 avec diaphonie comparée à une erreur de $0,55 A_{RMS}$ pour 3 capteurs avec bruit et diaphonie en simulation au chapitre 4. Ceci montre donc que la distribution des ICs est importante et qu'un signal ne doit pas être trop bruité puisque cela vient affecter l'erreur pour chacun des capteurs. Il ne faut donc pas utiliser de données mesurées en dessous du plancher de bruit des capteurs avec FastICA.

Tableau 5.9 Erreurs (A_{RMS}) de l'installation 1 pour 5 tests sans FastICA+LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	4,87	7,04	3,75	5,22
Résultat 2	4,23	6,57	3,21	4,67
Résultat 3	5,20	7,17	3,83	5,40
Résultat 4	4,53	6,86	3,97	5,12
Résultat 5	4,76	6,82	4,01	5,20
Erreur moyenne par capteur	4,72	6,89	3,76	5,12
Écart-type de l'erreur	0,37	0,23	0,32	0,30

Tableau 5.10 Erreurs (A_{RMS}) de l'installation 1 pour 5 tests avec FastICA+LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	2,14	2,75	3,72	2,87
Résultat 2	1,98	6,29	3,09	3,79
Résultat 3	0,61	3,35	3,08	2,35
Résultat 4	1,99	5,10	6,22	4,44
Résultat 5	0,77	2,95	3,30	2,34
Erreur moyenne par capteur	1,50	4,09	3,88	3,16
Écart-type de l'erreur	0,74	1,54	1,33	1,21

5.4.2 Résultats des tests avec l'installation 2 et FastICA

Les valeurs des paramètres sont fixées comme suit : $T = 5000$, $\mu = 0,001$, $\hat{N} = 500$, $tma_{ratio} = 75$.

La figure 5.12 montre un exemple de la distribution de la cafetière, du fusil à air chaud ainsi que des plinthes pour le test 1 de l'installation 2. La distribution pour la cafetière reste semblable à l'installation 1, mais celle pour le fusil à air chaud est différente puisque la vitesse maximale du fusil à air chaud n'est pas utilisée avec l'installation 2 pour ne pas

surcharger le disjoncteur puisque les plinthes consomment plus de courant que le purificateur d'air. On peut remarquer à la figure 5.12 que la distribution des données pour les plinthes n'est pas gaussienne contrairement au purificateur d'air de l'installation 1.

