

Ministère de l'enseignement Supérieur et de la recherche Scientifique

وزارة التعليم العالي والبحث العلمي

Badji Mokhtar Annaba University
Université Badji Mokhtar - Annaba
Faculté des Sciences de l'Ingénierie
Département électromécanique



جامعة باجي مختار - عنابة
كلية علوم الهندسة
قسم الالكتروميكانيك

THÈSE

Présentée pour obtenir le diplôme de

DOCTORAT

Spécialité : Mécatronique

Filière : Electromécanique

Par :

HARIDA Issam

Sujet de la thèse :

Contribution au diagnostic des systèmes par les méthodes de traitement du signal avancées

Thèse soutenue le 27 janvier 2026 devant le jury composé de :

N°	Nom complet	Grade	Institution	Qualité
01	HADJADJ AOUL Elias	Prof.	Univ. Badji Mokhtar - Annaba	Président
02	BOURAS Hichem	MCA	Univ. Badji Mokhtar - Annaba	Rapporteur
03	RAMDANI Messaoud	Prof.	Univ. Badji Mokhtar - Annaba	Examineur
04	LAKEHAL Abdelaziz	Prof.	Univ. Mohamed Cherif Messaidia- Souk Ahras	Examineur
05	LACHOURI Abderrazak	Prof.	Univ. 20 Août 1955 - Skikda	Examineur

Remerciement

Avant tout, je remercie ALLAH de m'avoir accordé les capacités, les compétences, les ressources, ainsi que les personnes qui m'ont soutenu, et tout ce dont j'avais besoin pour mener à bien ce travail.

Je tiens à exprimer ma profonde gratitude à mon directeur de thèse, le Dr BOURAS Hichem, ainsi qu'à mon ancien directeur de thèse, le Dr BOURAS Abdelkarim, pour leur encadrement et leurs conseils avisés tout au long de ces années. Leur soutien indéfectible a été crucial pour l'aboutissement de cette thèse.

Je remercie également le Professeur SAAD Saleh pour ses précieux conseils, ainsi que tous les membres du Laboratoire des Systèmes Électromécaniques (LSELM) pour leur aide et leur collaboration.

Je souhaite exprimer ma plus profonde gratitude au président et aux membres du jury pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail en tant qu'examineurs.

Enfin, j'adresse ma gratitude à tous les enseignants qui ont contribué de quelque manière que ce soit à cette réussite. Que chacun d'eux trouve ici l'expression de ma profonde reconnaissance.

Dédicaces

À mes chers parents, que Allah les protège, qui n'ont cessé de prier pour ma réussite,

À mon épouse, pour son encouragement et son soutien indéfectible tout au long de la réalisation de ce travail,

À mes frères,

À mes amis,

Résumé

Le présent travail de thèse est structuré en deux parties complémentaires. La première est consacrée à l'étude approfondie des principaux défauts mécaniques affectant les entraînements électromécaniques des machines électriques, ainsi qu'au développement d'outils avancés de surveillance et de diagnostic destinés à améliorer leur fiabilité et leur disponibilité. La seconde partie repose sur une campagne expérimentale réalisée au laboratoire d'Électromécanique. Elle vise à valider l'efficacité de différentes approches de diagnostic prédictif appliquées aux défauts mécaniques. Cette validation s'appuie, d'une part, sur l'analyse vibratoire d'un système éolien à l'aide de techniques de traitement du signal et, d'autre part, sur l'analyse de la signature du courant moteur (MCSA) ainsi que des signaux vibratoires d'un moteur à induction au moyen de techniques d'intelligence artificielle. Les travaux menés permettent d'établir une comparaison détaillée entre les méthodes de diagnostic basées sur le traitement du signal, notamment la transformée de Fourier rapide (FFT) et la transformée en ondelettes discrètes (DWT), et celles fondées sur l'intelligence artificielle, en particulier l'apprentissage automatique (Machine Learning, ML), représenté par les k plus proches voisins (k -NN), les machines à vecteurs de support (SVM) et les arbres de décision (DT), ainsi que l'apprentissage profond (Deep Learning, DL), représenté par les réseaux de neurones profonds (DNN), les réseaux de neurones convolutifs (CNN) et les réseaux de neurones récurrents de type Long Short-Term Memory (LSTM). Les performances de ces différentes approches sont évaluées et comparées afin d'identifier leurs avantages et leurs limites, ainsi que de déterminer leur pertinence pour la détection et le diagnostic des défauts mécaniques dans les systèmes électromécaniques.

Mots-clés : Diagnostic des défauts, maintenance prédictive, système éolien, moteur asynchrone, analyse vibratoire, MCSA, traitement du signal, intelligence artificielle, apprentissage automatique, apprentissage profond, réseaux de neurones.

Abstract

This thesis is structured into two complementary parts. The first part is devoted to an in-depth study of the main mechanical faults affecting electromechanical drives in electrical machines, as well as to the development of advanced monitoring and diagnostic tools aimed at improving their reliability and availability. The second part is based on an experimental campaign conducted in the Electromechanics Laboratory. Its objective is to validate the effectiveness of various predictive diagnostic approaches applied to mechanical faults. This validation relies, on the one hand, on the vibration analysis of a wind energy system using signal processing techniques and, on the other hand, on Motor Current Signature Analysis (MCSA) as well as vibration signal analysis of an induction motor using artificial intelligence techniques. The conducted research establishes a detailed comparison between diagnostic methods based on signal processing, particularly the Fast Fourier Transform (FFT) and the Discrete Wavelet Transform (DWT), and those based on artificial intelligence, especially Machine Learning (ML), represented by the k -Nearest Neighbors (k -NN), Support Vector Machines (SVM), and Decision Trees (DT), as well as Deep Learning (DL), represented by Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) recurrent neural networks. The performance of these different approaches is evaluated and compared in order to identify their advantages and limitations, and to assess their suitability for the detection and diagnosis of mechanical faults in electromechanical systems.

Keywords: Fault diagnosis, predictive maintenance, wind energy system, induction motor, vibration analysis, MCSA, signal processing, artificial intelligence, machine learning, deep learning, neural networks.

ملخص

ينقسم عمل هذه الأطروحة إلى جزأين متكاملين. يُخصّص الجزء الأول لدراسة معمقة لأهم الأعطال الميكانيكية التي تؤثر على أنظمة القيادة الكهروميكانيكية للآلات الكهربائية، بالإضافة إلى تطوير أدوات متقدمة للمراقبة والتشخيص بهدف تحسين موثوقيتها وتوافرها. أما الجزء الثاني فيعتمد على حملة تجريبية أُنجِزت في مخبر الكهروميكانيك. ويهدف إلى التحقق من فعالية مختلف مناهج التشخيص التنبئي المطبقة على الأعطال الميكانيكية. ويستند هذا التحقق، من جهة، إلى التحليل الاهتزازي لنظام طاقة ريحية باستخدام تقنيات معالجة الإشارة، ومن جهة أخرى، إلى تحليل بصمة تيار المحرك (MCSA) وكذلك الإشارات الاهتزازية لحرك حثي باستخدام تقنيات الذكاء الاصطناعي. وقد مكّنت الأعمال المنجزة من إجراء مقارنة مفصلة بين طرق التشخيص المعتمدة على معالجة الإشارة، ولا سيما تحويل فورييه السريع (FFT) وتحويل المويجات المتقطع (DWT)، وبين الطرق المعتمدة على الذكاء الاصطناعي، وخاصة التعلم الآلي (Machine Learning, ML)، المتمثل في خوارزمية أقرب الجيران (k -NN)، وآلات المتجهات الداعمة (SVM)، وأشجار القرار (DT)، وكذلك التعلم العميق (Deep Learning, DL)، المتمثل في الشبكات العصبية العميقة (DNN)، والشبكات العصبية الالتفافية (CNN)، والشبكات العصبية التكرارية من نوع الذاكرة طويلة وقصيرة المدى (LSTM). وقد تم تقييم ومقارنة أداء هذه المقاربات المختلفة من أجل تحديد مزاياها وحدودها، وكذلك تحديد مدى ملاءمتها للكشف عن الأعطال الميكانيكية وتشخيصها في الأنظمة الكهروميكانيكية.

الكلمات المفتاحية تشخيص الأعطال، الصيانة التنبئية، نظام طاقة ريحية، محرك حثي، التحليل الاهتزازي، MCSA، معالجة الإشارة، الذكاء الاصطناعي، التعلم الآلي، التعلم العميق، الشبكات العصبية.

Contributions scientifiques

Publications Internationales

I. Harida, A. Bouras, & R. Boudiaf. (2022). [Vibration analysis for the detection of combined mechanical defects of a wind turbine](#) . *Energetika*, 68(1). DOI: <https://doi.org/10.6001/energetika.v68i1.4860>

R. Boudiaf, B. Abdelkarim, & H. Issam. (2024). [Bearing fault diagnosis in induction motor using continuous wavelet transform and convolutional neural networks](#). *International Journal of Power Electronics and Drive Systems (IJPEDS)*, 15(1), 591-602. DOI: <https://doi.org/10.11591/ijpeds.v15.i1.pp591-602>

I. Harida, A. Bouras, & H. Bouras. (2026). [Intelligent bearing fault diagnostic approach based on MCSA and deep learning](#). *Acta Mechanica et Automatica*, article accepté pour publication.

Communications

Harida, I., Abdelkarim, B., Boudiaf, R. A Sensorless Online Diagnostic Method For Electromechanical System Monitoring. *industrial engineering, maintenance and safety*, Oran, Algeria, 2022.

Harida, I., Abdelkarim, B. Advanced Method Of Detection Of Failures In Electric Motors Using The Vibration Signal. *National Conference on Mechanics and Maintenance*, Boumerdès, algerie, 2021.

Bouras, A., Boudiaf, R., Harida, I. Identifying Mechanical Faults of Wind Turbines with Current Signal Analysis. *eNergetics*, Belgrade, Serbia, 2021.

Boudiaf, R., Abdelkarim, B., Harida, I. Vibration Analysis to Assess Mechanical Problems in Induction Motors. *industrial engineering, maintenance and safety*, Oran, Algerie, 2022.

Table des matières

Remerciement	i
Dédicaces	ii
Résumé	iii
Abstract	iv
Résumé en arabe	vi
Contributions Scientifiques	vi
Table des matières	vii
Liste des figures	x
Liste des tableaux	xiii
Liste des abréviations	xvi
Introduction générale	1
1 Étude préliminaire des moteurs électriques et de leurs anomalies	5
1.1 Introduction	5
1.2 Moteurs asynchrones	5
1.2.1 Constitution de la machine asynchrone	5
1.2.2 Les différents types du moteur asynchrone	6
1.2.3 Principe de fonctionnement de la machine asynchrone	7
1.3 Moteurs à courant continu	10
1.3.1 Constitution de la machine à courant continu	10
1.3.2 Principe de fonctionnement de la machine à courant continu	11
1.3.3 Bilan de puissance	12
1.4 Générateurs AC	14
1.4.1 Constitution d'un générateur	14
1.4.2 Principe de fonctionnement d'un générateur AC	15
1.4.3 Bilan de puissance	16
1.5 Les organes de liaisons et de transfert de puissance	16

1.5.1	Les accouplements	17
1.5.2	Les éléments de transmission souple	18
1.5.3	Les Engrenages	19
1.5.4	Généralités sur les roulements à billes	19
1.6	Les Défauts mécaniques des principaux organes de liaison	20
1.6.1	Le Déséquilibre massique ou (Balourd)	20
1.6.2	Défaut d'alignement	21
1.6.3	Mauvais serrage mécanique	22
1.6.4	Défauts dans les engrenages	22
1.6.5	Défauts des roulements	24
1.7	Synthèse des Méthodes de Diagnostic	25
1.7.1	Méthodes de diagnostic avec modèle analytique	25
1.7.2	Méthodes de diagnostic sans modèle à priori	25
1.7.3	Méthode par Traitement de Signal (MTS)	26
1.7.4	Méthode par Intelligence Artificielle (MIA)	26
1.8	Conclusion	26
2	Techniques de diagnostic basées sur le traitement du signal	27
2.1	Introduction	27
2.2	Techniques d'analyse du signal pour le diagnostic des défauts mécaniques	27
2.2.1	Transformée de Fourier (FT)	27
2.2.2	Transformée de Fourier à court terme (STFT)	28
2.2.3	Spectrogramme de Gabor	29
2.2.4	Transformée en ondelettes continue (CWT)	29
2.2.5	Transformée en ondelettes discrète (DWT)	30
2.3	Signatures Vibratoires des Principaux Défauts	31
2.3.1	Déséquilibre Massique ou Balourd	31
2.3.2	Désalignement	32
2.4	Signatures des anomalies de roulement dans le courant statorique	33
2.5	Conclusion	37
3	Techniques de diagnostic basées sur le <i>Machine Learning</i>	38
3.1	Introduction	38
3.2	Les fondations de l'AI, ML et DL	38
3.2.1	Les méthodes d'apprentissage	38
3.2.2	Les applications de l'apprentissage Supervisé	39
3.2.3	Les notions de base de ML	40
3.3	Arbres de Décision	43
3.3.1	Principe	43
3.3.2	Critères de Division	44
3.3.3	Algorithmes de Construction	44
3.3.4	Élagage de l'Arbre	45
3.3.5	Exemple	46
3.4	k -NN (k -Nearest Neighbors)	49
3.4.1	Principe	49
3.4.2	Choix du Nombre de Voisins k	50
3.4.3	Mesure de la Distance	50
3.4.4	Apprentissage Paresseux (Lazy Learning)	52
3.4.5	Sensibilité au Bruit	52

3.4.6	Algorithme K plus proches voisins	52
3.4.7	Exemple :	52
3.5	Machine à Vecteurs de Support (SVM)	54
3.5.1	Le cas Linéairement séparable	54
3.5.2	Le cas non Linéairement séparable	56
3.5.3	Exemple	58
3.6	Conclusion	59
4	Techniques de diagnostic basées sur le <i>Deep Learning</i>	60
4.1	Introduction	60
4.2	Réseaux de neurones profonds (DNN)	60
4.2.1	Architectures du réseau de neurones	60
4.2.2	Les Fonctions d'activation	63
4.2.3	L'apprentissage du réseau de neurone	65
4.2.4	Algorithme d'apprentissage par rétropropagation	71
4.2.5	Exemple	71
4.3	Réseaux de Neurones Convolutifs (CNN)	73
4.3.1	Architecture générale d'un CNN	74
4.3.2	Exemple	75
4.4	Réseaux de neurones LSTM	77
4.4.1	Structure d'une cellule LSTM	77
4.5	Conclusion	78
5	Application et discussion	80
5.1	Introduction	80
5.2	Application de l'analyse vibratoire à la détection des défauts mécaniques combinés d'une éolienne	80
5.2.1	Méthodologie	80
5.2.2	Signatures fréquentes des défaillances dans un système éolien	81
5.2.3	Résultats et discussion	82
5.3	Application de CWT et CNN au diagnostic des défauts de roulements des moteurs à induction	84
5.3.1	Approche proposée	84
5.3.2	Description de la base de données MFPT	86
5.3.3	Résultats et discussion	87
5.4	Application de la MCSA et DNN au diagnostic des défauts de roulements	91
5.4.1	Méthodologie	91
5.4.2	Méthode proposée	96
5.4.3	Résultats expérimentaux	96
5.5	Conclusion	99
	Bibliographie	103

Liste des figures

1.1	Les Éléments Constitutifs du Moteur Asynchrone	6
1.2	Rotor bobiné	7
1.3	Rotor à cage	7
1.4	Représentation des champs magnétiques tournant	8
1.5	Bilan des puissances de la machine asynchrone	9
1.6	Les Éléments Constitutifs du Moteur à courant continu	10
1.7	Vue du Moteur à courant continu	11
1.8	Règle de la main droite	12
1.9	Bilan des puissances de la machine à courant continu	13
1.10	Schéma d'un générateur AC	15
1.11	Schéma d'un générateur DC	15
1.12	Fem induite dans un générateur AC	15
1.13	Bilan des puissances d'un générateur AC	17
1.14	Différents types d'accouplements	17
1.15	Les courroies de transmission	19
1.16	Les types d'engrenage	20
1.17	Les roulements à billes	20
1.18	Image Vibratoire théorique d'un balourd	21
1.19	(a) : À gauche balourd statique, (b) : À droite balourd dynamique	21
1.20	Image illustrant le désalignement radial et angulaire	22
1.21	Désalignement de paliers se traduisant par une flexion de l'arbre	22
1.22	Image de mauvais serrage mécanique	23
1.23	Différents types d'engrenage	23
1.24	Roulement à billes	24
1.25	Défaut d'écaillage réel	25
2.1	Concept de la Transformée de Fourier	27
2.2	Passage de la caractéristique temporelle à la STFT	28
2.3	Principe de la Transformée d'Ondelettes Continues	30
2.4	Principe de la Transformée d'Ondelettes Discrète	31
2.5	Signal vibratoire généré par un balourd	31
2.6	Spectre théorique d'un défaut de balourd	32
2.7	Le signal temporel et l'image vibratoire théorique de désalignement parallèle	33
2.8	Images vibratoires d'un défaut d'alignement angulaire	33
2.9	Désalignement de paliers se traduisant par une flexion de l'arbre	34

2.10	Dimension de roulement à billes	35
2.11	Fréquences des défauts d'un roulement (623) : 1- théorique. 2- empirique. 3- moyenne sur 2000 roulements	36
3.1	Les domaines de l'AI, ML et DL	39
3.2	Exemple	42
3.3	L'arbre de décision (à gauche) partitionne en 5 zones (à droite)	44
3.4	Arbre de décision pour le diagnostic de machines électriques	48
3.5	Calcul des distances d_i d'un échantillon à classer x_{test}	49
3.6	Le cas Linéairement séparable	55
3.7	Transformation des données dans un espace de redescription.	57
4.1	Architecture du perceptron	61
4.2	Réseau de neurone à une seule couche cachée	61
4.3	Réseau de neurone à deux couches cachées	62
4.4	Architecture d'exemple	71
4.5	Architecture générale d'un réseau de neurones convolutif (CNN)	74
4.6	Structure interne d'une cellule LSTM	77
5.1	Configuration expérimentale pour la détection de défauts combinés	81
5.2	Signaux dans le domaine fréquentiel de différentes conditions du générateur	82
5.3	Décomposition du signal de vibration du générateur sain à une vitesse de 2400 tr/min	83
5.4	Décomposition du signal de vibration de déséquilibre du rotor et d'un défaut de desserrage à une vitesse de 2400 tr/min	83
5.5	Organigramme de la méthodologie proposée pour la classification des défauts. . .	84
5.6	Scalogrammes obtenus à l'aide de l'ondelette Morse : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.	87
5.7	Scalogrammes obtenus à l'aide de l'ondelette Morlet : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.	88
5.8	Scalogrammes obtenus à l'aide de l'ondelette Bump : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.	88
5.9	Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Morse. .	89
5.10	Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Morlet. .	89
5.11	Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Bump. .	89
5.12	Matrices de confusion du modèle CNN pour les différents types d'ondelettes. (a) Ondelette Morse ; (b) ondelette Morlet ; (c) ondelette Bump.	90
5.13	Schéma du banc d'essai expérimental	91
5.14	Sans charge	92
5.15	Charge de 100 W	92
5.16	Charge de 200 W	92
5.17	Charge de 300 W	92
5.18	Distribution des caractéristiques : (a) moyenne, (b) écart-type (Std) et (c) valeur efficace (RMS) pour les conditions saines et défectueuses.	93
5.19	Système de diagnostic des défauts de roulements basé sur un réseau DNN.	94
5.20	Matrice de confusion de la méthode DNN.	95