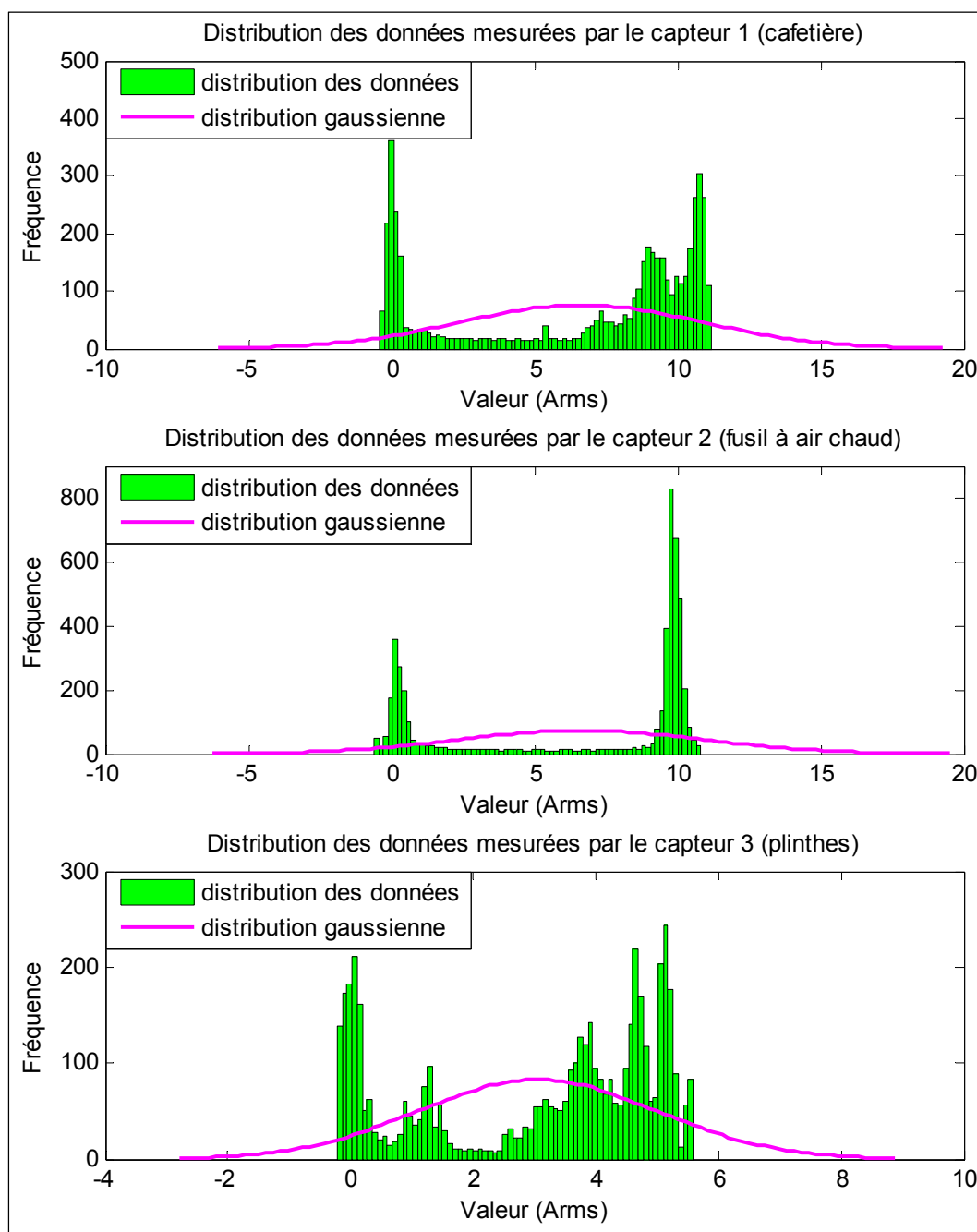


Figure 5.12 Exemple de la distribution des données provenant des capteurs HES pour l'installation 2

Les signaux pour chacun des appareils de l'installation 2 du test 1 sont montrés à la figure 5.13. Les signaux calibrés sont maintenant beaucoup plus près de ceux mesurés par les CTs au niveau de la forme et de l'amplitude si l'on compare avec la figure 5.10. Comme mentionnée précédemment, la calibration des plinthes est plus difficile que les autres appareils puisqu'il y a souvent des changements très rapides de la consommation de courant qui ne sont pas toujours enregistrés par les capteurs dus à leur vitesse d'échantillonnage de 3,08 Hz.

La convergence de la moyenne de l'erreur totale pour les 5 tests de l'installation 2 avec LMS après FastICA est montrée à la figure 5.14. L'erreur converge aux alentours de 175 échantillons donnant une erreur totale de 0,59 A_{RMS} . Le temps de convergence est donc de 2 minutes 6 secondes ce qui est semblable au temps de convergence de l'installation 1 de 2 minutes 24 secondes. La convergence des coefficients est beaucoup plus stable pour l'installation 2 que l'installation 1 si l'on compare avec la figure 5.11. La raison principale est que FastICA a réussi à extraire les bons signaux pour chacun des appareils, ce qui permet à LMS de converger plus facilement à des coefficients plus stables. FastICA a pour sa part convergé en moyenne en 7 itérations en utilisant 5000 échantillons par itération pour les 5 tests.

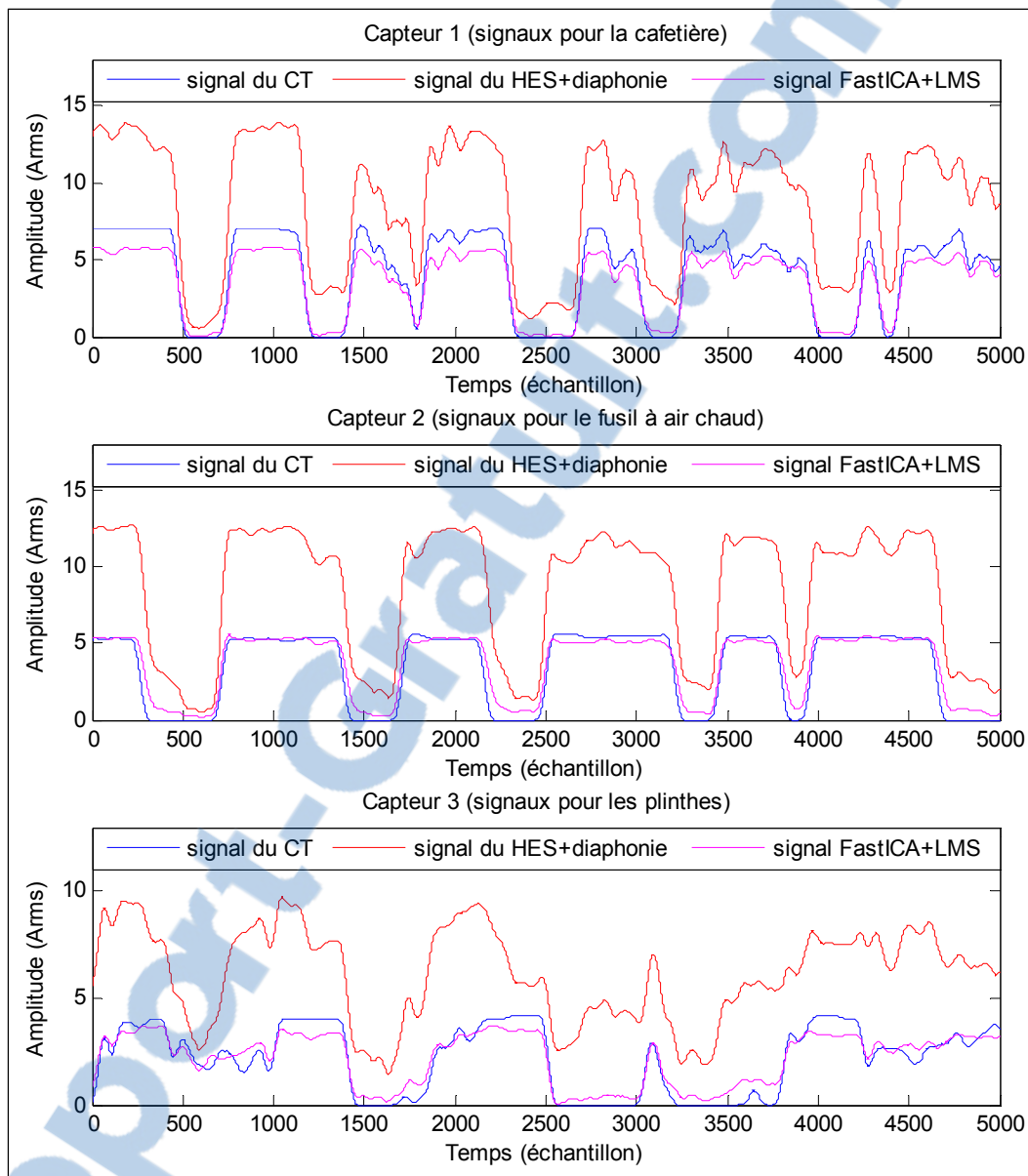


Figure 5.13 Exemple des signaux pour l'installation 2 avec FastICA et LMS incluant l'effet de la diaphonie

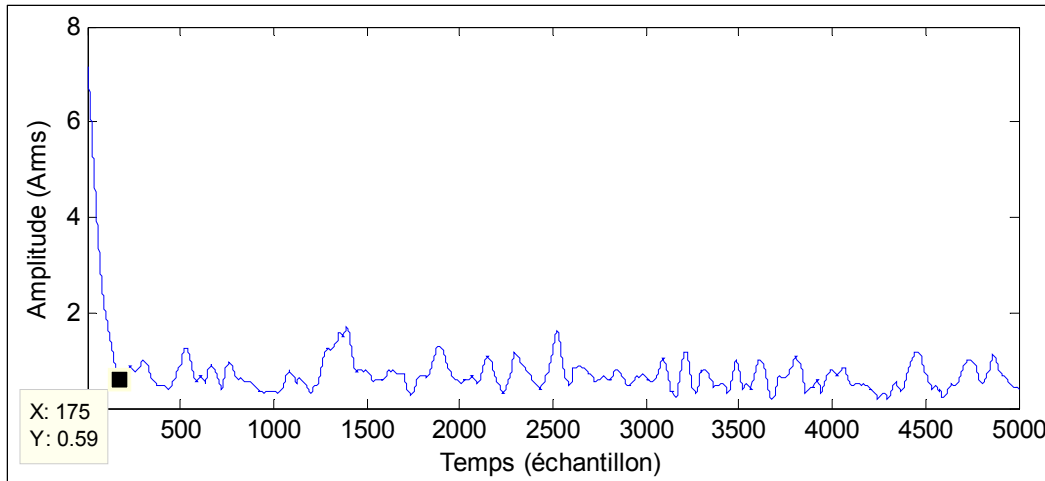


Figure 5.14 Convergence moyenne de l'erreur totale (A_{RMS}) avec LMS après FastICA de l'installation 2 pour 5 tests

On peut voir au tableau 5.11 l'erreur pour chaque capteur et chacun des tests sans calibration pour l'installation 2 tandis que le tableau 5.12 présente les mêmes résultats après calibration avec FastICA et LMS. En analysant les résultats, on peut constater que l'erreur moyenne par capteur des 3 capteurs après calibration pour l'installation 2 est beaucoup moins élevée que pour l'installation 1, soit $0,85 A_{RMS}$ au lieu de $3,16 A_{RMS}$. Par contre, l'écart-type moyen entre l'erreur des capteurs pour chacun des tests est plutôt élevé à $0,41$, ce qui signifie que la différence de l'erreur entre les tests est souvent importante. FastICA a donc parfois de la difficulté à bien extraire la diaphonie des signaux mélangés comme on peut le constater en regardant les résultats du test 3 où les moins bons résultats sont obtenus parmi les 5. Puisque les résultats du test 3 ne sont pas causés par la présence d'une quantité importante de bruit ou la distribution gaussienne d'un des appareils, l'indépendance entre les signaux lors de ce test n'a donc pas été suffisamment respectée et cela a diminué l'efficacité de FastICA à extraire les ICs des signaux mélangés. On obtient donc une erreur moyenne par capteur de $0,85 A_{RMS}$ avec diaphonie pour l'installation 2 comparée à une erreur de $0,55 A_{RMS}$ pour 3 capteurs avec bruit et diaphonie en simulation au chapitre 4.