5.21	Matrice de confusion de la méthode KNN.	95
5.22	Matrice de confusion de la méthode SVM.	95
5.23	Matrice de confusion de la méthode DT.	95
5.24	Matrice de confusion de la méthode CNN.	95
5.25	Matrice de confusion de la méthode LSTM.	95
5.26	Architecture de la méthode proposée	97
5.27	Organigramme de la méthode proposée pour le diagnostic en ligne d'un moteur électrique	97
5.28	Matrice de confusion de la méthode DNN proposé.	98
5.29	Matrice de confusion de la méthode KNN proposé.	98
5.30	Matrice de confusion de la méthode SVM proposé.	98
5.31	Matrice de confusion de la méthode DT proposé.	98
5.32	Matrice de confusion de la méthode CNN proposé.	98
5.33	Matrice de confusion de la méthode LSTM proposé.	98
5.34	Comparaison des méthodes conventionnelles et les méthodes proposé	99

Liste des tableaux

1.1	Comparaison des caractéristiques des accouplements à dents et rigides	18
3.1	Données pour le diagnostic de machines électriques	47
4.1	Résultats de classification du CNN	76
5.1	Détails des machines	81
5.2	Caractéristiques des signaux, longueurs d’acquisition et nombre de segments des ensembles d’apprentissage et de test pour les différentes conditions de défaut. . .	87
5.3	Structure de la base de données	92
5.4	Architecture du réseau de neurones profond (DNN)	93
5.5	Comparaison détaillée des différentes méthodes	94
5.6	Évaluation comparative entre les différentes méthodes	99

Liste des abréviations

Acronyme	Signification
AC	Alternating Current
AI	Artificial Intelligence
CNN	Convolutional Neural Networks
CTW	Continuous Wavelet Transform
DC	Direct Current
DL	Deep Learning
DNN	Deep neural network
DT	Decision Tree
DTW	Discrete Wavelet Transform
FFT	Fast Fourier Transform
FT	Fourier Transform
GT	Gabor Transform
k NN	k -Nearest Neighbors
LSTM	Long Short-Term Memory
MAS	Machine Asynchrone
MCSA	Analyse de la signature du courant
MIA	Méthode par Intelligence Artificielle
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Squared Error
MTS	Méthode par Traitement de Signal
NAG	Nesterov Accelerated Gradient
PCA	Principal Component Analysis
RBF	Radial Basis Function
RNN	Recurrent Neural Networks
RTF	Résolution temps-fréquence
SARSA	State-action-reward-state-action
SGD	Stochastic Gradient Descent
STFT	Short-Time Fourier Transform

SVM	Support Vector Machine
SVR	Support Vector Regression
TFCT	Transformée de Fourier à Court Terme
TO	Transformée en ondelettes

Notation	Description	Unité
a	Vecteur d'activation	-
b	Biais du réseau	-
C	Nombre de classes	-
d_i	Distance euclidienne	-
D	Ensemble de données d'apprentissage	-
E	Force électromotrice	V
$E[G^2]_t$	Moyenne carrés des gradients	-
f	Fréquence	Hz
$f(x)$	Valeur prédite	-
g	Glissement	-
$g(a)$	Fonction d'activation	-
I	Courant	A
J	Fonction de coût	-
k	Nombre de voisins dans k -NN	-
$k(x, x_i)$	Fonction noyau	-
L	Fonction de Lagrange	-
n	Fréquence de rotation	tr/s
N	Nombre de spires	-
N	Nombre ensemble de données	-
P	Puissance	W
P_a	Puissance absorbée	W
P_e	Puissance absorbée par l'inducteur	W
P_f	Pertes fer induit	W
P_i	Puissance absorbée par l'induit	W
P_m	Pertes mécaniques	W
P_{em}	Puissance électromagnétique	W
P_{ji}	Pertes par effet joule dans l'induit	W
P_{je}	Pertes par effet Joule dans l'inducteur	W
P_{mec}	Puissance mécanique	W
P_u	Puissance mécanique utile	W
R	Résistance	Ω
R_e	Résistance de l'inducteur	Ω
R_r	Résistance rotorique	Ω
R_s	Résistance statorique	Ω
t	Temps	s

U	Tension	V
U_e	Tension d'alimentation de l'inducteur	V
U_i	Tension aux bornes de l'induit	V
U_s	Tension statorique	V
v	Vitesse linéaire	m/s
w	Vecteur de poids	-
x	Vecteur d'entrée	-
y	Étiquette	-
α	Taux d'apprentissage	-
α_i	Multiplicateurs de Lagrange	-
ϵ	Petite constante	-
μ	Moyenne	-
μ	Matrice de covariance	-
ω	Vitesse angulaire	rad/s
σ	Écart type	-

Introduction générale

DANS les entraînements électromécaniques, en particulier ceux de grandes puissances, les moteurs électriques et les organes qui leur sont associés, tels que les accouplements et les réducteurs, jouent un rôle crucial dans la conversion de l'énergie électrique et le transfert de la puissance mécanique, adaptant ainsi les vitesses aux diverses charges des procédés technologiques. Comme le soulignent Vas [70], les systèmes d'entraînement modernes constituent l'élément central de la performance énergétique et de la productivité industrielle. Le diagnostic prédictif de défauts s'inscrit aujourd'hui dans une stratégie globale de maintenance intelligente visant à réduire les coûts d'exploitation et à améliorer la disponibilité des installations [47]. Parmi les différentes machines électriques utilisées dans les entraînements électromécaniques, les moteurs à induction se distinguent par leur simplicité de construction, leur robustesse et leur adaptation aux vitesses variables. Selon Fitzgerald et al. [25], ils représentent la majorité des machines installées dans l'industrie lourde en raison de leur fiabilité et de leur faible coût de maintenance. Cependant, ils demeurent susceptibles de subir diverses anomalies mécaniques. D'après Thomson & Gilmore [66], les défaillances mécaniques constituent aujourd'hui une part significative des pannes recensées dans les moteurs à induction de forte puissance. Face aux contraintes mécaniques importantes auxquelles ils sont soumis, les entraînements électriques utilisant des moteurs à induction de grandes puissances peuvent être touchés simultanément par plusieurs anomalies. Comme l'indique Randall [56], la coexistence de défauts combinés complique considérablement l'interprétation des signatures vibratoires et électriques. Dans ce contexte, la mécatronique joue un rôle de plus en plus crucial. Comme le précise Bolton [15], l'approche mécatronique favorise l'intégration de capteurs intelligents et d'algorithmes avancés de traitement du signal dans les stratégies de surveillance conditionnelle. L'intégration croissante des énergies renouvelables impose une exigence accrue de fiabilité des systèmes électromécaniques intégrés aux réseaux intelligents. Selon Nandi et al. [49], les défauts électriques sont devenus moins fréquents grâce aux progrès des procédés de fabrication, tandis que les défaillances mécaniques restent prédominantes. La modélisation des défauts multiples demeure un défi scientifique en raison des phénomènes de modulation et d'interaction non linéaire entre signatures fréquentielles [5]. Cette approche s'inscrit dans le cadre de l'industrie 4.0, où l'exploitation des données massives et des méthodes d'apprentissage automatique devient essentielle [46].

Etat de l'art

Le diagnostic des défauts des machines électriques constitue un domaine de recherche particulièrement actif en raison de l'importance de ces équipements dans les secteurs industriels, énergétiques et des transports. Les défaillances des composants tels que les roulements, le stator ou le rotor peuvent entraîner des arrêts de production coûteux, réduire la disponibilité des installations et affecter leur sécurité de fonctionnement [32]. Par conséquent, le développement de systèmes de surveillance et de diagnostic performants est devenu un enjeu majeur dans le cadre de la maintenance prédictive et de l'industrie 4.0 [31]. Les premières méthodes de diagnostic reposaient principalement sur les techniques de traitement du signal appliquées aux signaux de courant, de vibration, de flux magnétique ou d'émission acoustique [50]. Parmi les approches les plus utilisées figurent l'analyse temporelle, l'analyse fréquentielle basée sur la transformée de Fourier rapide (FFT), l'analyse de la signature du courant moteur (MCSA) ainsi que les méthodes temps-fréquence telles que la transformée en ondelettes continue (CWT) et la transformée de Fourier à court terme (STFT) [11, 57]. Ces techniques permettent d'extraire des indicateurs représentatifs de l'état de santé de la machine et demeurent largement utilisées dans les applications industrielles actuelles. Avec l'évolution des capacités de calcul et l'essor de l'intelligence artificielle, de nouvelles approches basées sur l'apprentissage automatique ont été développées afin d'améliorer les performances du diagnostic [72]. Les algorithmes tels que les machines à vecteurs de support (SVM), les k plus proches voisins (k -NN), les arbres de décision et les réseaux de neurones artificiels ont montré leur efficacité pour la classification des différents défauts des machines électriques [71]. Toutefois, ces méthodes nécessitent généralement une étape préalable d'extraction manuelle des caractéristiques, ce qui limite parfois leur capacité de généralisation. Plus récemment, l'apprentissage profond a suscité un intérêt considérable dans le domaine du diagnostic intelligent des machines électriques. Les réseaux de neurones convolutifs (CNN), les réseaux à mémoire longue à court terme (LSTM) ainsi que les architectures Transformer permettent d'extraire automatiquement les caractéristiques pertinentes à partir des données brutes ou des représentations temps-fréquence des signaux [30, 72]. Plusieurs études récentes ont démontré que ces architectures offrent des performances supérieures aux méthodes conventionnelles pour la détection et l'identification des défauts de roulements, de stator et de rotor [39]. La tendance actuelle consiste à combiner les techniques avancées de traitement du signal avec les modèles d'apprentissage profond afin de tirer profit à la fois de la richesse des représentations temps-fréquence et de la puissance de généralisation des réseaux neuronaux [39, 32]. Ces approches hybrides constituent aujourd'hui l'une des orientations les plus prometteuses pour le développement de systèmes de maintenance prédictive intelligents adaptés aux exigences de l'industrie moderne. Malgré les progrès significatifs réalisés au cours des dernières années, plusieurs défis demeurent, notamment la robustesse des modèles face aux variations des conditions de fonctionnement, la disponibilité limitée des données de défauts réels et la généralisation des systèmes de diagnostic à différentes configurations de machines électriques [72, 31]. Ces limitations motivent la poursuite des recherches dans ce domaine et justifient les travaux présentés dans cette thèse.

Organisation de la thèse

Ce mémoire sera divisé en **cinq chapitres** pour explorer en profondeur les différents aspects du diagnostic des défauts dans les systèmes électromécaniques.

Dans le premier chapitre, nous présenterons, dans sa première partie, les principaux éléments constitutifs du moteur asynchrone, du moteur à courant continu et de la génératrice AC. Nous examinerons en détail leur fonctionnement et leur construction. Ensuite, dans la deuxième partie, nous exposerons brièvement les éléments traditionnels liés à l'arbre rotorique, tels que les accouplements et les roulements, nécessaires pour assurer la transmission du mouvement et de la puissance mécanique à la charge du procédé. Nous discuterons des principaux défauts mécaniques qui peuvent affecter ces éléments et nous fournirons des informations sur leurs effets potentiels sur le système dans son ensemble. La troisième partie se concentrera sur les principales défaillances mécaniques des organes de liaison, telles que les désalignements, les déséquilibres et les défauts de roulement, souvent à l'origine d'arrêts intempestifs du procédé. Nous explorerons les méthodes de diagnostic actuelles et les avancées récentes dans ce domaine, en mettant en lumière les techniques les plus efficaces pour détecter et identifier ces défauts.

Le deuxième chapitre présente un état de l'art des techniques de diagnostic des défauts des machines électriques. Il s'intéresse particulièrement aux méthodes de traitement du signal les plus couramment utilisées pour l'analyse des signaux de surveillance, telles que la transformée de Fourier (TF), la transformée de Fourier à court terme (STFT), le spectrogramme de Gabor (GT), la transformée en ondelettes continue (CWT) et la transformée en ondelettes discrètes (DWT). Ce chapitre décrit également les signatures fréquentielles et temporelles associées aux principaux défauts mécaniques, notamment le balourd, le désalignement et les défauts de roulement. L'objectif est de mettre en évidence les avantages et les limites de ces approches et de fournir le cadre théorique nécessaire aux travaux de diagnostic proposés dans cette thèse.

Le troisième chapitre est consacré à l'étude des méthodes de diagnostic basées sur l'apprentissage automatique (*Machine Learning*). Dans ce chapitre, nous présenterons tout d'abord les fondements de l'intelligence artificielle ainsi que les principaux paradigmes d'apprentissage utilisés dans le développement des systèmes de diagnostic intelligents. Nous nous intéresserons ensuite aux algorithmes de classification les plus couramment employés pour le diagnostic des défauts des machines électriques, notamment les k plus proches voisins (k -NN), les machines à vecteurs de support (SVM) et les arbres de décision (DT). Enfin, nous aborderons les différentes étapes de conception d'un système de diagnostic basé sur l'apprentissage automatique, depuis la préparation des données et l'extraction des caractéristiques jusqu'à l'entraînement, la validation et l'évaluation des performances des modèles.

Le quatrième chapitre est consacré à l'étude des méthodes de diagnostic basées sur l'apprentissage profond (*Deep Learning*). Après une présentation des concepts fondamentaux de l'apprentissage profond, nous détaillerons les architectures les plus utilisées dans le domaine du diagnostic des défauts des machines électriques. Une attention particulière sera accordée aux réseaux de neurones profonds (*Deep Neural Networks* - DNN), aux réseaux de neurones convolutifs (*Convolutional Neural Networks* - CNN) ainsi qu'aux réseaux à mémoire longue à court terme (*Long Short-Term Memory* - LSTM), dont les principes de fonctionnement, les architectures.

Le cinquième et dernier chapitre est consacré à la validation expérimentale et à l'application des différentes méthodes de diagnostic étudiées dans cette thèse sur des systèmes réels. Dans un premier temps, une application de l'analyse vibratoire à la détection des défauts mécaniques combinés d'une éolienne est présentée afin d'évaluer la capacité des techniques conventionnelles de traitement du signal à identifier simultanément plusieurs anomalies mécaniques. Dans un second temps, une approche basée sur la transformée en ondelettes continue (CWT) couplée à un

réseau de neurones convolutif (CNN) est développée pour le diagnostic des défauts de roulements des moteurs à induction. Cette approche exploite les représentations temps-fréquence générées par la CWT afin d'améliorer les performances de classification. Enfin, une troisième étude est consacrée à l'application de l'analyse de la signature du courant moteur (MCSA) associée à un réseau de neurones profond (DNN) pour le diagnostic des défauts de roulements, mettant en évidence, à travers une analyse comparative de différentes méthodes d'intelligence artificielle, la pertinence des réseaux de neurones profonds (DNN), qui se sont révélés les plus performants pour le diagnostic des défauts de roulements à partir des signaux de courant.

Les travaux présentés dans ce chapitre ont fait l'objet de plusieurs publications scientifiques, témoignant de leur pertinence et de leur contribution au domaine du diagnostic des machines électriques. Une analyse comparative des résultats obtenus est également réalisée afin d'évaluer les performances des différentes approches proposées. Cette analyse permet de mettre en évidence les avantages et les limites des méthodes conventionnelles basées sur le traitement du signal ainsi que des approches fondées sur l'intelligence artificielle. Enfin, une synthèse générale est proposée afin d'identifier les solutions les plus adaptées aux exigences des systèmes modernes de surveillance et de maintenance prédictive.

Étude préliminaire des moteurs électriques et de leurs anomalies

1.1 Introduction

Ce chapitre vise à présenter de manière concise la structure des moteurs asynchrones et des moteurs à courant continu, ainsi que celle des générateurs. Il offre un aperçu des composants qui leur sont associés, indispensables au transfert de la puissance mécanique, tout en répondant aux exigences de vitesse et d'efficacité propres aux applications industrielles. En outre, une analyse synthétique des divers défauts pouvant affecter ces composants sera abordée. Cette analyse mettra en lumière les techniques de détection couramment employées dans ce domaine.

1.2 Moteurs asynchrones

Le moteur asynchrone, couramment désigné comme moteur d'induction, représente aujourd'hui le type de moteur électrique le plus répandu dans le secteur industriel. Parmi ses nombreux avantages, on note une compacité appréciable, un rendement énergétique élevé, une fiabilité remarquable et une simplicité d'installation. L'un de ses principaux atouts réside dans l'absence de contacts électriques glissants, ce qui lui confère une structure simple et robuste, ainsi qu'une fabrication aisée.

Les moteurs asynchrones couvrent une large gamme de puissances, allant de quelques watts à plusieurs mégawatts. Lorsqu'ils sont alimentés directement par un réseau industriel à tension et fréquence fixes, ils maintiennent une vitesse de rotation qui est proche de la vitesse synchrone. Ce type de moteur est principalement utilisé pour les applications nécessitant une vitesse constante.

1.2.1 Constitution de la machine asynchrone

[2, 1] Les machines asynchrones peuvent être subdivisées, du point de vue mécanique, en trois composantes distinctes :

- Le stator, constituant fixe de la machine, où l'alimentation électrique est connectée.
- Le rotor, élément rotatif essentiel qui génère le mouvement nécessaire à la rotation de la charge mécanique.
- Les paliers, composants mécaniques permettant le bon fonctionnement de l'arbre moteur en facilitant sa rotation.

La figure (fig. 1.1) représente les principaux éléments constitutifs du moteur asynchrone.