Les différences principales entre la simulation et les résultats expérimentaux sont que les signaux en simulation ont tous une distribution uniforme parfaite, soit non gaussienne, sont

tous parfaitement indépendants les uns des autres et que la différence entre la forme des signaux sources et celle des signaux mesurés est beaucoup moins élevée qu'en pratique. Tous ces paramètres ont donc un impact beaucoup plus élevé sur l'efficacité de FastICA tandis que l'algorithme LMS est seulement affecté par la différence entre la forme des signaux sources par rapport à la forme des signaux mesurés.

Tableau 5.11 Erreurs (A_{RMS}) de l'installation 2 pour 5 tests sans FastICA+LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	4,77	5,48	4,25	4,83
Résultat 2	4,69	5,24	3,78	4,57
Résultat 3	4,82	5,92	4,32	5,02
Résultat 4	4,91	5,68	3,42	4,67
Résultat 5	4,56	5,74	3,40	4,57
Erreur moyenne par capteur	4,75	5,61	3,83	4,73
Écart-type de l'erreur	0,13	0,26	0,44	0,28

Tableau 5.12 Erreurs (A_{RMS}) de l'installation 2 pour 5 tests avec FastICA+LMS

	Capteur 1	Capteur 2	Capteur 3	Moyenne
Résultat 1	0,90	0,64	0,49	0,68
Résultat 2	0,44	0,77	0,36	0,52
Résultat 3	1,64	1,45	1,06	1,38
Résultat 4	0,94	0,55	1,39	0,96
Résultat 5	1,02	0,60	0,55	0,73
Erreur moyenne par capteur	0,99	0,80	0,77	0,85
Écart-type de l'erreur	0,43	0,37	0,43	0,41

5.5 Conclusion

Le chapitre est amorcé par la présentation des deux installations expérimentales utilisées pour les tests. La précision des CTs selon le courant mesuré à l'entrée ainsi que la consommation de courant de chacun des appareils sont ensuite présentées. La première section de ce chapitre est terminée en montrant la valeur du plancher de bruit pour chacun des capteurs utilisés.

Les résultats des tests expérimentaux avec l'algorithme LMS sans diaphonie, soit en ayant les capteurs éloignés les uns des autres, sont présentés dans la section suivante. Cinq tests d'une heure chacun sont effectués pour les deux installations en utilisant les appareils aléatoirement.

Une erreur moyenne par capteur pour les 3 capteurs sans calibration de $2,84 A_{RMS}$ comparée à une erreur de $0,47 A_{RMS}$ après calibration est obtenue pour l'installation 1. La calibration du signal du purificateur d'air est difficile puisque celui-ci est très bruité étant donné qu'il consomme moins d'un ampère pour chacune de ses vitesses, ce qui est en dessous du plancher de bruit des capteurs.

Pour ce qui est de l'installation 2, une erreur moyenne par capteur pour les 3 capteurs de $2,57 A_{RMS}$ sans calibration comparée à $0,49 A_{RMS}$ après calibration est obtenue. Cette fois-ci la calibration des plinthes est plus concluante puisque le signal mesuré est en haut du plancher de bruit des capteurs. Par contre, puisque le taux d'échantillonnage des HES est seulement de 3,08 Hz, les changements rapides de la consommation des plinthes contrôlées par le thermostat électronique sont difficiles à mesurer. Les objectifs de départ d'un temps de convergence < 5 minutes et d'une erreur moyenne par capteur $< 1 A_{RMS}$ sont respectés pour l'installation 1 et 2 avec LMS.

Suite aux résultats expérimentaux avec LMS, les tests avec la combinaison de FastICA et LMS sont ensuite effectués. Étant donné qu'aucune diaphonie n'a été détectée en pratique en

rapprochant les 3 capteurs, de la diaphonie a été ajoutée en simulation aux mesures expérimentales.

Pour l'installation 1, une erreur moyenne par capteur pour les 3 capteurs de 5,12 A_{RMS} sans calibration comparée à une erreur de 3,16 A_{RMS} après calibration est obtenue. Cette erreur très élevée après calibration est due au fait que le signal mesuré du purificateur d'air est très bruité et a donc une distribution qui est près d'une distribution gaussienne, ce qui empêche FastICA d'extraire le bon signal pour le purificateur d'air. FastICA a extrait un signal semblable à celui du fusil à air chaud pour le capteur 3 au lieu de celui du purificateur d'air, ce qui fausse la calibration de l'amplitude avec LMS. En théorie, si seulement un des ICs a une distribution gaussienne, FastICA est en mesure de l'extraire quand même. Par contre, cela n'a pas fonctionné dans notre cas dû au niveau de bruit élevé contenu dans le signal mesuré du purificateur d'air.

Pour l'installation 2, une erreur moyenne par capteur pour les 3 capteurs de 4,73 A_{RMS} sans calibration comparée à une erreur de 0,85 A_{RMS} après calibration est obtenue. La réduction de l'erreur pour l'installation 2 est plus grande comparée à l'installation 1 due au fait que la distribution des données des plinthes n'est pas gaussienne et que le signal mesuré est en haut du plancher de bruit des capteurs. Par contre, l'erreur pour certains tests est plus élevée que d'autres étant donné le non-respect de l'indépendance entre les ICs pour ces tests en particulier.

Pour terminer, l'algorithme FastICA a vraisemblablement de la difficulté à éliminer complètement la diaphonie des signaux mélangés. Cette difficulté provient du non-respect de la distribution des ICs, qui peut être parfois près d'une distribution gaussienne, et de l'indépendance entre les ICs qui n'est pas tout à fait respectée. L'objectif initial d'un temps de convergence < 5 minutes est respecté pour les deux installations, mais l'objectif d'une erreur moyenne par capteur $< 1 A_{RMS}$ est seulement respecté pour l'installation 2.

CONCLUSION

Les WSNs utilisés pour sensibiliser les usagers à leur consommation énergétique sont des outils très pratiques, mais pour que ce produit soit abordable pour le consommateur et facile à installer, le coût ainsi que la taille des capteurs doivent être minimisés. Un type de capteur qui répond aux deux critères précédents est le HES. Par contre, puisque la sensibilité du HES dépend de la proximité de celui-ci au fil qu'il mesure, des incertitudes sur le gain du capteur en découlent. La calibration numérique d'un tel système est donc primordiale afin d'offrir au client un produit compétitif au niveau du coût et aussi au niveau de la précision des mesures. C'est pourquoi des algorithmes numériques tels que LMS et FastICA peuvent être utilisés afin d'augmenter la précision des mesures sans coût additionnel sur le produit. L'objectif de ce travail est donc d'utiliser l'algorithme LMS et FastICA afin de calibrer un réseau de capteurs HES et d'amener l'erreur par capteur $< 1 A_{RMS}$ en moins de 5 minutes.

Durant ce projet, autant au niveau des simulations qu'au niveau expérimental, il est démontré que chacun des algorithmes a son utilité afin de réduire l'erreur de précision des capteurs. LMS permet de réduire l'erreur lorsqu'il y a ou non présence de bruit, mais seulement en l'absence de diaphonie. Il est montré qu'il est possible de diminuer l'erreur même en présence de diaphonie en combinant LMS à l'algorithme FastICA.

Des essais ont tout d'abord été réalisés en simulation. Il est démontré que l'erreur RMS moyenne par capteur reste stable pour un nombre de capteurs variant entre 3 et 30 et que l'erreur moyenne obtenue après calibration LMS est de $0,0008 A_{RMS}$ sans bruit et de $0,53 A_{RMS}$ lorsqu'un bruit de $0.50 A_{RMS}$ ($\sigma_n = 0,5$) est ajouté aux mesures. De plus, il est montré que l'algorithme LMS n'est pas en mesure de régler le problème de diaphonie. Tout de même, l'objectif initial d'une erreur par capteur $< 1 A_{RMS}$ est respecté dans tous les cas pour l'algorithme LMS en simulation.

Les simulations montrent aussi que l'utilisation de LMS suite à FastICA permet de calibrer l'amplitude des signaux. Contrairement aux résultats obtenus avec LMS, l'erreur moyenne

par capteur avec l'ajout de diaphonie n'est pas stable et augmente linéairement selon le nombre de capteurs dans le système. Après calibration, une erreur moyenne par capteur sans bruit de $0,13 A_{RMS}$ pour 3 capteurs allant jusqu'à $0,51 A_{RMS}$ pour 30 capteurs est obtenue tandis qu'avec l'ajout de bruit ($\sigma_n = 0,5$) une erreur de $0,55 A_{RMS}$ pour 3 capteurs allant jusqu'à $0,83 A_{RMS}$ pour 30 capteurs est obtenue. Encore une fois, l'objectif initial d'une erreur par capteur $< 1 A_{RMS}$ est respecté dans tous les cas pour la combinaison de FastICA et LMS en simulation.

L'efficacité des algorithmes est ensuite démontrée à l'aide de deux installations expérimentales. L'installation 1 est composée d'une cafetière, d'un fusil à air chaud et d'un purificateur d'air tandis que l'installation 2 est composée d'une cafetière, d'un fusil à air chaud et de 6 plinthes électriques branchées en parallèle contrôlées par un thermostat électronique.