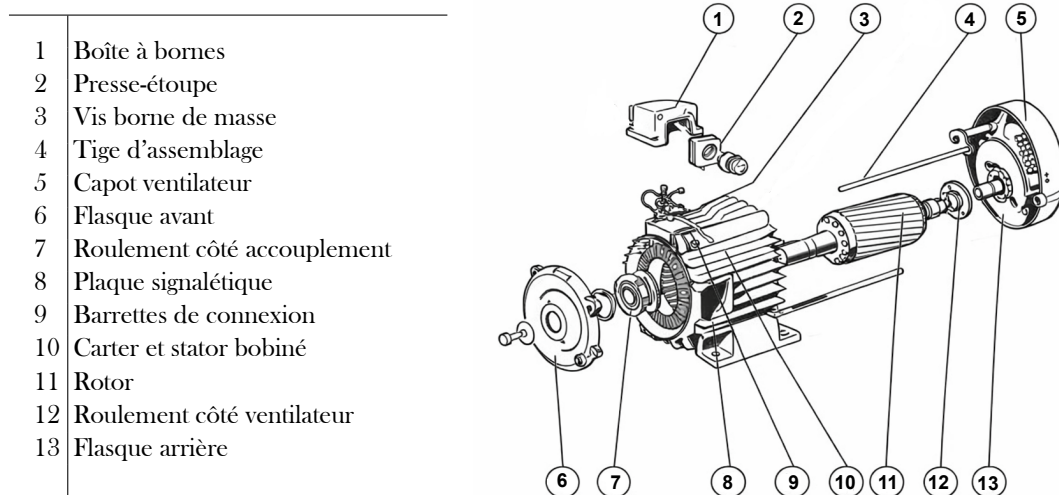


Figure 1.1: Les Éléments Constitutifs du Moteur Asynchrone

1.2.1.1 Le stator

Le stator constitue la partie immobile du moteur. Il est composé d'une carcasse à laquelle est attachée une couronne de tôles en acier de qualité spécifique, dotées de rainures. Des bobinages de section adaptée sont insérés dans ces rainures, formant un ensemble d'enroulements qui comprend autant de circuits qu'il y a de phases dans le réseau d'alimentation.

1.2.1.2 Le rotor

Le rotor représente la partie mobile du moteur et tourne à une vitesse inférieure à celle du champ tournant. Il est positionné à l'intérieur du stator et est constitué d'un empilage de tôles d'acier formant un cylindre, court-circuité à chaque extrémité par une couronne en cuivre ou en aluminium. L'ensemble forme une cage d'écureuil, et il est nécessaire que le rotor soit feuilleté afin de réduire les pertes par courant de Foucault.

1.2.2 Les différents types du moteur asynchrone

[2, 1] Les différents types de moteurs asynchrones se distinguent uniquement par leur rotor, dans tous les cas, le stator demeure identique, du moins dans son principe.

1.2.2.1 Rotor bobiné

Le rotor est formé d'un enroulement bobiné à l'intérieur d'un circuit magnétique, composé de disques en tôle empilés sur l'arbre de la machine. Les rainures découpées dans ces tôles sont légèrement inclinées par rapport à l'axe de la machine pour réduire les variations de réluctance liées à la position angulaire entre le rotor et le stator, ainsi que certaines pertes dues aux harmoniques.

Les extrémités des enroulements du rotor sont connectées à des bagues montées sur l'arbre, sur lesquelles frottent des balais en carbone. Ce type de moteur est surtout utilisé dans des applications où les démarrages sont difficiles et/ou fréquents. La figure (fig. 1.2) représente le type de rotor à cage.

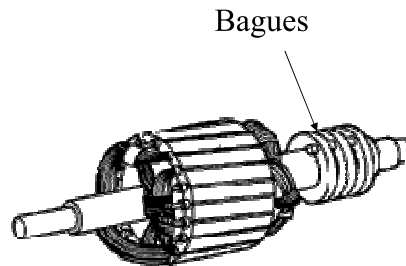


Figure 1.2: Rotor bobiné

1.2.2.2 le Rotor à cage

Le circuit du rotor est composé de barres conductrices réparties uniformément entre deux couronnes métalliques formant les extrémités, rappelant ainsi la forme d'une cage d'écureuil. Naturellement, cette cage est insérée dans un circuit magnétique similaire à celui d'un moteur à rotor bobiné. Les barres sont fabriquées en cuivre, en bronze ou en aluminium, en fonction des caractéristiques mécaniques et électriques souhaitées par le constructeur. Dans certaines conceptions, notamment pour des moteurs à basse tension (par exemple 230/400 V), la cage est réalisée par coulée et centrifugation d'aluminium.

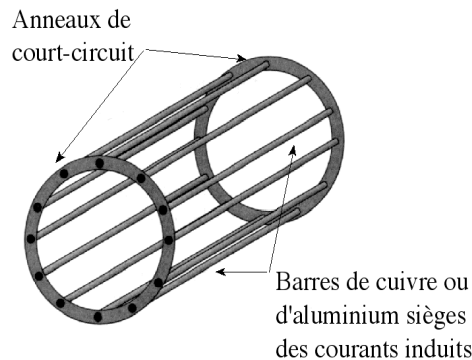


Figure 1.3: Rotor à cage

1.2.3 Principe de fonctionnement de la machine asynchrone

[2, 1] Les trois enroulements du stator, alimentés par un réseau triphasé équilibré, créent dans l'entrefer un champ magnétique tournant à la fréquence de rotation synchronique n_s . Les conducteurs du rotor sont soumis à ce champ tournant, ce qui induit des courants de Foucault. Selon la loi de Lenz ("les courants induits s'opposent par leurs effets à la cause qui les a engendrés"), et étant donné que les enroulements du rotor sont en court-circuit, les courants peuvent circuler. Les forces de Laplace résultantes exercent des moments sur le rotor, provoquant sa rotation à la fréquence n . En principe, la fréquence de rotation du rotor est inférieure à la fréquence de synchronisme n_s : $n < n_s$. Les deux fréquences ne peuvent pas être synchrones, d'où le nom de moteur asynchrone. La représentation des champs magnétiques tournants est montrée dans la figure (fig. 1.4).

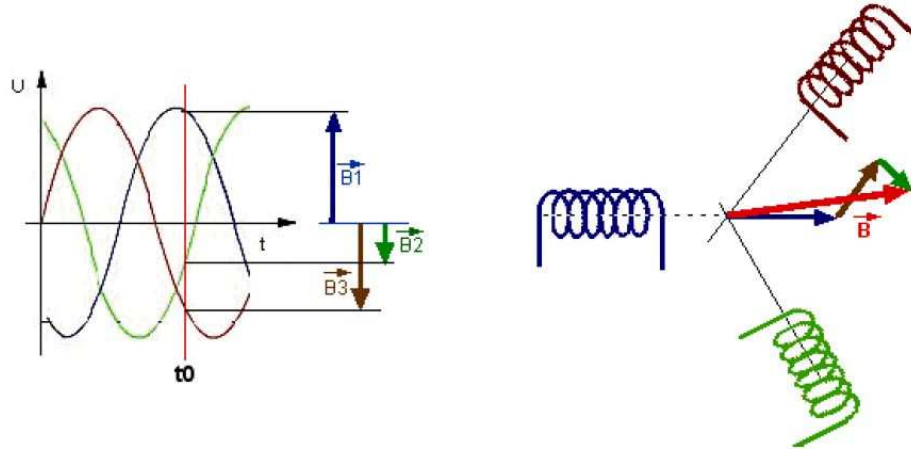


Figure 1.4: Représentation des champs magnétiques tournant

1.2.3.1 Glissement

Le rotor tourne à la fréquence de rotation n , qui est inférieure à celle du champ tournant, lequel tourne à la fréquence de rotation n_s . La fréquence de glissement est définie comme la différence entre ces deux fréquences de rotation :

$$n_g = n_s - n \quad (1.1)$$

Le glissement d'un moteur asynchrone est défini comme le rapport entre la différence de fréquence entre la fréquence de glissement et la fréquence de synchronisme :

$$g = \frac{n_s - n}{n_s} \quad (1.2)$$

1.2.3.2 Fréquence de synchronisme d'une MAS

La fréquence associée à la rotation du champ magnétique tournant (fréquence synchronique) est exprimée par la relation :

$$n_s = \frac{f}{p} \quad (1.3)$$

p : Le nombre de paires de pôles créés par le stator.

f : La fréquence des tensions d'alimentation en Hz.

1.2.3.3 Bilan de puissance

[2, 1] La figure (fig. 1.1) résume le bilan des puissances de la machine asynchrone.

La puissance électrique absorbée par le stator :

$$P_a = \sqrt{3} UI \cos \varphi \quad (1.4)$$

P_a : La puissance électrique absorbée en watts (W).

U : La tension entre deux phases en volts (V).

I : L'intensité du courant de ligne en ampères (A).

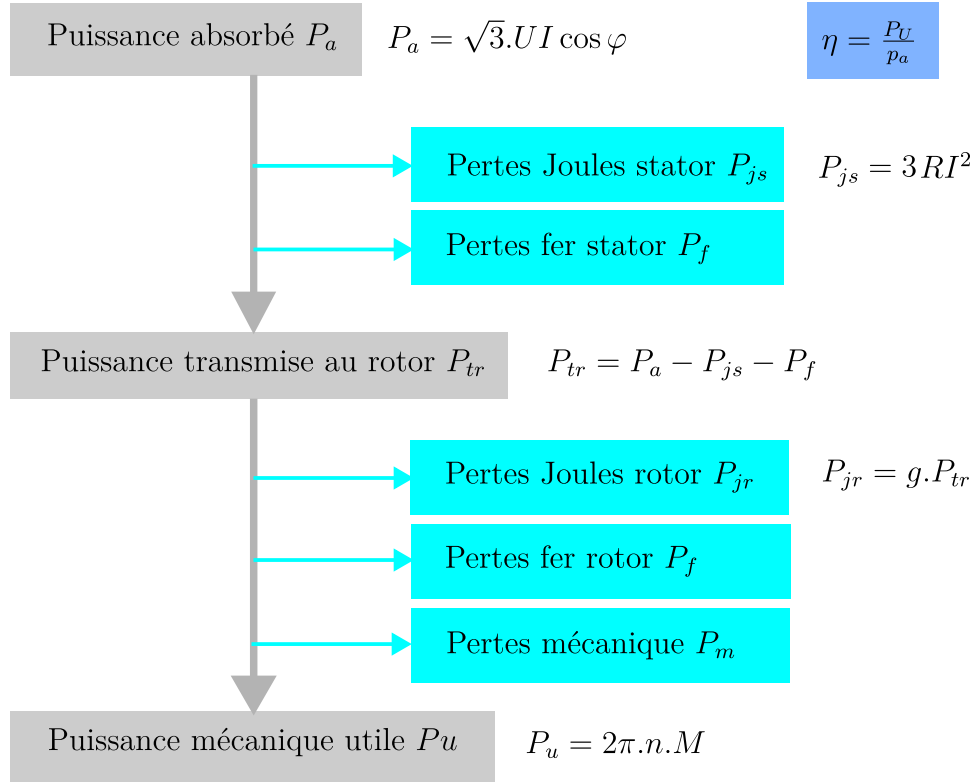


Figure 1.5: Bilan des puissances de la machine asynchrone

Les pertes statoriques :

- Pertes joules triphasés :

$$P_{js} = 3.RI^2 \quad (1.5)$$

R : résistance mesurée entre deux bornes du stator.

- Pertes magnétiques dues au phénomène d'hystérésis : P_f (pertes fer)

La puissance transmise du stator au rotor :

$$P_{tr} = P_a - P_{js} - P_f \quad (1.6)$$

Les pertes rotoriques :

- Pertes par effet joules : Dues aux courants de Foucault (correspond à la perte de vitesse : glissement).

$$P_{jr} = g.P_{tr} \quad (1.7)$$

- Pertes mécaniques : Dues aux frottements : P_m

La puissance mécanique utile :

Soit M le moment du couple moteur, on a :

$$P_{tr} = P_a - P_{js} - P_f$$

$$P_u = 2\pi.n.M \quad (1.8)$$

n en tr/s où f en Hz

Les pertes constantes :

$$P_{tr} = P_a - P_{js} - P_f \quad (1.9)$$

Elles ne dépendent pas du régime de fonctionnement du moteur et peuvent donc être mesurée à vide.

Le rendement

$$\eta = \frac{P_U}{P_a} \quad (1.10)$$

1.3 Moteurs à courant continu

Le moteur à courant continu est l'un des moteurs électriques les plus couramment utilisés dans de nombreuses applications industrielles, commerciales et résidentielles. Sa popularité découle de sa simplicité de construction, de son contrôle précis de la vitesse et de son couple élevé à basse vitesse.

1.3.1 Constitution de la machine à courant continu

[52] La machine à courant continu comprend les composants principaux suivants :

- Une section statique nommée **Stator**, chargée de l'induction.
- Une partie mobile désignée comme **Rotor**, assurant le rôle d'induit.
- Une connexion entre le rotor et les composants externes de la machine connue sous le nom de **Collecteur**.

La figure (fig. 1.6) représente les éléments constitutifs du moteur à courant continu.

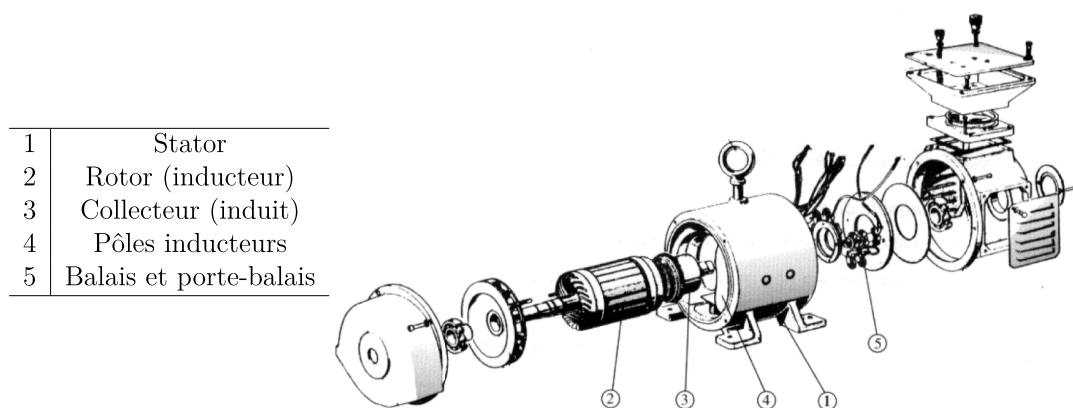


Figure 1.6: Les Éléments Constitutifs du Moteur à courant continu

1.3.1.1 Inducteur

L'inducteur peut être constitué soit d'aimants permanents en ferrite, soit de bobines disposées autour des noyaux polaires. Lorsque les bobines sont alimentées par un courant continu, elles génèrent un champ magnétique à travers le circuit magnétique de la machine, spécialement dans l'entrefer qui sépare la partie fixe de la partie mobile, où se trouvent les conducteurs. Ce flux magnétique et ce champ sont dirigés du pôle Nord vers le pôle Sud.

1.3.1.2 Induit

L'induit est constitué d'un noyau en fer qui guide les lignes de champ magnétique, et les conducteurs sont placés dans des encoches sur le rotor, où deux conducteurs forment une boucle.

1.3.1.3 Collecteur et balais

Le collecteur consiste en un groupe de lames de cuivre isolées, placées à l'extrémité du rotor. Les balais, soutenus par le stator, font contact avec le collecteur.

La figure (fig. 1.7) donne une vue générale du moteur à courant continu.

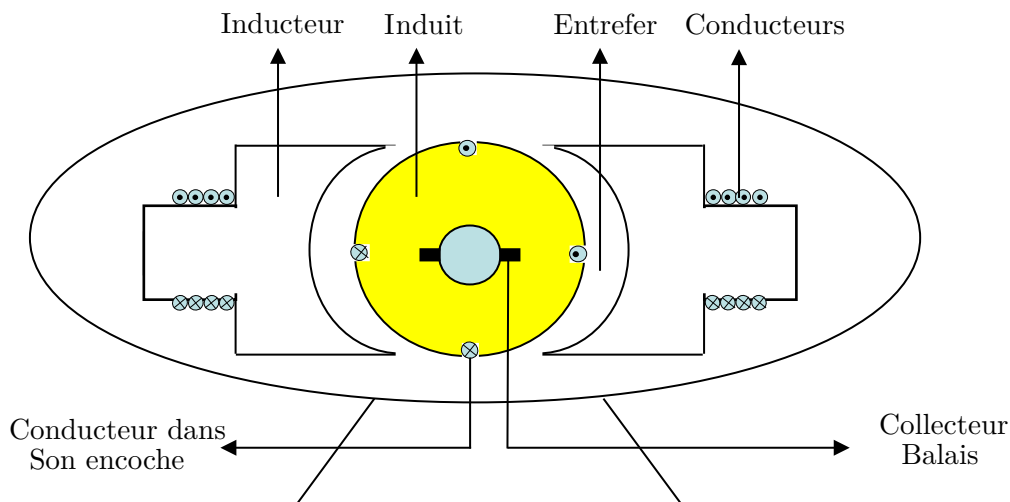


Figure 1.7: Vue du Moteur à courant continu

1.3.2 Principe de fonctionnement de la machine à courant continu

[52] Une machine à courant continu comporte un nombre n de conducteurs actifs au niveau de l'induit. Le flux magnétique utile sous un pôle créé par l'inducteur est Φ , exprimé en webers, et N représente la vitesse de rotation de l'arbre du rotor, en tours par seconde.

Deux situations peuvent se produire :

- Lorsqu'un conducteur est traversé à la fois par un courant électrique et est situé dans un champ magnétique, il est alors sujet à une force électromagnétique.
- Lorsqu'un conducteur est en rotation tout en étant immergé dans un champ magnétique, il engendre une force électromotrice.

La loi de Laplace stipule que l'interaction d'un champ magnétique B avec un courant I dans un conducteur de longueur L génère une force F . Cette force est déterminée par la règle de la main droite (fig. 1.8) et peut être calculée avec la formule : $F = I \cdot L \times B$.

L'ensemble des forces appliquées se combine pour produire un moment, induisant la rotation du moteur. Les conducteurs en action, au nombre de n , traversent les lignes du champ magnétique.

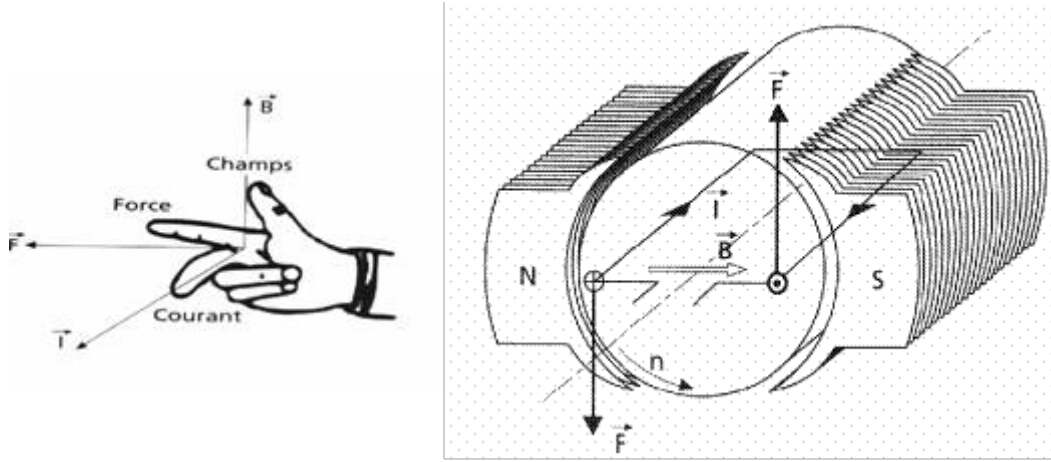


Figure 1.8: Règle de la main droite

Selon la loi de Lenz-Faraday, chaque conducteur génère donc des forces électromotrices induites $e = \frac{\Delta\Phi}{\Delta t} = n \cdot N \cdot \Phi$, où la force électromotrice (f.e.m) totale pour l'ensemble des N spires :

$$E = n \cdot N \cdot \Phi \quad (1.11)$$

- E : La force électromotrice en volts (V).
- N : La fréquence de rotation en tours par seconde (tr.s^{-1}).
- Φ : Le flux en webers (Wb).
- n : Le nombre de conducteurs actifs.

Cette équation est cruciale pour le fonctionnement de la machine, car elle établit le lien entre le flux magnétique Φ , la tension électrique E , et la fréquence de rotation N , représentant respectivement des grandeurs magnétique, électrique et mécanique.

En considérant que $\Omega = 2\pi \cdot N$, une autre équation couramment utilisée lie les trois types de grandeurs. Elle prend en compte la vitesse angulaire Ω exprimée en radians par seconde :

$$E = K \cdot \Phi \cdot \Omega \quad (1.12)$$

- E : La force électromotrice en volts (V).
- Ω : La vitesse angulaire en radians par seconde (rad.s^{-1}).
- Φ : Le flux en webers (Wb).
- K : Constante.

1.3.3 Bilan de puissance

[52] La figure (fig. 1.1) résume le bilan des puissances de la machine à courant continu.

La puissance absorbée:

$$P_a = P_i + P_e \quad (1.13)$$

P_i Puissance absorbée par l'induit.

P_e Puissance absorbée par l'inducteur.

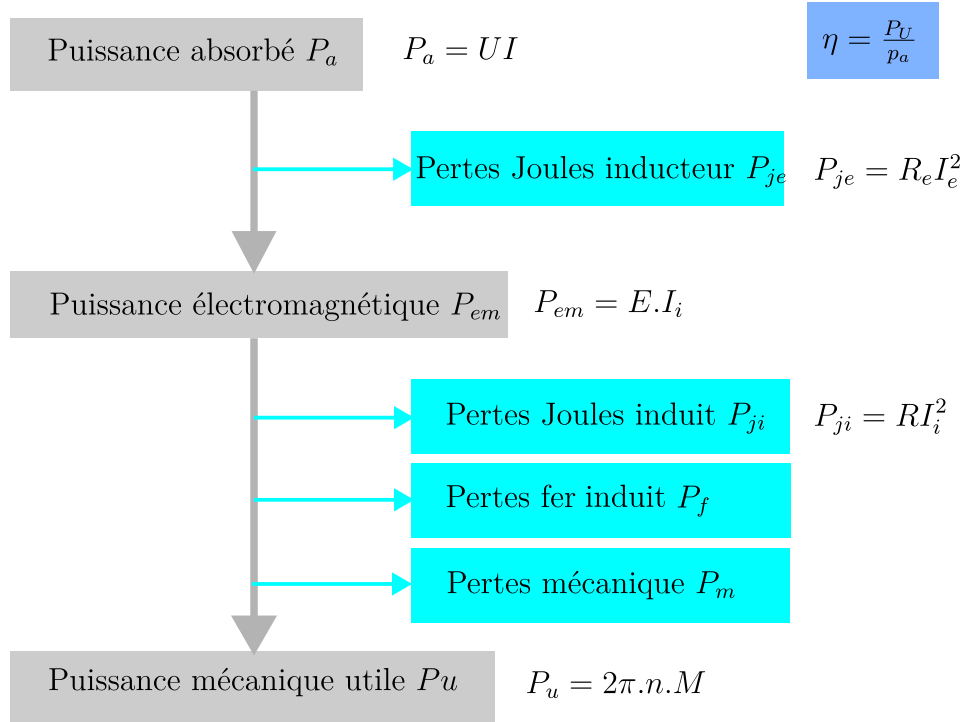


Figure 1.9: Bilan des puissances de la machine à courant continu

- **Puissance absorbée par l'induit :**

$$P_i = U_i I_i \quad (1.14)$$

U_i : La tension aux bornes de l'induit en volts (V).

I_i : L'intensité du courant dans l'induit en ampères (A).

- **Puissance absorbée par l'inducteur :**

$$P_e = U_e I_e \quad (1.15)$$

U_e : La tension d'alimentation de l'inducteur en volts (V).

I_e : L'intensité du courant dans l'inducteur en ampères (A).

Les pertes par effet Joule dans l'inducteur :

$$P_{je} = R_e I_e^2 \quad (1.16)$$

R_e : La résistance de l'inducteur en ohms (Ω).

I_e : L'intensité du courant dans l'inducteur en ampères (A).

La puissance électromagnétique :

$$P_{em} = E \cdot I_i = C_{em} \cdot \Omega \quad (1.17)$$

E La f.e.m du moteur en volts (V)

I L'intensité du courant dans l'induit en ampères (A)

C_{em} Le moment du couple électromagnétique en newton-mètres (N.m)

Ω La vitesse angulaire en radians par seconde (rad.s^{-1})

Les pertes dans l'induit :

- Pertes par effet joules dans l'induit :

$$P_{ji} = RI_i^2 \quad (1.18)$$

- Pertes fer induit : P_f
- Pertes mécaniques : P_m

La puissance mécanique utile :

Soit M le moment du couple moteur, on a :

$$P_u = 2\pi.n.M \quad (1.19)$$

n en tr/s où f en Hz

Le rendement

$$\eta = \frac{P_U}{P_a} \quad (1.20)$$

1.4 Générateurs AC

Les générateurs AC, également connus sous le nom d'alternateurs, jouent un rôle crucial dans la production d'électricité à partir de sources renouvelables telles que l'énergie éolienne, hydraulique, et bien d'autres encore. Dans le cas des éoliennes, les générateurs AC transforment l'énergie cinétique du vent en énergie électrique. Lorsque les pales d'une éolienne tournent, elles entraînent un rotor connecté au générateur. Ce mouvement de rotation crée un champ magnétique variable, induisant ainsi un courant alternatif dans les bobines du générateur.

1.4.1 Constitution d'un générateur

Un générateur électrique, qu'il soit AC ou DC, est essentiellement constitué de trois parties principales : le rotor, le stator et les composants de transport d'énergie.

Rotor et Stator :

Le rotor est la partie mobile du générateur qui tourne à l'intérieur du stator. Il est généralement constitué de bobines de matériau conducteur. Le stator est la partie fixe du générateur qui entoure le rotor. Le rotor est habituellement constitué d'un aimant permanent ou d'un électroaimant alimenté par courant continu.

Système de Conversion d'Énergie :

- Dans un générateur AC, les composants de transport d'énergie utilisent des bagues collectrices. Elles sont protégées de l'arbre par de l'isolation. Les balais relient les bagues collectrices aux circuits externes. Ils se trouvent à la surface extérieure des bagues collectrices (Voir figure [fig. 1.10](#)).
- Pour un générateur DC, les composants de transport d'énergie utilisent des commutateurs ou des diodes pour redresser le courant alternatif en courant continu. Cela peut être accompli par un système de collecteur et de balais (Voir figure [fig. 1.11](#)).

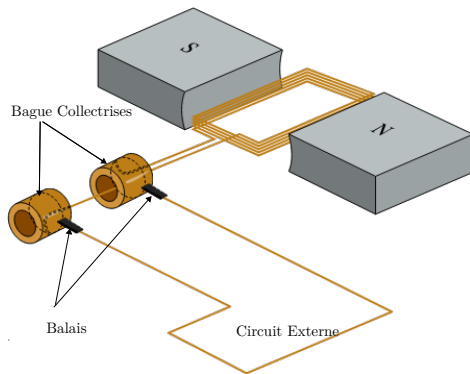


Figure 1.10: Schéma d'un générateur AC

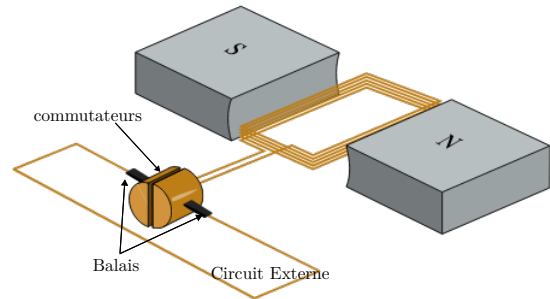


Figure 1.11: Schéma d'un générateur DC

1.4.2 Principe de fonctionnement d'un générateur AC

La bobine est tournée dans le champ magnétique pour générer un champ magnétique fort. Lorsque la bobine d'un côté se déplace vers le haut avec le champ électromagnétique, une force électromotrice (fem) est induite dans une direction. Alors que la rotation de la bobine se poursuit et que ce côté de la bobine se déplace vers le bas et qu'un autre côté de la bobine se déplace vers le haut, une fem est générée dans la direction opposée. La règle de la main droite de Fleming est utilisée pour déterminer la direction de la fem induite. Ce processus se répète pour chaque cycle et la fem générée est de type alternatif.

Fem induite dans un générateur AC

Le signal de sortie d'un générateur AC est montré dans la figure (fig. 1.12).

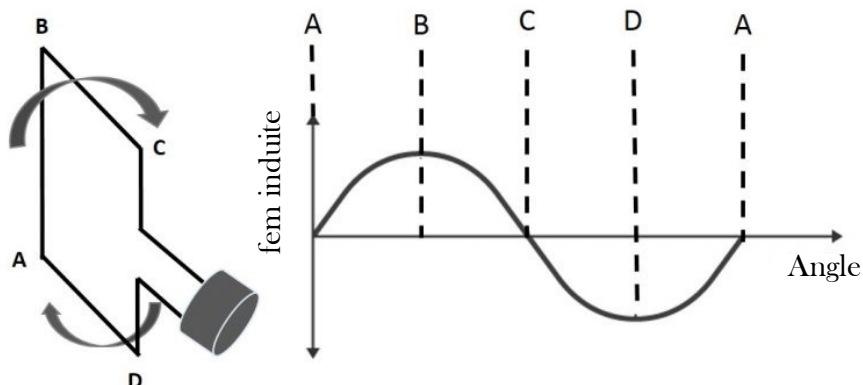


Figure 1.12: Fem induite dans un générateur AC

A - Lorsque la bobine atteint 0° , la bobine se déplace parallèlement à la direction du champ électromagnétique et, par conséquent, ne génère aucune fem.

B - Lorsque la bobine atteint 90° , la bobine se déplace à 90° par rapport au champ électromagnétique et induit ainsi une fem maximale.

C - Lorsque la bobine atteint 180° , la bobine se déplace de nouveau parallèlement au champ magnétique et n'induit donc aucune fem.

D - Lorsque la bobine est à 270° , la bobine se déplace à nouveau à 90° par rapport au champ magnétique et induit ainsi une fem maximale. Ici, la fem générée est contraire à celle de B.

A - Lorsque la bobine est à 360° , la bobine a effectué un tour complet et se déplace parallèlement au champ électromagnétique, induisant ainsi une fem nulle.

Calcul du courant alternatif induit

Considérez une bobine de forme rectangulaire avec 'N' spires qui tourne dans un champ magnétique uniforme 'B' avec une vitesse angulaire ω . L'angle entre le champ magnétique 'B' et la normale à la bobine à tout moment 't' est donné par $\theta = \omega t$.

Dans cette position, le flux magnétique est perpendiculaire au plan de la bobine et est donné par $B \cos \omega t$.

Le flux magnétique lié à une bobine de N spires est $\phi = B \cos \omega t \times A$, où A est l'aire de la bobine. La fem induite dans la bobine est donnée par les lois de l'induction électromagnétique de Faraday :

$$\epsilon = -\frac{d\phi}{dt} = -\frac{d(NBA \cos \omega t)}{dt} = NBA\omega |\sin \omega t| \quad (1.21)$$

Lorsque la bobine tourne de 90° , la valeur du sinus devient 1 et la fem induite sera maximale. Ainsi, l'équation (1.21) se réduit à :

$$\epsilon_0 = NB_m A \omega = NB_m A 2\pi f \quad (1.22)$$

Où B_m fait référence à la densité de flux maximale en Wb/m^2 , 'A' fait référence à l'aire de la bobine en m^2 , et f est la fréquence de rotation de la bobine en tours/seconde.

En substituant l'équation (1.22) dans (1.21), nous obtenons :

$$\epsilon = \epsilon_0 \sin \omega t \quad (1.23)$$

Le courant alternatif induit est donné par :

$$I = \frac{\epsilon}{R} = \frac{\epsilon_0 \sin \omega t}{R}. \quad (1.24)$$

1.4.3 Bilan de puissance

La figure (fig. 1.13) résume le bilan des puissances d'un générateur AC.

1.5 Les organes de liaisons et de transfert de puissance

[42, 2, 1] Les machines sont composées de composants élémentaires assemblés ensemble. La détection d'une défaillance potentielle peut être réalisée en surveillant les signaux provenant des capteurs qui mesurent les paramètres associés à ces composants.

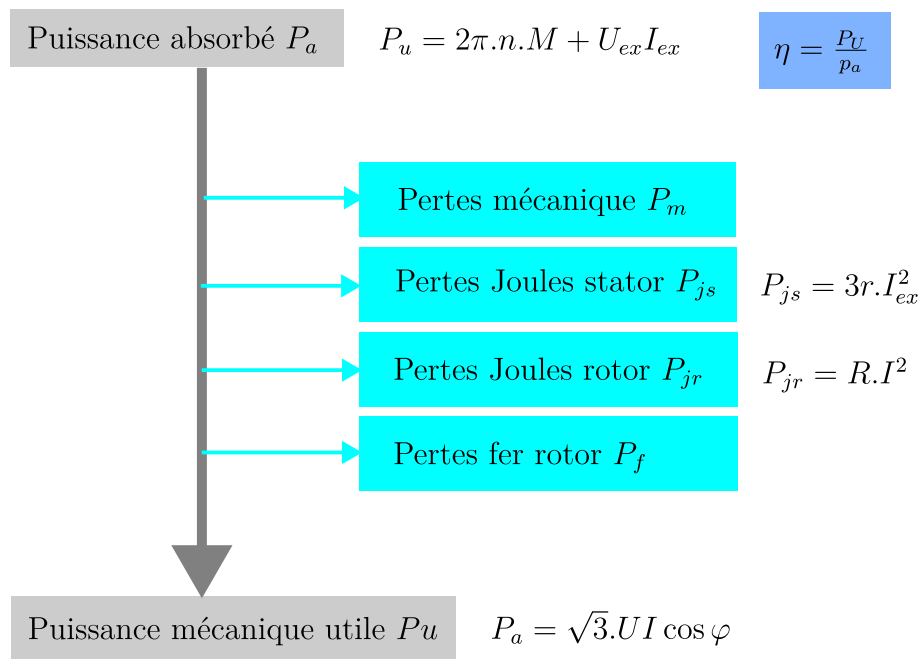


Figure 1.13: Bilan des puissances d'un générateur AC

1.5.1 Les accouplements

Ils relient les arbres en rotation de deux machines, pouvant être :

- rigides,
- à dents,

plus ou moins souples pour absorber les vibrations de torsion et les désalignements des deux machines à accoupler.

- **à cardan** : transmettant les vibrations de torsion mais permettant un désalignement important des deux arbres,
- **interruptibles** : embrayage, coupleurs.

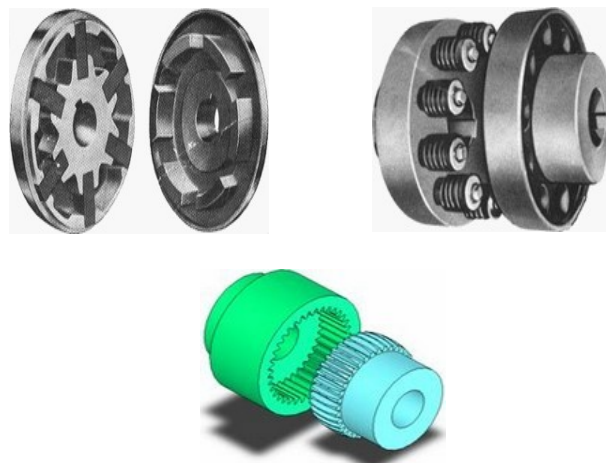


Figure 1.14: Différents types d'accouplements

Les différences entre les accouplements rigides ou à dents sont présentées dans le tableau (table 1.1). Les différents types d'accouplements sont montrés dans la figure (fig. 1.14).

Les défauts des accouplements rigides ou élastiques résultent d'un désalignement important des arbres, d'un manque de souplesse de l'accouplement ou d'un manque d'amortissement. La détection de ces défauts nécessite l'utilisation de techniques de mesure vibratoire, de positionnement des arbres et de température. Ils ont des conséquences sur l'ensemble des composants de la ligne d'arbre.