Pour les résultats avec LMS sans diaphonie en ayant les capteurs éloignés les uns des autres, une erreur moyenne par capteur pour les 3 capteurs de $0,47 A_{RMS}$ est obtenue pour l'installation 1 ainsi qu'un temps de convergence de 3 minutes 58 secondes. Pour l'installation 2, une erreur de $0,49 A_{RMS}$ est obtenue ainsi qu'un temps de convergence de 4 minutes 12 secondes. Les résultats montrent qu'il est important que le signal mesuré par un capteur soit plus élevé que le plancher de bruit de celui-ci et que les capteurs ont de la difficulté à mesurer des variations rapides de courant. Les objectifs de départ d'une erreur moyenne par capteur $< 1 A_{RMS}$ et d'un temps de convergence < 5 minutes sont respectés pour les tests avec l'installation 1 et 2.

Pour les résultats avec la combinaison de FastICA et LMS, de la diaphonie est ajoutée en simulation aux mesures expérimentales. Une erreur moyenne par capteur pour les 3 capteurs de $3,16 A_{RMS}$ est obtenue avec l'installation 1 pour un temps de convergence de 2 minutes 24 secondes avec LMS suite à FastICA ainsi qu'une convergence en 8 itérations pour l'algorithme FastICA. Pour sa part, l'installation 2 présente une erreur de $0,85 A_{RMS}$ pour un temps de convergence de 2 minutes 6 secondes avec LMS et une convergence en 7 itérations

pour l'algorithme FastICA. Les mauvais résultats de l'installation 1 sont causés par le signal du purificateur d'air qui est très bruité. Ce bruit induit donc une distribution plutôt gaussienne pour ce signal et empêche FastICA d'extraire le bon IC des signaux mélangés. En théorie, si seulement un des ICs a une distribution gaussienne, FastICA est en mesure de l'extraire. Par contre, cela n'a pas fonctionné dans notre cas puisque le niveau de bruit contenu dans le signal du purificateur d'air est très élevé. LMS a donc été dans l'incapacité de calibrer l'amplitude des signaux par la suite. Pour ce qui est de l'installation 2, l'erreur obtenue pour certains tests est plus élevée comparée aux autres. Ce problème est causé par le non-respect de l'indépendance entre les ICs pour certains des tests et a diminué l'efficacité de FastICA à extraire la diaphonie des signaux mélangés. Les résultats des tests avec la combinaison de FastICA et LMS ont respecté l'objectif initial d'un temps de convergence < 5 minutes pour les deux installations, mais seulement l'erreur moyenne par capteur obtenue pour l'installation 2 a respecté l'objectif d'une erreur $< 1 A_{RMS}$.

Pour terminer, l'algorithme LMS a été en mesure de faire la calibration des capteurs pour les tests sans diaphonie autant en simulation qu'en pratique tandis que la combinaison de FastICA et LMS a réussi à calibrer les capteurs pour les tests avec diaphonie en simulation. Par contre, la calibration en milieu expérimentale n'a pas été aussi concluante dus à certains non-respects des critères nécessaires pour l'utilisation de FastICA, soit l'indépendance des composantes et la non-gaussianité de celles-ci. LMS est donc un algorithme simple, efficace et adaptatif pouvant faire la calibration sans diaphonie, mais requérant un choix avisé du pas d'adaptation. Pour sa part, FastICA est un algorithme ne nécessitant aucun pas d'adaptation, mais celui-ci est relativement complexe et non adaptatif pouvant régler le problème de diaphonie pourvu que les restrictions de base pour l'utilisation de cet algorithme soient respectées.

RECOMMANDATIONS

Certaines recommandations sont nécessaires pour l'utilisation de l'algorithme LMS et FastICA en pratique et la possibilité de les implémenter dans une UCT afin de faire la calibration des données en temps réel sans la nécessité d'un logiciel externe de traitement des données comme Matlab[®]. Quelques alternatives possibles aux algorithmes présentées dans ce mémoire sont aussi proposées.

Pour ce qui est de l'algorithme LMS, c'est un algorithme simple, adaptatif, sans restriction nécessaire pour son utilisation, ne demandant aucune opération d'inversion de matrice, mais n'étant pas en mesure de résoudre le problème de diaphonie. Par contre, il est important de bien choisir la valeur du pas d'adaptation μ afin que l'algorithme LMS converge correctement. Puisque le système n'a pas besoin de converger rapidement et que la précision des mesures est plus importante, un petit pas d'adaptation est recommandé. L'UCT actuellement utilisée est déjà très chargée au niveau des opérations et des traitements qu'elle doit effectuer pour le système actuel. Ceci signifie donc qu'un algorithme simple comme LMS est adéquat afin de ne pas trop surcharger davantage l'UCT. L'utilisation de l'algorithme LMS est envisageable dans un contexte où la présence de diaphonie est faible. Les résultats avec cet algorithme ont été concluants autant au niveau des simulations que des expérimentations.