La modélisation de l'accouplement est réalisée à l'aide de la méthode des éléments finis. La

Table 1.1: Comparaison des caractéristiques des accouplements à dents et rigides

Caractéristique	Accouplement à dents	Accouplement rigide
Transmission de puissance	Par pignon et roue dentée	Manchon fretté
Montage	Très soigné	Normal
Graissage	De l'huile très pure est nécessaire. Problématique à partir de certaines vitesses périphériques	Supprimé
Usure des différentes parties	Possible en cas de montage peu soigné	inexistante
Nombre de paliers axiaux	Paliers axiaux séparés pour la turbine et le compresseur	Un seul palier axial est possible (côté turbine)
Contraintes des paliers axiaux	Influencée par le coefficient de frottement	Uniforme
Compensation de poussée	Impossible	Possible
Transmission de la dilatation différentielle	Non	Oui
Influence réciproque sur la vitesse de rotation critique	Faible	Normale

rigidité de flexion de l'accouplement peut être exprimée par la relation :

$$K = k_0 + f(\phi) \cdot g(\alpha) \quad (1.25)$$

1.5.2 Les éléments de transmission souple

Les transmissions par poulies et courroies, illustrées dans la figure (fig. 1.15), fonctionnent de manière silencieuse et sont principalement utilisées à des vitesses élevées avec de grands espacements possibles entre les poulies. Maintenir une tension initiale adéquate sur les courroies est essentiel pour assurer l'adhérence et la transmission du mouvement. Souvent, un système à entraxe ajustable ou un dispositif supplémentaire de tension (comme un galet enrouleur, etc.) est nécessaire pour régler la tension initiale et compenser l'étirement des courroies au fil du temps. À l'exception des courroies crantées, il y a généralement un léger glissement des courroies sur les poulies pendant le fonctionnement normal, ce qui peut entraîner une légère variation du rapport

de transmission. Celui-ci n'est pas exactement équivalent au rapport des diamètres des deux poulies.

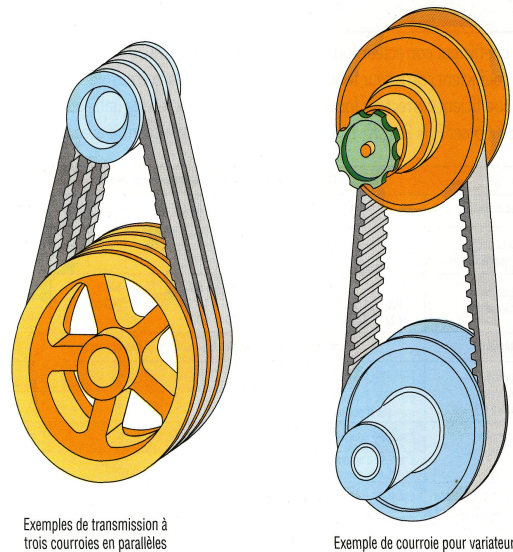


Figure 1.15: Les courroies de transmission

Les courroies

Ce sont les courroies plates, en V ou crantées, ainsi que les chaînes. Une courroie peut engendrer plusieurs fréquences.

Chaînes et Courroies crantées

Les fréquences caractéristiques des chaînes et des courroies crantées sont liées au nombre de dents des poulies dentées ou crantées ainsi qu'aux vitesses de rotation de ces poulies.

1.5.3 Les Engrenages

Un mécanisme d'engrenage est un système mécanique constitué de deux roues dentées en prise qui peuvent servir à deux fonctions principales :

- Transmettre le mouvement de rotation entre elles.
- Propulser un fluide (dans le cas d'une pompe à engrenages).

Lorsqu'il s'agit de transmettre le mouvement, les deux roues dentées sont en contact direct et transfèrent de la puissance par frottement. Un engrenage peut être composé d'un pignon (la roue la plus petite ou unique), ainsi que d'une roue, d'une crémaillère, d'une couronne ou d'une vis (voir figure [fig. 1.16](#)). Lorsqu'il y a plus de deux roues dentées en interaction, on parle de train d'engrenages.

1.5.4 Généralités sur les roulements à billes

[68, 2, 1] Un roulement à billes est constitué de deux bagues coaxiales séparées par des billes, légèrement lubrifiées, et maintenues en position par une cage.

En général, les roulements comportent quatre éléments principaux : une bague externe, une bague interne, des éléments roulants et un système de lubrification. Selon l'application, les

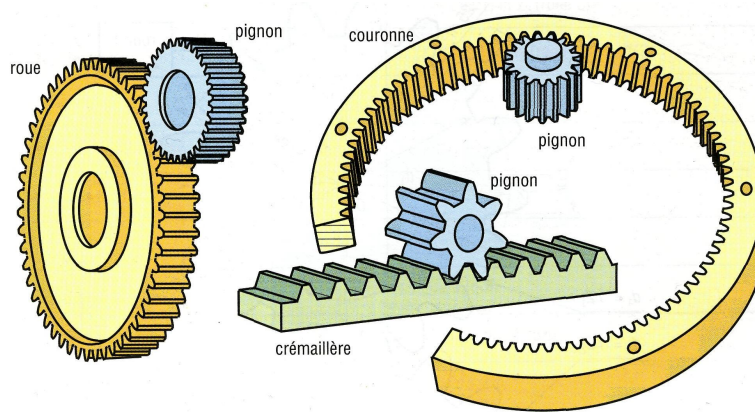


Figure 1.16: Les types d'engrenage

éléments roulants peuvent être des billes, des rouleaux ou des aiguilles. Chaque type de roulement présente des caractéristiques spécifiques concernant les charges radiales et axiales supportées, la vitesse limite et le degré de désalignement toléré.

Notre étude des roulements se concentrera sur un modèle couramment employé, à savoir les

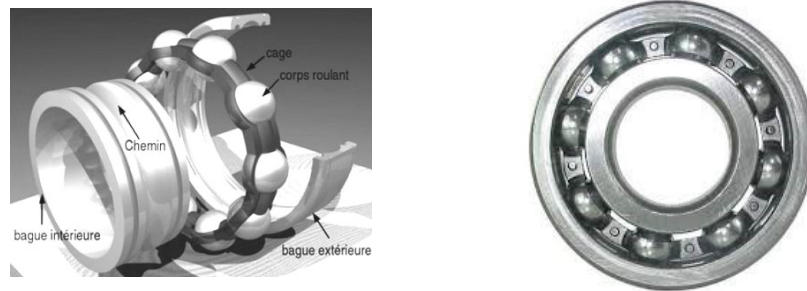


Figure 1.17: Les roulements à billes

roulements métalliques lubrifiés, à une seule rangée de billes, à contact radial, qui sont les plus fréquents dans les machines électriques de puissance moyenne. Ces roulements comprennent une bague externe, une bague interne et une série de billes maintenues par une cage, ainsi que de la graisse pour la lubrification. Une vue éclatée de ce type de roulement est illustrée à la figure (fig. 1.17). Pour la machine étudiée, la bague externe est fixe et en contact avec le stator, tandis que la bague interne est mobile et en contact avec le rotor. Les billes sont en contact, d'une part, avec la piste de roulement interne et, d'autre part, avec la piste de roulement externe. On utilisera indifféremment les termes piste de roulement, bande de roulement ou chemin de roulement.

1.6 Les Défauts mécaniques des principaux organes de liaison

1.6.1 Le Déséquilibre massique ou (Balourd)

[26, 2, 1] indépendamment de la précision mise en œuvre lors de la fabrication des machines, il est impossible d'aligner parfaitement l'axe de rotation avec le centre de gravité de chaque section élémentaire du rotor (fig. 1.18). On parle alors de déséquilibre du rotor.

Les déséquilibres sont souvent causés par des imperfections dans l'usinage, l'assemblage et le montage, ou résultent du déséquilibre de la masse de la charge que le moteur doit entraîner. Imaginons un rotor en parfait équilibre. En y ajoutant une masse à un emplacement spécifique

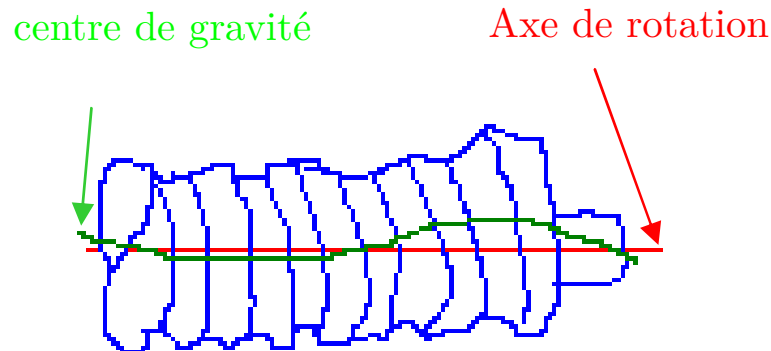


Figure 1.18: Image Vibratoire théorique d'un balourd

du rotor (voir Figure [fig. 1.19](#)), on crée un déséquilibre statique. Pendant la rotation du rotor, cette masse génère une force radiale qui est proportionnelle à la vitesse de rotation.

Dans le cas du balourd statique, le rotor reste en équilibre parfait lorsqu'il est immobile.

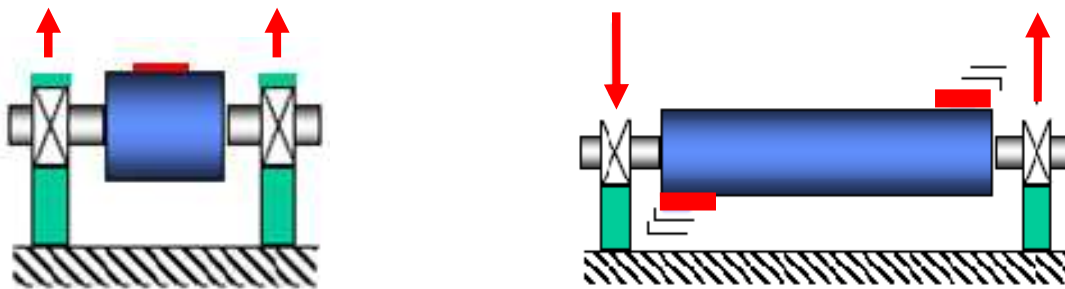


Figure 1.19: (a) : À gauche balourd statique, (b) : À droite balourd dynamique

Cependant, dès que le rotor commence à tourner, les deux forces impliquées créent un couple. Ces forces, bien que de même magnitude et tournant à la même vitesse de rotation, se dirigent dans des sens opposés.

1.6.2 Défaut d'alignement

[26, 2, 1] Le désalignement constitue l'une des principales raisons de la diminution de la longévité des équipements. Cela peut impliquer soit deux arbres connectés par un accouplement, soit deux roulements supportant le même axe.

1.6.2.1 Désalignement d'arbres accouplés

Le désalignement est un problème plus courant que le déséquilibre, et la raison en est assez évidente. Bien que l'on utilise des paliers auto-alignants et des accouplements flexibles, aligner deux arbres et leurs paliers de manière à éliminer complètement toute force susceptible de provoquer des vibrations reste une tâche difficile.

Il existe deux formes de désalignement : celui qui est radial et celui qui est angulaire. (Voir Figure [fig. 1.20](#))

- **Désalignement radial :** Ceci se produit lorsque les axes de rotation des deux machines ont une orientation angulaire identique, mais sont situés à des niveaux verticaux différents.

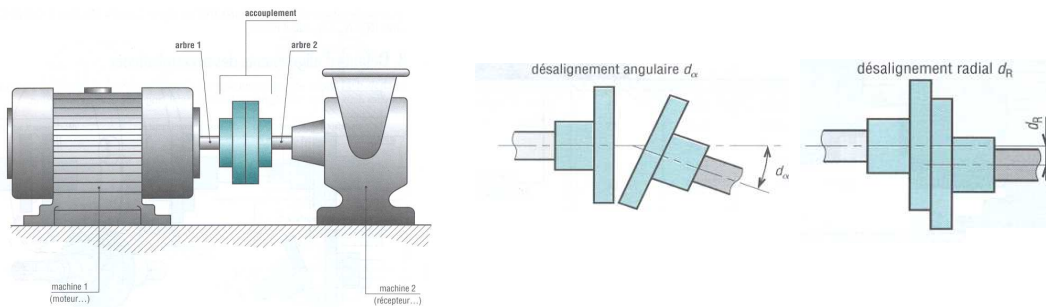


Figure 1.20: Image illustrant le désalignement radial et angulaire

- **Désalignement angulaire :** Cette situation survient lorsque les axes de rotation de deux machines sont orientés à des angles distincts.

1.6.2.2 Désalignement des paliers

Les axes des deux paliers d'un même corps de machine ne sont pas alignés concentriquement (voir Figure [fig. 1.21](#)) Cette divergence peut résulter d'un assemblage défectueux d'un palier, de l'alignement inadéquat des attaches de fixation, ou encore d'une déformation du châssis (par exemple due à des variations thermiques), entraînant ainsi une déformation de l'arbre du rotor.

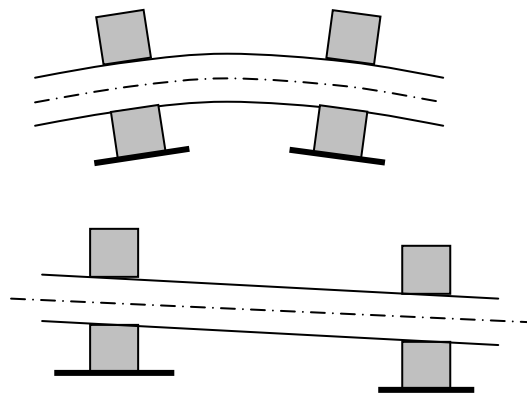


Figure 1.21: Désalignement de paliers se traduisant par une flexion de l'arbre

1.6.3 Mauvais serrage mécanique

[26, 2, 1] Lorsque le serrage d'un palier est insuffisant ou qu'il montre une capacité de déplacement partiel dans le plan radial, cela provoque un déséquilibre initial susceptible d'augmenter l'amplitude des vibrations, selon l'importance du desserrage du palier (Voir Figure [fig. 1.22](#)).

1.6.4 Défauts dans les engrenages

[26, 2, 1] Un engrenage est formé de deux roues dentées qui s'engrènent pour transmettre de la puissance entre deux arbres adjacents, avec un rapport de vitesse constant. Selon l'alignement relatif des arbres, il existe trois catégories d'engrenages (voir Figure [fig. 1.23](#)) :

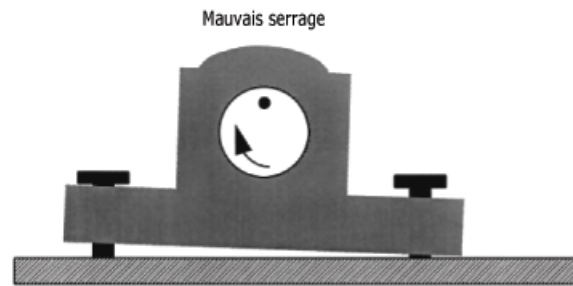


Figure 1.22: Image de mauvais serrage mécanique

- Engrenages parallèles
- Engrenages coniques
- Système roue /vis sans fin

Les configurations des dents des engrenages peuvent varier, se présentant sous forme droite, hélicoïdale ou à chevrons. Les charges appliquées diffèrent en fonction :

- Engrenages parallèles à denture rectiligne ou à chevrons : forces uniquement radiales.
- Engrenages hélicoïdaux, coniques et à vis sans fin : charges mixtes (radiales et axiales).

Les engrenages sont des composants utilisés pour transmettre le mouvement, susceptibles de présenter divers défauts comme :

- Altérations des dents taillées.
- Ecaillages aux dents cémentées.
- Apparition de fissures.
- Fractures résultant de fissures.
- Grippages entraînant des dégradations significatives.

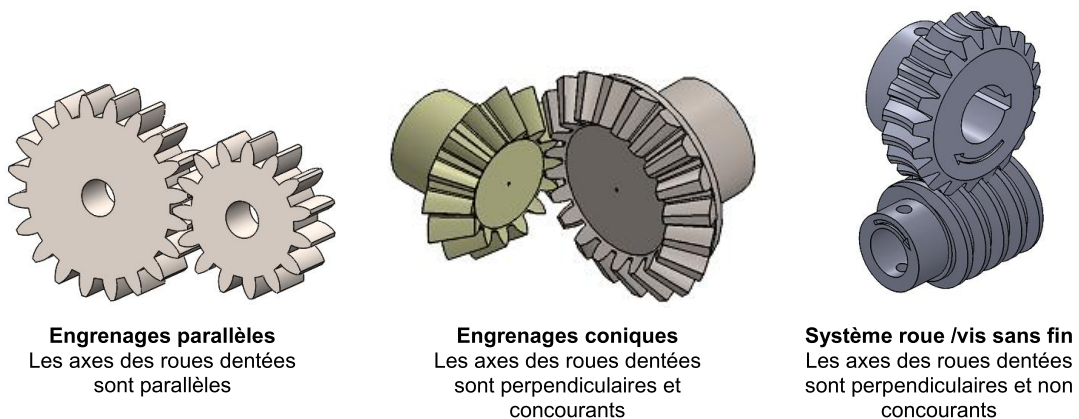


Figure 1.23: Différents types d'engrenage

Les principales irrégularités observées sur la surface des dents d'un réducteur simple incluent les imperfections réparties sur l'ensemble de la denture et les défauts localisés sur des zones spécifiques. De manière générale, les altérations superficielles des engrenages résultent de divers phénomènes pouvant survenir pendant l'exploitation du système.

1.6.5 Défauts des roulements

[20, 60, 64, 26] Fréquemment utilisés pour orienter la rotation, de nombreux dispositifs mécaniques intègrent des paliers à roulements en raison de leur efficacité supérieure dans la réduction des frictions pour les éléments en mouvement rotatif. Les roulements sont typiquement composés de :

- Bague extérieure.
- Bague intérieure.
- Cage.
- Bille.

La figure (fig. 1.24) illustre les différents éléments constitutifs d'un roulement à billes.

Souvent, la détérioration se manifeste par un écaillage sur l'une des voies ou sur un élément

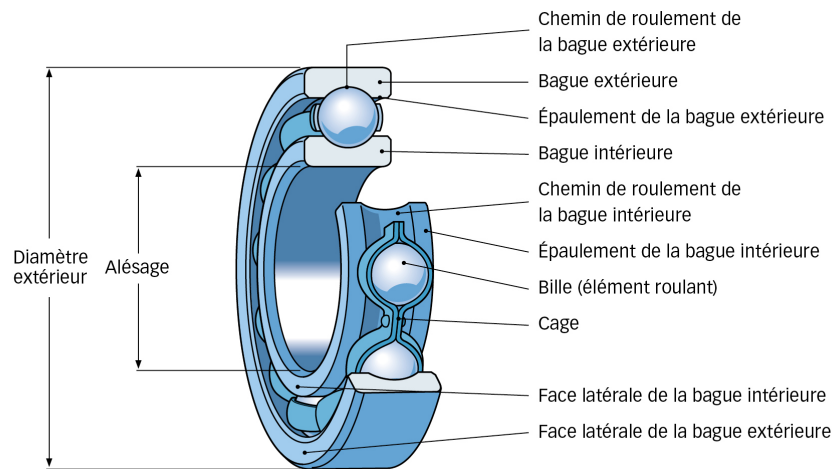


Figure 1.24: Roulement à billes

mobile du roulement, entraînant une percussion à chaque rotation.

1.6.5.1 Ecaillage

Les contraintes élevées entre les parties mobiles et les pièces fixes entraînent des fissures. Ces interactions superficielles génèrent des écaillages qui peuvent entraîner des dégradations matérielles. La configuration de la zone de contact est influencée par la qualité de l'ajustement du roulement.

Le phénomène d'écaillage peut être observé dans les situations suivantes :

- Déformation du rotor.
- Application excessive de force sur les bagues intérieures.
- Application excessive de force sur les bagues extérieures.

La figure (fig. 1.25) illustre un défaut d'écaillage réel des différents éléments d'un roulement à billes.

1.6.5.2 Détérioration indépendante de la fatigue

Usure : Que les roulements soient graissés ou non, il n'est pas garanti que les éléments roulants ne glissent pas sur les bagues intérieures et extérieures. Le film de lubrifiant peut se rompre, provoquant ainsi des glissements et conduisant à l'usure.



Figure 1.25: Défaut d'écaillage réel

Grippage : Le blocage survient lorsqu'il y a une insuffisance de lubrification ; les surfaces métalliques en contact génèrent de la chaleur, favorisant ainsi la formation de petites soudures. Un durcissement de la graisse, une usure qui provoque le déplacement non parallèle des rouleaux ou des aiguilles par rapport aux bagues, peuvent entraîner un blocage.

Avarie de cage : Les structures sont délicates. Lorsqu'on assemble le roulement, elles peuvent se déformer, ce qui risque de provoquer le blocage des pièces mobiles. De plus, l'assemblage peut entraîner leur rupture.

- Écoulement de la bague extérieure.
- Écoulement de la bague intérieure.
- Écoulement simultané des deux bagues.

1.7 Synthèse des Méthodes de Diagnostic

1.7.1 Méthodes de diagnostic avec modèle analytique

Les techniques de détection utilisent des informations préalables sur le système, nécessitant la maîtrise des modèles et des paramètres décrivant le processus physique étudié. En comparant les signaux expérimentaux du fonctionnement du système avec ceux générés par les modèles (ou les paramètres estimés), ces méthodes permettent de repérer et d'identifier les éventuelles défaillances à venir.

Pour appliquer ces méthodes, il est nécessaire de comprendre les limites de variation des modèles et des paramètres en fonction des conditions opérationnelles du système. Ces approches peuvent être regroupées en trois catégories :

- Méthodes d'estimation d'état
- Approche de génération de résidus
- Méthodologies d'identification

Pour cette première catégorie de méthodes de diagnostic, il est essentiel de comprendre le comportement dynamique de la machine en utilisant un modèle basé sur la connaissance. Cependant, ces techniques ne sont pas capables d'identifier tous les types de défauts de la machine, comme les défaillances des roulements, par exemple.

1.7.2 Méthodes de diagnostic sans modèle à priori

Ces approches de diagnostic non basées sur des modèles consistent à surveiller et analyser diverses grandeurs telles que les courants (par exemple, par l'analyse de la signature du courant,

MCSA), les vibrations, les flux et le couple. Elles ne nécessitent pas de modèle précis du système, mais se fondent sur la reconnaissance de signatures caractéristiques. Les signatures de défauts, obtenues soit par modélisation, soit par des mesures sur des maquettes, sont généralement stockées dans une base de données. L'analyse s'effectue par une interprétation de type signal ou via un système expert. Les défauts détectés par ces méthodes incluent la rupture de barres ou de segments d'anneaux au rotor, l'excentricité statique et dynamique, la défaillance des paliers et le court-circuit entre spires au stator.

1.7.3 Méthode par Traitement de Signal (MTS)

Pour des raisons de simplicité et d'efficacité, l'approche signal est largement adoptée en diagnostic de nos jours. Cette méthode s'appuie sur une compréhension du comportement d'un système en bon état, et les signaux mesurés sont ensuite comparés à cette référence. Parmi les techniques courantes d'analyse des signaux en régime transitoire et permanent, on trouve l'analyse spectrale, le spectrogramme, l'analyse temporelle et la distribution de Wigner-Ville [4].

1.7.4 Méthode par Intelligence Artificielle (MIA)

L'utilisation de l'intelligence artificielle dans la supervision et le diagnostic connaît une croissance marquée, contribuant ainsi à améliorer l'efficacité et la fiabilité des diagnostics. Dans le secteur de la surveillance des machines électriques, ces technologies sont de plus en plus adoptées par les industriels et les chercheurs pour optimiser les performances des systèmes [3, 8]. Le terme "Intelligence Artificielle" englobe diverses approches comme les systèmes experts, les réseaux de neurones et la logique floue, qui peuvent être employées séparément ou en combinaison pour renforcer leurs performances.

1.8 Conclusion

Dans ce chapitre, nous avons présenté en première partie les principaux éléments constitutifs du moteur asynchrone, du moteur à courant continu et de la génératrice AC. Ensuite, dans la deuxième partie, nous avons exposé brièvement les éléments traditionnels liés à l'arbre rotorique, nécessaires pour assurer la transmission du mouvement et de la puissance mécanique à la charge du procédé. La troisième partie s'est concentrée sur les principales défaillances mécaniques des organes de liaison, lesquelles entraînent souvent un arrêt intempestif du procédé. Afin de remédier à ce problème, le diagnostic de défaut a gagné en importance dans les milieux industriels. Pour chaque type de défaut, nous avons identifié les causes principales, les mécanismes et les symptômes associés, ainsi que les techniques de détection correspondantes.

Étant donné la complexité des modèles analytiques des défauts mécaniques simples et en particulier combinés auxquels on s'intéresse dans le cadre du travail de diagnostic et de détection prédictives, nous avons opté pour différentes approches sans modèle a priori, utilisant le traitement du signal et l'apprentissage automatique (Deep Learning et Machine Learning).

Dans le deuxième chapitre, nous aborderons les différentes méthodes de traitement du signal appliquées au diagnostic des machines électriques. Les techniques d'analyse dans les domaines temporel, fréquentiel et temps-fréquence seront présentées, en soulignant leurs principes de fonctionnement ainsi que leur rôle dans l'extraction des caractéristiques représentatives des défauts. Cette étude permettra d'identifier les approches les plus adaptées à la surveillance et au diagnostic des anomalies mécaniques et électriques.

Techniques de diagnostic basées sur le traitement du signal

2.1 Introduction

Le traitement du signal joue un rôle essentiel dans le diagnostic des machines électriques en permettant l'extraction d'informations pertinentes à partir des signaux de fonctionnement. Ce chapitre présente les principales techniques d'analyse dans les domaines temporel, fréquentiel et temps-fréquence, largement utilisées pour la détection et l'identification des défauts. Les principes fondamentaux de ces méthodes ainsi que leur contribution à l'extraction des caractéristiques représentatives des anomalies mécaniques et électriques seront étudiés. Cette analyse permettra de mettre en évidence les approches les plus adaptées à la surveillance de l'état des machines et à la maintenance prédictive.

2.2 Techniques d'analyse du signal pour le diagnostic des défauts mécaniques

2.2.1 Transformée de Fourier (FT)

Parmi les méthodes d'analyse du courant pour la détection précoce et le diagnostic des anomalies mécaniques, la FFT (Fast Fourier Transform) est la plus utilisée. Elle contient des informations essentielles concernant le signal courant analysé.

Généralement, la FFT décompose un signal en une somme de fonctions exponentielles complexes de différentes fréquences, comme illustré dans la figure (fig. 2.1). Les coefficients obtenus par la décomposition correspondent à ceux de la transformée de Fourier $X(f)$. La valeur d'un

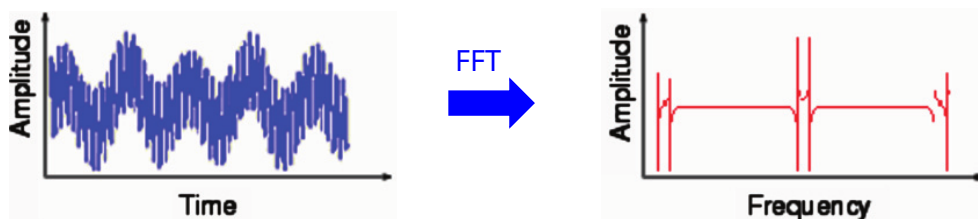


Figure 2.1: Concept de la Transformée de Fourier

coefficient de Fourier à la fréquence f s'obtient en multipliant le signal $x(t)$ par une exponentielle de fréquence f et en l'intégrant sur tout le temps.

$$X(f) = \int_{-\infty}^{\infty} x(t) \cdot e^{-j2\pi ft} dt \quad (2.1)$$

L'analyse spectrale par l'application de la FFT du courant statorique permet d'identifier les signatures fréquentielles causées par les oscillations de couple pendant le fonctionnement dégradé de la machine. L'émergence des raies spectrales aux fréquences $|f_s \pm k \cdot f_c|$ est très significative pour la détection des défauts mécaniques simples et multiples affectant le système électromécanique [65, 53, 61]. Généralement, les fréquences caractéristiques des anomalies mécaniques sont fournies par la formule suivante :

$$f_d = f_1 \left\{ k \pm \frac{n}{p} (1 - s) \right\} \quad (2.2)$$

Où p : paires de pôles ; s : rotor glissement fréquence ; f_1 : fréquence fondamental (Hz) ; f_d : la fréquence des défauts associés en (Hz) ; $k = 1, 3, 5 \dots$ et $n = 1, 2, 3 \dots$, p : est le nombre de paires de pôles.

Cependant, cette technique permet de déterminer les fréquences présentes, mais elle ne fournit pas d'information sur les instants où ces fréquences apparaissent. Par conséquent, elle est inadaptée aux signaux non stationnaires, bien que ces derniers soient plus courants que les signaux stationnaires.

L'adaptation de techniques avancées d'analyse du signal telles que la STFT, l'analyse de Gabor et la CTW Dans le domaine temporo-fréquentiel est fortement recommandée. Ces outils d'analyse temps-fréquence permettent d'obtenir une image à haute résolution, mais nécessitent un traitement plus complexe que les techniques de diagnostic classiques de la FFT [14, 10, 62].

2.2.2 Transformée de Fourier à court terme (STFT)

La transformée de Fourier à court terme offre des représentations cartographiques d'un signal en deux dimensions (temps - fréquence), comme illustré dans la figure (fig. 2.2). La transformée de

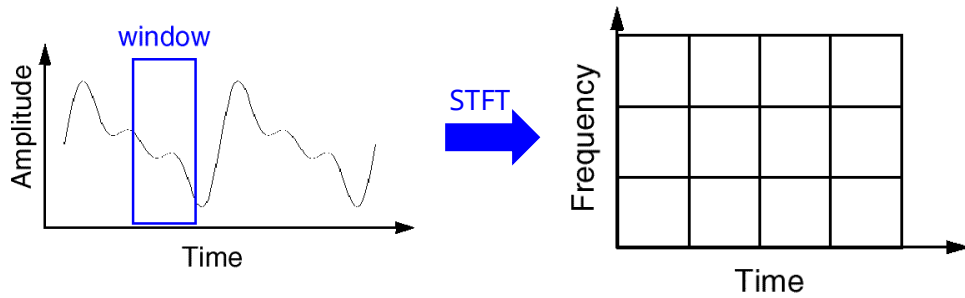


Figure 2.2: Passage de la caractéristique temporelle à la STFT

Fourier à court terme (TFCT), également désignée par le terme de TF à fenêtre glissante, C'est une expansion de la méthode de la transformée de Fourier classique pour traiter des données non stationnaires [62]. Le concept consiste à analyser le signal par fenêtres successives. La transformée de Fourier de chaque portion de signal fenêtrée est déterminée de la manière suivante :

$$STFT_x^w(\tau, f) = \int x(t) \cdot w^*(t - \tau) \cdot e^{-j2\pi ft} dt \quad (2.3)$$

$x(t)$: correspond au signal temporel échantillonné.

$w(t)$: est la fenêtre temporelle a une largeur de T et est centrée autour de τ , ce qui permet d'extraire une partie du signal.

w^* représente le complexe conjugué de w .

$$S_x(t, f) = \left| x(t) \cdot h(t - \tau) \cdot e^{-j\omega t} \right|^2 \quad (2.4)$$

Le spectrogramme de la transformée de Fourier à court terme (TFCT) peut être exploité comme un outil pour l'analyse temps-fréquentielle du signal courant transitoire à des fins de diagnostic et de détection des anomalies mécaniques affectant un système d'entraînement électromécanique, tel qu'un moteur à induction avec réducteur.

Cependant, il est impossible d'obtenir simultanément les résolutions temporelles et fréquentielles. Ce problème est intimement lié au concept d'incertitude d'Heisenberg, qui exprime Les résolutions en temps et en fréquence ne peuvent pas être simultanément arbitrairement petites, puisque :

$$\Delta t \cdot \Delta f \geq \frac{1}{4\pi} \quad \text{avec} \quad \begin{cases} \Delta t^2 = \frac{\int t^2 |w(t)|^2 dt}{\int |w(t)|^2 dt} \\ \Delta f^2 = \frac{\int f^2 |w(f)|^2 df}{\int |w(f)|^2 df} \end{cases} \quad (2.5)$$

2.2.3 Spectrogramme de Gabor

Tout comme la transformée de Fourier (TF), la transformée de Gabor (TG) Un signal contient toutes les informations qu'il transporte. Le spectrogramme de Gabor est utilisé pour estimer la teneur en fréquence d'un signal. Il représente, comme indiqué dans l'équation (2.6), un signal du domaine temporel, $s(t)$, comme une combinaison linéaire de fonctions élémentaires [58].

$$s(t) = \sum_{m=0}^{m-1} \sum_{n=0}^{n-1} c_{m,n} h_{m,n}(t) \quad (2.6)$$

Où :

$c_{m,n}$ sont les coefficients associés aux fonctions de base $h_{m,n}(t)$.

$h_{m,n}(t)$ Est la fonction élémentaire de la fréquence de temps.

L'idée du spectrogramme de la transformée de Gabor est simple et peut constituer une bonne alternative au spectrogramme de la transformée de Fourier à court terme (TFCT) et à la distribution de Wigner-Ville. Il propose une résolution temporelle et fréquentielle améliorée. (RTF) pour le diagnostic et la détection prédictive des anomalies.

2.2.4 Transformée en ondelettes continue (CWT)

Elle est caractérisée comme suit :

$$CWT_x^\psi(\tau, s) = \frac{1}{\sqrt{|s|}} \cdot \int_t x(t) \cdot \psi^* \left(\frac{t - \tau}{s} \right) dt \quad (2.7)$$

τ est défini comme le coefficient de translation, un nombre réel. Le paramètre s représente le coefficient d'échelle, aussi appelé facteur de dilatation de l'ondelette mère ψ , un nombre réel. La fonction $\psi(t)$ désigne l'ondelette mère, et Ψ^* est son complexe conjugué. Les coefficients d'ondelettes $CWT_x^\psi(\tau, s)$ sont ainsi déterminés.

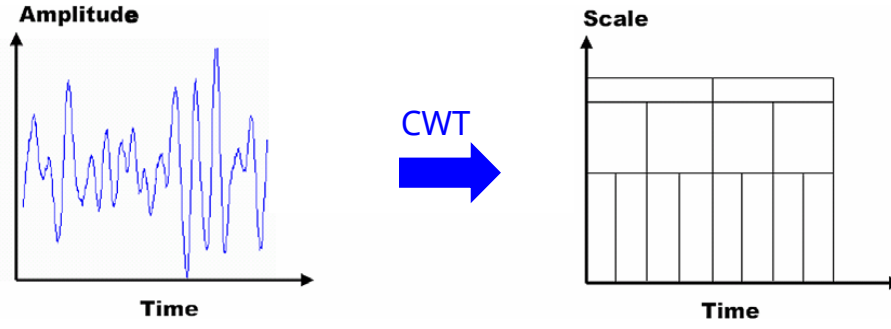


Figure 2.3: Principe de la Transformée d'Ondelettes Continues

L'amélioration de l'analyse de la STFT peut se faire par la transformée en ondelettes (TO), dont l'idée principale est de définir une analyse similaire, mais en faisant dépendre la largeur de la fenêtre d'analyse de sa position, tout en régulant l'indépendance de l'analyse dans le domaine temporel ou fréquentiel (voir Figure fig. 2.3), en respectant le principe d'incertitude d'Heisenberg. L'analyse par ondelettes est basée sur le concept d'échelle. Ainsi, La méthode de transformation en ondelettes continue offre une solution à ce problème. Bien que la STFT et la TO puissent constituer une alternative à l'analyse de Fourier traditionnelle, l'analyse de Fourier reste plus performante dans certains domaines. Dans certains cas spécifiques, il devient indispensable d'explorer des méthodes qui se rapprochent davantage de l'optimalité, tout en s'efforçant de maintenir les avantages de ces approches. [27, 22, 40].

2.2.5 Transformée en ondelettes discrète (DWT)

La Transformée en Ondelettes Discrète (DWT) permet d'appliquer une analyse multi-résolution sur un signal vibratoire. On peut considérer ce processus comme une décomposition d'un signal original à travers deux filtres complémentaires : un filtre passe-haut (H) et un filtre passe-bas (L). Le filtre passe-bas représente l'approximation, correspondant aux basses fréquences du signal, tandis que le filtre passe-haut représente les détails, correspondant aux hautes fréquences du signal S ($s_1, s_2, \dots, s_n$) (voir Figure fig. 2.4). Ainsi, l'ondelette de Daubechies est utilisée pour calculer les coefficients d'approximation et de détail à chaque niveau en utilisant l'expression suivante :

$$s(t) = \sum_i \alpha_i^n \psi_i^n(t) + \sum_{j=1}^n \sum_i \beta_i^j \varphi_i^j(t) = a_n + d_n + \dots + d_1 \quad (2.8)$$

α, β représentent les échelles et les coefficients de l'ondelette,

ψ_n, ψ_j sont respectivement la fonction décomposée au niveau n et la fonction d'ondelette au niveau j ,

n représente le niveau de décomposition. a_n indique le signal d'approximation correspondant au niveau n , tandis que d_n désigne le détail du signal au même niveau.

Plusieurs travaux de recherche ont démontré que la DWT (Transformée en Ondelettes Discrète) fournit un outil puissant pour le diagnostic et la surveillance des défauts car elle fournit des informations temps-fréquence sur le signal [29, 6].