Pour les alternatives à LMS, la calibration des capteurs pourrait être testée avec l'algorithme des moindres carrés normalisé (NLMS, *normalized least mean square*) (Haykin, 2003). L'avantage de l'algorithme NLMS comparé à LMS est que celui-ci est normalisé par rapport à la puissance à l'entrée. Ceci aide donc à la stabilité de l'algorithme puisque l'algorithme LMS est sensible au choix du paramètre μ .

Concernant l'algorithme FastICA, celui-ci est un algorithme relativement complexe, qui n'est pas adaptatif, ayant des restrictions très strictes pour son utilisation, nécessitant l'inversion de matrices, mais sans pas d'adaptation à configurer et en mesure de résoudre le problème de

diaphonie. Comme les résultats expérimentaux l'ont démontré, les restrictions avant l'utilisation de FastICA doivent être absolument respectées sinon les résultats sont grandement affectés. Les ICs doivent absolument être indépendantes les unes des autres et leur distribution ne doit pas être gaussienne. Dans le cas de ce système, FastICA ne peut pas être utilisé seul afin de calibrer les capteurs et doit être combiné à l'algorithme LMS, ce qui rend l'implémentation encore plus difficile en pratique. Puisque comme mentionné précédemment, l'UCT est déjà très chargée avec le traitement des mesures qu'elle reçoit ainsi que toutes les autres opérations qu'elle doit effectuer, l'implémentation de la combinaison de FastICA et LMS pourrait ralentir son fonctionnement. Par contre, la raison principale qui empêche la recommandation de l'utilisation de FastICA dans un cas réel est due aux restrictions associées avec l'utilisation de cet algorithme. Puisqu'on peut seulement supposer l'indépendance entre les ICs et que ceux-ci ne doivent pas avoir une distribution gaussienne, cela signifie donc que l'algorithme ne fonctionnera pas dans tous les cas en pratique. La combinaison de FastICA et LMS est donc seulement recommandée dans un cas où la présence de diaphonie serait très élevée et où l'algorithme LMS seul ne serait pas en mesure de faire la calibration des capteurs. D'autres tests avec l'utilisation de cet algorithme combiné à LMS dans un cas réel pour 20 à 30 capteurs seraient nécessaires afin d'avoir une meilleure idée de son efficacité. Pour ce qui est des algorithmes alternatifs à FastICA, il serait intéressant que ceux-ci soient plus simples au niveau des calculs et qu'ils aient moins de restrictions pour leur utilisation.

Les algorithmes AMUSE et SOBI (Cichocki, 2004; S. Chauhan, 2007) utilisent tous deux seulement des statistiques de deuxième ordre (SOS, *second order statistics*) contrairement à FastICA qui utilise des statistiques d'ordre supérieur (HOS, *higher order statistics*). Ces deux algorithmes ne requièrent pas l'indépendance statistique entre les ICs ainsi qu'une distribution non gaussienne de celles-ci comparées à FastICA qui impose de telles restrictions. Par contre, les ICs estimées par ces deux algorithmes ne sont pas forcément indépendantes, mais uniquement spatio temporellement décorréelées. L'évaluation des performances de chaque algorithme combiné à LMS serait donc très intéressante pour des recherches futures afin de tester s'ils sont en mesure de corriger le problème de diaphonie.

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

- Allegro Microsystems Inc. 2011. « Hall effect sensor (A1393SEHLT-T) ». < <http://www.allegromicro.com/en/Products/Magnetic-Linear-And-Angular-Position-Sensor-ICs/Linear-Position-Sensor-ICs/A1391-2-3-5.aspx> >. Consulté le 2013.
- Beaufort Samson, G., M. A. Levasseur, F. Gagnon et G. Gagnon. 2014. « Auto-calibration of Hall effect sensors for home energy consumption monitoring ». *Electronics Letters* (United Kingdom). 27 février 2014, p. 403-405.
- Biezl. 2009. « Current transformer ». < http://commons.wikimedia.org/wiki/File:Stromwandler_Zeichnung.svg >.
- Cichocki, A. 2004. « Frequently Asked Questions, Tips & Tricks for ICALAB ». < <http://www.bsp.brain.riken.jp/ICALAB/faq/faq.html> >.
- Current Cost. 2014. « ENVI ». < <http://www.currentcost.com/product-envi.html> >. Consulté le 4 février 2014.
- EKM Metering Inc. 2013a. « EKM-OmniMeter I v.3 Universal Smart Meter, Single Phase or 3-phase, 120 to 480V, 50/60Hz up to 5000 amps ». In *EKM Metering Inc.* < <http://www.ekmmetering.com/ekm-omnimeter-i-v-3-3-phase-3-wire-or-4-wire-120-to-480v-50-60hz-single-phase-2-or-3-wire-120v-to-480v-50-60-hz.html> >. Consulté le 14 janvier 2014.
- EKM Metering Inc. 2013b. « Split-core CT, 200A, 200A:26.6mA, 13mm (0.51") Hole Diameter, SCT-013-200 ». < <http://documents.ekmmetering.com/EKM-SCT-13-200-CT-spec-sheet.pdf> >. Consulté le 14 janvier 2014.
- EKM Metering Inc. 2013c. « Split-core CT, 400A, 400A:26.6mA, 23mm (0.91") Hole Diameter, SCT-023-400 ». < <http://documents.ekmmetering.com/EKM-SCT-23-400-CT-spec-sheet.pdf> >. Consulté le 12 juin 2013.
- Fluke. 2014. « Pince multimètre TRMS AC Fluke 365 ». In *Fluke*. < <http://www.fluke.com/fluke/cafr/pinces-multimetres/fluke-365.htm?PID=70409> >. Consulté le 13 janvier 2014.
- Haykin, Simon. 1984. *Introduction to adaptive filters*, 1st. United States: Macmillan Publishing Company, 217 p.
- Haykin, Simon S.; Widrow, Bernard. 2003. *Least-mean-square adaptive filters*. N.J.: Hoboken.