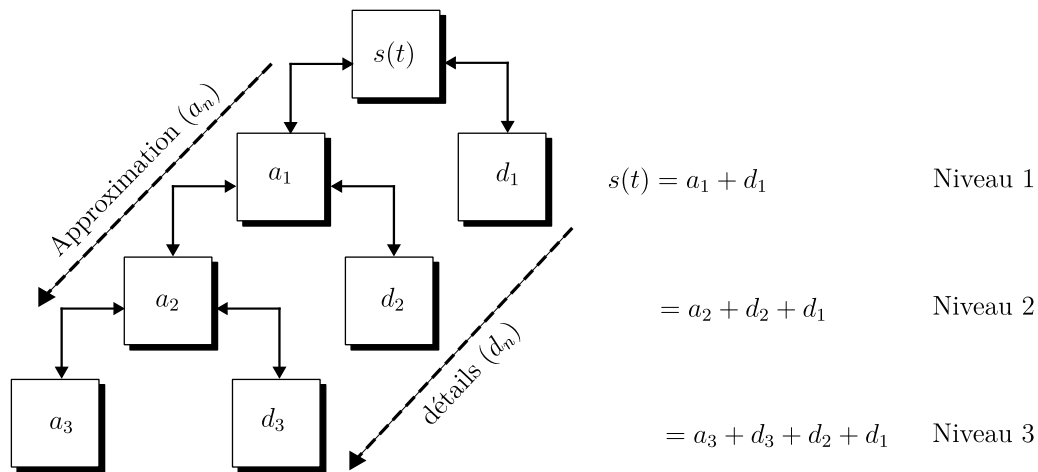


Figure 2.4: Principe de la Transformée d'Ondelettes Discrète

2.3 Signatures Vibratoires des Principaux Défauts

2.3.1 Déséquilibre Massique ou Balourd

[42, 26, 2] Quelle que soit l'attention portée à la fabrication des machines, il est impossible d'aligner parfaitement l'axe de rotation en relation avec le centre de gravité de chaque segment élémentaire du rotor. En raison de cette non-concentricité, des forces centrifuges, proportionnelles à la vitesse de rotation, s'exercent conformément à la relation $F = Mr\omega^2$, ce qui entraîne la déformation du rotor. Le rotor est alors considéré comme déséquilibré. Ce déséquilibre résulte habituellement d'imperfections lors de l'usinage, de l'assemblage, du démontage ou découle de :

- Une altération mécanique peut résulter de la perte d'une ailette, de l'érosion ou de l'encrassement.
- Une altération thermique peut être causée par des déformations dues à des dilatations des composants du rotor ou à des variations de températures à des emplacements précis.

Lorsqu'on évalue l'amplitude du signal vibratoire fourni par un capteur positionné sur le palier supportant un rotor déséquilibré (voir figure fig. 2.5), on observe que l'amplitude atteint son maximum lorsque la masse se trouve en haut du rotor, et son minimum lorsqu'elle est en bas, répétant ce cycle à chaque rotation du rotor. Le signal vibratoire présente ainsi une périodicité d'un tour complet, ce qui correspond à une fréquence similaire à celle du mouvement de rotation du rotor [63]. La présence de balourd dans un plan radial génère une vibration dont le spectre

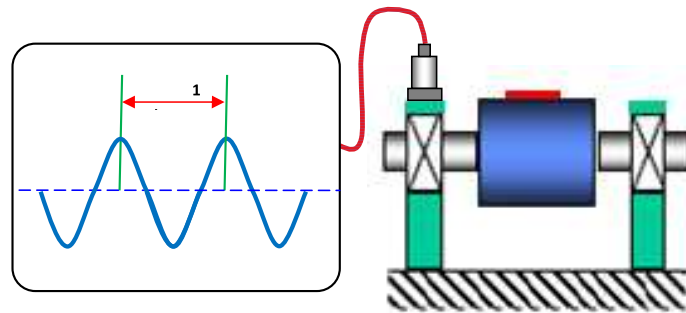


Figure 2.5: Signal vibratoire généré par un balourd

comporte une composante principale à la fréquence de rotation F_r . Cette composante se manifeste

comme le pic le plus élevé, accompagné de pics d'amplitudes plus faibles aux harmoniques de F_r (voir figure fig. 2.6). La bande de mesure est prévue pour tenir compte des petites variations autour de la vitesse de rotation indiquée par le fabricant.

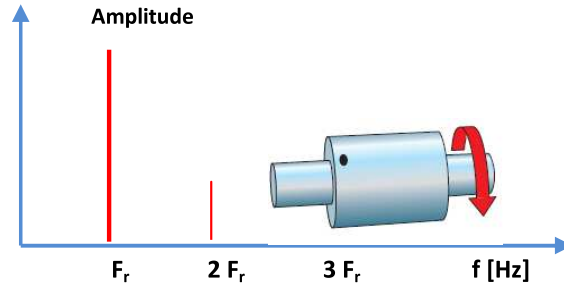


Figure 2.6: Spectre théorique d'un défaut de balourd

2.3.2 Désalignement

[42, 26, 2] Le désalignement est un problème plus fréquent que le balourd, et la raison en est bien simple. Malgré l'emploi de paliers auto-alignants et d'accouplements flexibles, il représente un défi de mettre en ligne deux arbres ainsi que leurs paliers, afin d'assurer qu'il n'y ait aucune force qui puisse causer la vibration. Un désalignement provoque des vibrations aux fréquences de rotation, ainsi qu'aux harmoniques d'ordre 2, 3 et 4 de la fréquence de rotation. Un désalignement se manifeste par des niveaux :

- Axiaux très élevés, le double de la fréquence de rotation..
- Radiaux élevés, le double de la fréquence de rotation.

On distingue trois types de désalignement :

2.3.2.1 Désalignement parallèle

Le graphique représentant le signal temporel associé à un problème de désalignement est illustré dans la figure (fig. 2.7). Ce signal présente une composante périodique à la fréquence de rotation (avec une période égale à un tour), ainsi que des composantes récurrentes à des intervalles de $1/2$ tour et $1/3$ tour. Cela se manifeste généralement par la présence de composantes ayant des fréquences correspondant à des multiples de la fréquence de rotation (2, 3 voire 4 fois), avec des amplitudes plus élevées que celle de la composante principale.

Un désalignement est détecté par un pic d'amplitude notable à environ deux fois la fréquence de rotation (parfois trois ou quatre fois). Il y a une vibration dans la direction radiale à la fréquence de rotation d'ordre 2 (parfois d'ordre 3, rarement d'ordre 4), présentant des amplitudes plus élevées que celles de la composante principale (voir figure fig. 2.7). Ce phénomène se présente de manière similaire dans la direction axiale. Le désalignement parallèle se caractérise par une vibration radiale élevée.

2.3.2.2 Désalignement angulaire

Il survient une vibration axiale associée aux harmoniques d'ordre 1, 2, 3 ou 4 de la fréquence de rotation, qui ont des amplitudes plus grandes que celles des composantes radiales correspondantes (voir figure fig. 2.8). Le désalignement angulaire se caractérise par une vibration axiale élevée.

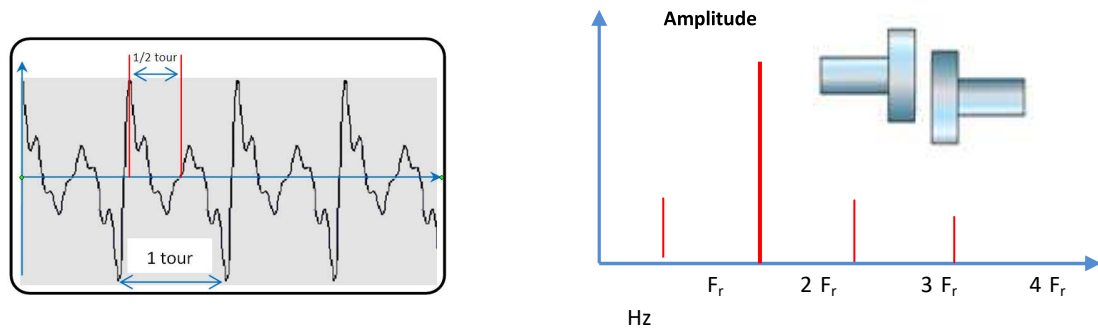


Figure 2.7: Le signal temporel et l'image vibratoire théorique de désalignement parallèle

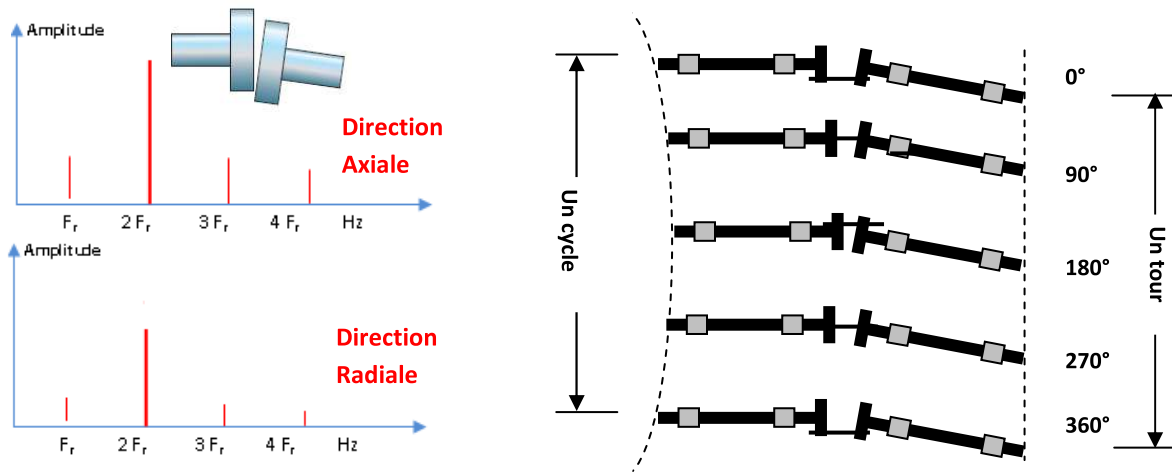


Figure 2.8: Images vibratoires d'un défaut d'alignement angulaire

2.3.2.3 Désalignement des Paliers

Les axes des deux paliers dans une seule unité de machine présentent une non-concentration (voir figure fig. 2.8). Ce phénomène peut résulter d'une installation défectueuse d'un palier, d'un mauvais ajustement des supports de fixation ou encore d'une déformation du châssis due, par exemple, à des variations thermiques, entraînant une déformation de l'arbre du rotor. Cette anomalie se traduit spectralement par un comportement similaire à celui observé lors d'un désalignement angulaire.

2.4 Signatures des anomalies de roulement dans le courant statorique

[69, 2] Les roulements à billes peuvent présenter des défauts dus au montage/assemblage ou indirectement par les efforts mécaniques de l'ensemble moteur et procédé entraîné. La détérioration peut également être induite par des courants lors d'une alimentation utilisant un onduleur de puissance (variation de vitesse). La détection d'un défaut peut être envisagée soit par une

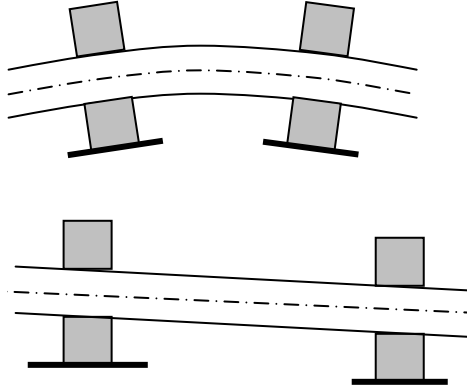


Figure 2.9: Désalignement de paliers se traduisant par une flexion de l'arbre

analyse des vibrations, soit par l'analyse du contenu spectral du courant absorbé, ou encore par la modification du bruit sonore.

D'une part, Schoen et al. [61] montrent que la relation de la vibration de roulement au spectre du courant statorique peut être déterminée en se rappelant que toute excentricité dans l'entrefer produit des défauts dans la densité de flux de celui-ci, et donne par conséquent la relation suivante :

$$f_{ecc} = |f_e \pm k \cdot f_{rm}| \quad (2.9)$$

où :

f_{ecc} : Fréquence caractéristique de l'excentricité.

f_e : Fréquence d'alimentation électrique.

f_{rm} : Fréquence de la machine tournante.

k : Entier représentant les ordres harmoniques.

Dans la situation où les billes du roulement supportant le rotor sont en cause, toute anomalie dans ce roulement entraînera un déplacement radial du rotor par rapport au stator. Ce déplacement entraîne des changements dans la distribution de champ magnétique à l'intérieur de la machine [61]. Ces modifications se traduisent par l'apparition de courants statoriques à différentes fréquences :

$$|f_a \pm k \cdot f_{défaut}| \quad \text{avec } k = 1, 2, 3, \dots \quad (2.10)$$

Avec :

$f_a = 50 \text{ Hz}$: fréquence d'alimentation ou source

$f_{défaut}$: fréquence caractéristique des vibrations induites par les défauts.

D'autre part, Blodt [13] suggère une expression caractéristique pour les trois types de défauts de la manière suivante :

$$\text{Bague extérieure :} \quad f_{bng} = |f_s \pm k \cdot f_0| \quad (2.11)$$

$$\text{Bague intérieure :} \quad f_{bng} = |f_s \pm f_r \pm k \cdot f_i| \quad (2.12)$$

$$\text{Bille :} \quad f_{bng} = |f_s \pm f_{cage} \pm k \cdot f_b| \quad (2.13)$$

En tout cas, les fréquences typiques de vibration théoriques associées aux anomalies sur tous les composants rotatifs des paliers dépendent de leurs dimensions (voir Figure fig. 2.10 [68]). Un problème sur la bille entre en contact simultanément avec la partie externe et la partie interne du roulement à chaque tour complet de la bille autour de son axe. La fréquence liée à ce problème

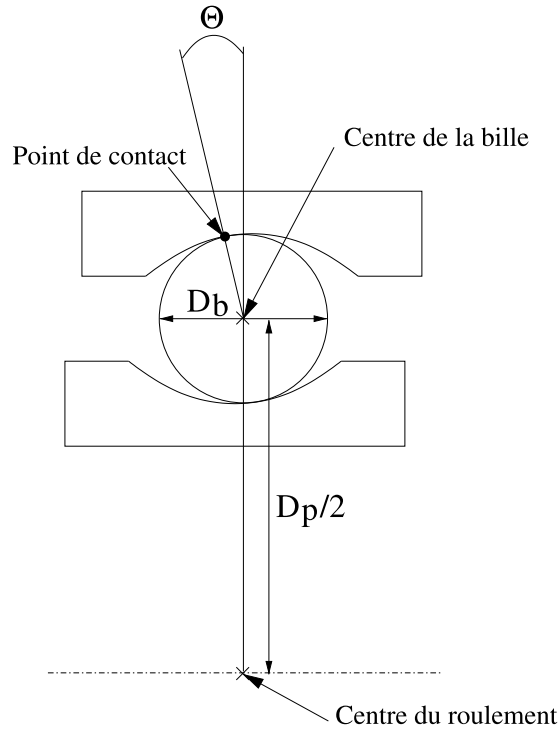


Figure 2.10: Dimension de roulement à billes

est donc [61] :

$$f_{bille} = f_r \frac{D_p}{D_b} \left[1 - \left(\frac{D_b \cos \theta}{D_p} \right)^2 \right] \quad (2.14)$$

Les occurrences de défauts sur les surfaces externe et interne se manifestent à chaque passage d'une bille au-dessus d'un défaut. Ce phénomène se répète N fois lors d'un tour complet de la bague (où N représente le nombre total de billes). Ainsi, les fréquences observées sur les surfaces externe et interne des bagues sont déterminées de cette manière [61] :

$$f_{be} = \frac{f_r}{2} N_b \left(1 - \frac{D_b \cos \theta}{D_p} \right) \quad (2.15)$$

$$f_{bi} = \frac{f_r}{2} N_b \left(1 + \frac{D_b \cos \theta}{D_p} \right) \quad (2.16)$$

$$f_c = \frac{f_r}{2} \left(1 - \frac{D_b \cos \theta}{D_p} \right) \quad (2.17)$$

Où :

- f_{be} : fréquence propre de la bague externe,
- f_{bi} : fréquence propre de la bague interne,
- f_c : fréquence propre de rotation de la cage,
- f_{bille} : fréquence propre de rotation des billes,
- f_r : fréquence de rotation mécanique,
- N_b : nombre de billes,
- D_b : diamètre des billes,
- D_p : diamètre de la cage,

- θ : angle de contact.

On remarquera que cinq paramètres physiques doivent être connus pour déterminer les fréquences caractéristiques de défaut du roulement : D_b diamètre de la bille, D_p diamètre moyen ou diamètre de la cage, N_b nombre de billes, θ angle de contact en radians, et f_r fréquence de rotation.

Les parties défailantes du roulement génèrent une fréquence caractéristique liée à son comportement dynamique, tandis que les vibrations mécaniques émises dépendent des vitesses de rotation individuelles de chaque élément. Toutefois, en l'absence de données sur les dimensions géométriques des roulements intégrés dans la machine asynchrone, il devient difficile d'identifier précisément les fréquences fondamentales associées aux défauts de roulement. De plus, Schoen a démontré dans [61] que les fréquences de vibration peuvent être estimées pour la plupart des roulements ayant entre 6 et 12 billes par les formules suivantes :

$$f_{be} = 0.4 \times N_b \times f_r \quad (2.18)$$

$$f_{bi} = 0.6 \times N_b \times f_r \quad (2.19)$$

$$f_{cage} = 0.4 \times f_r \quad (2.20)$$

Ces règles empiriques (USA), qui sont définies pour permettre d'évaluer grossièrement les fréquences qui apparaissent lors de la défaillance d'un roulement, ne sont qu'approximatives. Néanmoins, il existe des équations [7] qui sont basées sur des essais portant sur plus de 2000 roulements de divers types pour estimer les fréquences f_{cage} , f_i et f_o . Ainsi, pour la série 6300, ses approximations donnent :

$$f_{be} = f_r \times \left(0.5 - \frac{1.2}{N_b} \right) \quad (2.21)$$

$$f_{bi} = f_r \times \left(0.5 + \frac{1.2}{N_b} \right) \quad (2.22)$$

$$f_{bille} = f_r \times \left(0.2 - \frac{1.2}{N_b} \right) \quad (2.23)$$

$$f_{cage} = f_r \times \left(0.5 - \frac{1.2}{N_b} \right) \quad (2.24)$$

Il existe une dispersion entre les fréquences des défauts des roulements calculées avec les formules théoriques et expérimentales développées ci-dessus (fig. 2.11) [7].