- Hugo Gävert, Jarmo Hurri, Jaakko Särelä, and Aapo Hyvärinen. 2005. « The FastICA package for MATLAB ». In *Independent Component Analysis (ICA) and Blind Source Separation (BSS)*. < <http://research.ics.aalto.fi/ica/fastica/> >. Consulté le 9 décembre 2013.
- Hyvärinen, A. 1999. « Fast and robust fixed-point algorithms for independent component analysis ». *Neural Networks, IEEE Transactions on*, vol. 10, n° 3, p. 626-634.
- Hyvärinen, A., et E. Oja. 2000. « Independent component analysis: algorithms and applications ». *Neural Netw.*, vol. 13, n° 4-5, p. 411-430.
- Hyvärinen, Aapo, Juha Karhunen et Erkki Oja. 2001. *Independent Component Analysis*, 1st. Coll. « Signal Processing, Communications, and Control ». Canada: Wiley-Interscience, 504 p.
- IEEE STANDARDS ASSOCIATION. 2003. « IEEE 802.15.4 ». < <http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf> >.
- Levasseur, M. A., S. Jomphe, D. Sicard, A. Dulipovici, F. Gagnon et G. Gagnon. 2012. « A wireless sensor network for residential electrical energy consumption measurement ». In *Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on*. (April 29 2012-May 2 2012), p. 1-4.
- Levasseur, Marc-André 2013. « Caractérisation et amélioration d'un système de mesures de la consommation électrique résidentielle ». École de technologie supérieure.
- Powerhouse Dynamics. 2014. « Residential eMonitor ». < <http://www.powerhousedynamics.com/residential-energy-efficiency> >. Consulté le 4 février 2014.
- Ramsden, Edward. 2006. *Hall-Effect Sensors Theory and Applications*, second. MA: Newnes.
- Rui, Tan, Xing Guoliang, Liu Xue, Jianguo Yao et Yuan Zhaohui. 2010. « Adaptive Calibration for Fusion-based Wireless Sensor Networks ». In *INFOCOM, 2010 Proceedings IEEE*. (14-19 March 2010), p. 1-9.
- S. Chauhan, R.J. Allemang, R. Martell, D.L. Brown;. 2007. « Application of Independent Component Analysis and Blind Source Separation Techniques to Operational Modal Analysis ». In *Proceedings of the 25th IMAC*. (Orlando (FL)).
- Sensors and Actuators. 2013. « Hall Effect Sensor ». < <http://sensors-actuators-info.blogspot.ca/2009/08/hall-effect-sensor.html> >.

- Takruri, M., S. Challa et R. Yunis. 2009. « Data fusion techniques for auto calibration in wireless sensor networks ». In *Information Fusion, 2009. FUSION '09. 12th International Conference on.* (6-9 July 2009), p. 132-139.
- Takruri, M., S. Rajasegarar, S. Challa, C. Leckie et M. Palaniswami. 2011. « Spatio-temporal modelling-based drift-aware wireless sensor networks ». *Wireless Sensor Systems, IET*, vol. 1, n° 2, p. 110-122.
- Tourancheau, B., Y. Mazzer, V. Gavan, F. Kuznik et G. Krauss. 2009. « Software Calibration of Wirelessly Networked Sensors ». In *Sensor Technologies and Applications, 2009. SENSORCOMM '09. Third International Conference on.* (18-23 June 2009), p. 314-319.
- Trivedi, Shubhendi. 2010. « Blind Source Separation in Magnetic Resonance Images ». < <http://onionesquereality.wordpress.com/2010/01/30/blind-source-separation-in-magnetic-resonance-images/> >. Consulté le 23 octobre 2013.
- Wang, Chao. 2009. « APPLICATION AWARE DATA GATHERING AND CALIBRATION IN MOBILE SENSOR NETWORKS ». Thesis in electrical engineering. Madison, University of Wisconsin, 200 p.