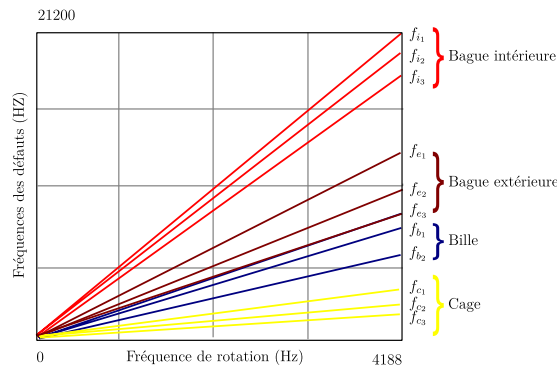


Figure 2.11: Fréquences des défauts d'un roulement (623) : 1- théorique. 2- empirique. 3- moyenne sur 2000 roulements

2.5 Conclusion

Dans ce chapitre, les principales techniques de diagnostic basées sur le traitement du signal ont été présentées et analysées. Les méthodes d'analyse temporelle, fréquentielle et temps-fréquence ont montré leur efficacité pour extraire des caractéristiques représentatives des défauts mécaniques et électriques des machines. Toutefois, l'interprétation de ces caractéristiques et la prise de décision nécessitent souvent des outils plus avancés capables de traiter de grandes quantités de données et d'automatiser le processus de diagnostic. Dans cette perspective, le chapitre suivant sera consacré aux techniques de diagnostic basées sur le *Machine Learning*, qui exploitent les caractéristiques extraites des signaux afin d'améliorer la précision et la fiabilité de la détection et de la classification des défauts.

Techniques de diagnostic basées sur le *Machine Learning*

3.1 Introduction

L'évolution des techniques de surveillance conditionnelle a favorisé l'intégration de l'intelligence artificielle dans le diagnostic des machines électriques. Parmi les approches développées, l'apprentissage automatique (*Machine Learning*) occupe une place importante grâce à sa capacité à exploiter les données de fonctionnement afin d'identifier automatiquement les défauts et d'améliorer la fiabilité du diagnostic. Ce chapitre est consacré à l'étude des principales techniques de diagnostic basées sur l'apprentissage automatique. Après une présentation des concepts fondamentaux de l'intelligence artificielle et des différents paradigmes d'apprentissage, les algorithmes de classification les plus utilisés dans le diagnostic des défauts, notamment les k plus proches voisins (k -NN), les machines à vecteurs de support (SVM) et les arbres de décision (DT), seront décrits.

3.2 Les fondations de l'AI, ML et DL

Avant tout, il est essentiel de comprendre que le Machine Learning (ML) constitue un domaine de l'Intelligence Artificielle (IA) qui exploite des techniques permettant aux ordinateurs d'apprendre à partir de données. Dans cette perspective, le Deep Learning (DL) se positionne comme un sous-domaine du ML, mettant particulièrement l'accent sur l'utilisation de réseaux neuronaux profonds (DNN) pour résoudre des problèmes complexes. Par conséquent, le DL s'inscrit dans le cadre plus large du ML, lequel appartient lui-même à l'ensemble de l'IA. Les différents domaines de l'IA, y compris le ML et le DL, sont illustrés dans la Figure (fig. 3.1)

3.2.1 Les méthodes d'apprentissage

Il existe trois grandes catégories de méthodes d'apprentissage, notamment l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. Chacune de ces approches a des caractéristiques distinctes en termes de données d'entrée, de résultats attendus et d'applications.

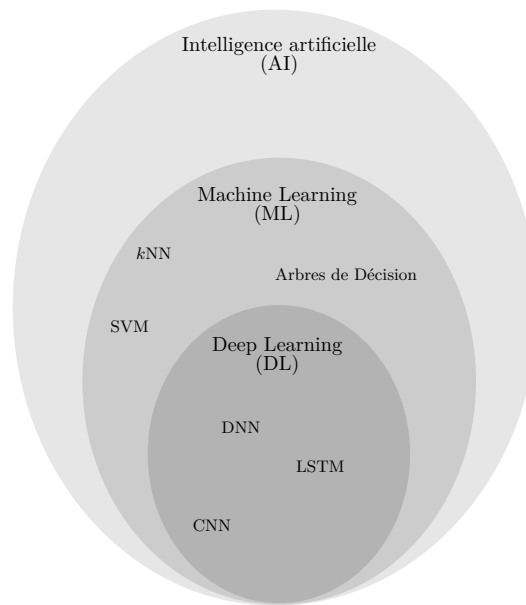


Figure 3.1: Les domaines de l'AI, ML et DL

3.2.1.1 Apprentissage Supervisé (Supervised Learning)

Dans l'apprentissage supervisé, le modèle est entraîné sur un ensemble de données étiquetées, c'est-à-dire des données pour lesquelles les résultats attendus sont connus. Le but est d'apprendre à prédire les étiquettes pour de nouvelles données. Les algorithmes d'apprentissage supervisé incluent les réseaux de neurones, les arbres de décision, les machines à vecteurs de support (SVM), les méthodes ensemblistes comme le Random Forest, et les régressions linéaires et logistiques.

3.2.1.2 Apprentissage Non Supervisé (Unsupervised Learning)

Contrairement à l'apprentissage supervisé, l'apprentissage non supervisé ne nécessite pas d'étiquettes pour les données d'entraînement. Les algorithmes non supervisés trouvent des structures cachées ou des modèles dans les données sans supervision. Les techniques courantes incluent le clustering (comme le k -means clustering) et la réduction de dimensionnalité (comme l'analyse en composantes principales PCA).

3.2.1.3 Apprentissage par Renforcement

L'apprentissage par renforcement implique des agents qui prennent des décisions dans un environnement pour maximiser une récompense cumulative. Les agents apprennent à travers l'interaction avec l'environnement en essayant différentes actions et en observant les récompenses résultantes. Les algorithmes d'apprentissage par renforcement incluent Q-Learning, SARSA, et les méthodes basées sur les réseaux de neurones comme les algorithmes de politique et de valeur.

3.2.2 Les applications de l'apprentissage Supervisé

L'apprentissage Supervisé offre la possibilité de construire des modèles pour aborder deux types de problèmes distincts [59]:

Les problèmes de Régression : Dans les problèmes de régression, l'objectif est de prédire la

valeur d'une variable continue, ce qui signifie qu'elle peut prendre une gamme infinie de valeurs. Exemple :

- Prédire le prix d'un appartement (y) selon sa surface habitable (x).
- Prédire la quantité d'essence consommée (y) selon la distance parcourue (x).

Les problèmes de Classification : Les problèmes de classification impliquent la classification d'un objet dans différentes classes. En d'autres termes, l'objectif est de prédire la valeur d'une variable discrète, qui est limitée à un ensemble fini de valeurs. Exemple :

- Prédire si un email est un spam (*classe* $y = 1$) ou non (*classe* $y = 0$) selon le nombre de liens présent dans l'email (x).
- Prédire si une tumeur est maligne (*classe* $y = 1$) ou bénigne (*classe* $y = 0$) selon la taille de la tumeur (x_1) et l'âge du patient (x_2).

3.2.3 Les notions de base de ML

Les notions de base du machine learning constituent le fondement sur lequel reposent les méthodes et les algorithmes utilisés pour enseigner aux machines à apprendre à partir de données.

3.2.3.1 Ensemble de données (Dataset)

L'ensemble de données (dataset) est une collection d'exemples utilisés pour entraîner, évaluer et tester les modèles d'apprentissage automatique. Il est composé de caractéristiques x (ou attributs) et d'étiquettes y (dans le cas de l'apprentissage supervisé). Les étiquettes y sont les réponses attendues pour chaque exemple x dans l'apprentissage supervisé. L'ensemble de données est réparti de la manière suivante :

Données d'Entraînement (Training Data) Les données d'entraînement sont le jeu de données sur lequel le modèle est formé. C'est à partir de ces données que le modèle apprend à faire des prédictions en ajustant ses paramètres lors du processus d'entraînement. Les données d'entraînement doivent être représentatives du problème que le modèle cherche à résoudre.

Données de Validation (Validation Data) Les données de validation sont utilisées pour évaluer les performances du modèle pendant l'entraînement et pour ajuster les hyperparamètres. Après chaque itération d'entraînement, le modèle est évalué sur les données de validation pour estimer ses performances sur des données qu'il n'a pas vues pendant l'entraînement. Cela permet de détecter le surajustement (overfitting) et de sélectionner les meilleurs hyperparamètres.

Données de Test (Test Data) Les données de test sont utilisées pour évaluer les performances finales du modèle une fois qu'il a été formé et validé. Ces données sont complètement distinctes des données d'entraînement et de validation. Le modèle n'a jamais été exposé à ces données auparavant. Les performances sur les données de test donnent une estimation de la capacité du modèle à généraliser sur de nouvelles données.

3.2.3.2 Validation croisée (Cross Validation)

La validation croisée consiste à diviser l'ensemble de données en plusieurs sous-ensembles, généralement appelés "folds". Ensuite, le modèle est entraîné sur plusieurs combinaisons de ces sous-ensembles et évalué sur les données restantes. La méthode de validation croisée la plus courante est la K -Fold Cross-Validation. Dans cette méthode, les données sont divisées en K sous-ensembles de taille égale. Le modèle est ensuite entraîné K fois en utilisant $K - 1$

sous-ensembles comme données d'entraînement et le sous-ensemble restant comme données de validation. Cette opération est répétée K fois, chaque sous-ensemble étant utilisé une fois comme jeu de validation. Les performances du modèle sont finalement moyennées sur les K itérations pour obtenir une estimation plus fiable de ses performances.

3.2.3.3 Surapprentissage et sous-apprentissage

Le surapprentissage (Overfitting) se produit lorsque le modèle apprend les détails spécifiques aux données d'entraînement au lieu de généraliser des modèles sous-jacents.

Le sous-apprentissage (Underfitting) se produit lorsque le modèle est trop simple pour capturer la structure des données.

3.2.3.4 Matrice de confusion

La matrice de confusion permet de visualiser les performances d'un modèle de classification en fournissant un aperçu clair du nombre de prédictions correctes et incorrectes pour chaque classe. Elle est utile pour évaluer la précision, le rappel, la spécificité et d'autres mesures de performance du modèle.

3.2.3.5 Normalisation (Normalization)

La normalisation, en Machine Learning, fait référence au processus de mise à l'échelle des caractéristiques (features) x dans un ensemble de données (Dataset) pour qu'elles aient une plage de valeurs commune et standardisée. Cela permet d'éviter que certaines caractéristiques ne dominent pas les autres de par leur échelle numérique, ce qui pourrait fausser le processus d'apprentissage du modèle. La méthode la plus couramment utilisée pour la normalisation des données est la méthode Z-score. Cette méthode transforme les données pour qu'elles aient une moyenne nulle et un écart type unitaire. Cela rend la distribution des données centrée autour de zéro avec une variance de 1. La formule de standardisation est la suivante :

$$X_{\text{std}} = \frac{X - \mu}{\sigma} \quad (3.1)$$

où X est la valeur initiale, μ est la moyenne de la variable, et σ est l'écart type de la variable.

3.2.3.6 hyperparamètres

Les hyperparamètres sont des paramètres qui ne sont pas appris directement par le modèle lors de l'entraînement, mais qui affectent le comportement et les performances du modèle. Contrairement aux paramètres du modèle, qui sont appris à partir des données d'entraînement, les hyperparamètres doivent être définis avant le processus d'entraînement et peuvent être ajustés pour obtenir de meilleures performances du modèle. Dans le contexte de l'apprentissage automatique, notamment lors de l'entraînement de modèles de réseaux neuronaux, plusieurs hyperparamètres sont utilisés, parmi lesquels :

Itération (Iteration) Une itération représente une seule mise à jour des paramètres du modèle pendant l'entraînement. Pendant une itération, un lot (batch) de données est passé à travers le modèle, les prédictions sont calculées, la perte est évaluée, et les paramètres du modèle sont ajustés à l'aide d'un algorithme d'optimisation (comme la descente de gradient). Après chaque itération, les poids du modèle sont mis à jour pour minimiser la fonction de perte par rapport aux données d'entraînement.

Époque (Epoch) Une époque représente une passe complète à travers l'ensemble des données d'entraînement. Pendant une époque, le modèle voit et traite toutes les données d'entraînement une fois. Une époque est composée de plusieurs itérations, le nombre exact d'itérations dépend de la taille du lot (batch size) et du nombre total d'échantillons d'entraînement. Après chaque époque, le modèle peut être évalué sur un ensemble de validation pour surveiller sa performance et détecter tout surajustement (overfitting).

Taille du Lot (Batch Size) La taille du lot fait référence au nombre d'échantillons d'entraînement inclus dans chaque itération. Les données d'entraînement sont divisées en petits lots, et chaque lot est utilisé pour calculer les gradients et mettre à jour les poids du modèle. Le choix de la taille du lot peut avoir un impact sur la vitesse d'apprentissage, la stabilité de la convergence et l'utilisation des ressources matérielles. Des tailles de lot plus grandes peuvent accélérer l'apprentissage mais peuvent nécessiter plus de mémoire, tandis que des tailles de lot plus petites peuvent offrir une convergence plus stable mais plus lente.

Par exemple, si l'ensemble d'entraînement contient 1000 échantillons et que le batch size est de 100, alors une époque (epoch) nécessitera 10 itérations pour être complétée, voir figure (fig. 3.2).

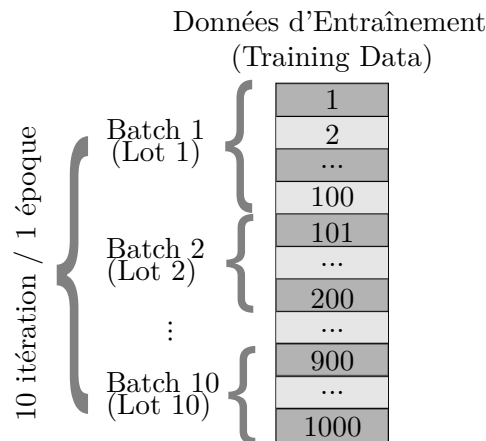


Figure 3.2: Exemple

3.2.3.7 Précision Globale (Accuracy)

La précision globale (*Accuracy*) est une métrique utilisée pour évaluer les performances générales d'un modèle de classification. Elle représente le rapport entre le nombre de prédictions correctes et le nombre total d'échantillons analysés.

Elle est définie par l'équation suivante :

$$\text{Accuracy} = \frac{\text{Nombre de prédictions correctes}}{\text{Nombre total d'échantillons}}$$

Une valeur élevée de l'accuracy indique que le modèle effectue correctement la majorité des prédictions.

3.2.3.8 Précision (Precision)

La précision (*Precision*) mesure la capacité du modèle à fournir des prédictions positives correctes parmi toutes les prédictions positives réalisées. Elle permet d'évaluer la fiabilité des résultats positifs obtenus par le modèle.

Elle est calculée par la formule suivante :

$$\text{Precision} = \frac{\text{Prédictions positives correctes}}{\text{Total des prédictions positives}}$$

Une valeur élevée de précision signifie que le modèle génère peu de fausses alertes.

3.2.3.9 Rappel (Recall)

Le rappel (*Recall*) mesure la capacité du modèle à détecter correctement les cas positifs présents dans le jeu de données. Cette métrique est importante lorsque l'objectif est de minimiser les cas non détectés.

Il est donné par la relation suivante :

$$\text{Recall} = \frac{\text{Prédictions positives correctes}}{\text{Nombre total de cas positifs réels}}$$

Un rappel élevé indique que le modèle est capable d'identifier la majorité des cas positifs.

3.2.3.10 Score F1 (F1-Score)

Le score F1 (*F1-Score*) combine la précision et le rappel afin de fournir une mesure globale équilibrée des performances du modèle. Il correspond à la moyenne harmonique entre ces deux métriques.

Le score F1 est exprimé par :

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Une valeur élevée du score F1 indique un bon équilibre entre la précision et le rappel du modèle.

3.3 Arbres de Décision

L'arbre de décision ou Decision Tree (DT) en anglais, est une structure arborescente similaire à un diagramme de flux. Il est utilisé dans les domaines de l'apprentissage supervisé pour la classification et la régression. Chaque nœud dans l'arbre représente une fonction de décision, et chaque branche représente le résultat de cette fonction [54].

Dans le contexte de la classification, les arbres de décision sont utilisés pour catégoriser les éléments dans différentes classes. En revanche, en régression, ils sont utilisés pour prédire une valeur continue [19].

3.3.1 Principe

L'idée est de diviser l'ensemble de données en sous-ensembles plus petits et plus simples en utilisant des décisions basées sur les caractéristiques des données. Une telle partition est illustrée sur la figure (fig. 3.3) pour des variables réelles [9]. Les arbres de décision peuvent être utilisés pour des tâches de classification et de régression.

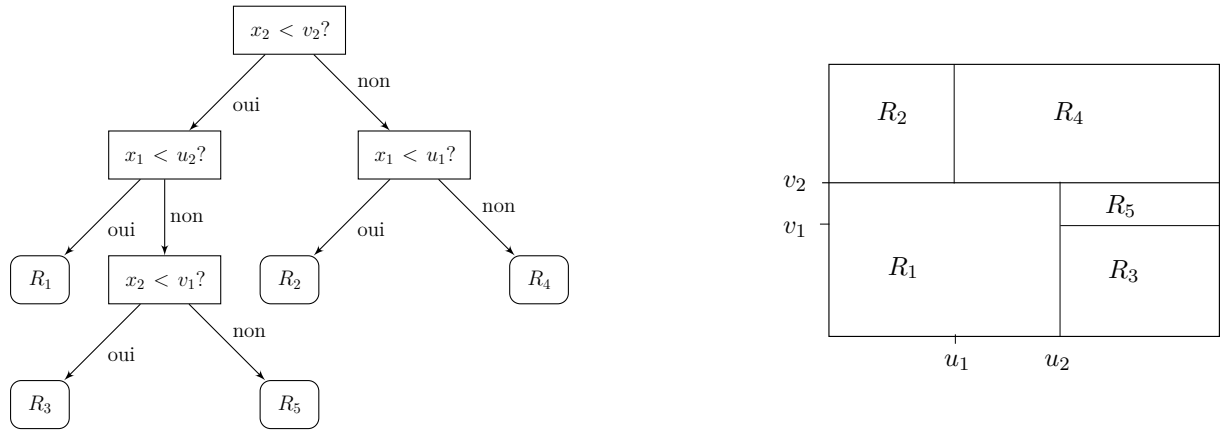


Figure 3.3: L'arbre de décision (à gauche) partitionne en 5 zones (à droite)

3.3.1.1 Sélection de l'attribut

L'arbre de décision commence par le nœud racine, où l'attribut qui donne la meilleure division (c'est-à-dire la plus "pure") est choisi. Plusieurs mesures, telles que le gain d'information, le ratio de gain ou l'indice Gini, peuvent être utilisées pour évaluer la "pureté" d'une division.

3.3.1.2 Division

Une fois l'attribut sélectionné, l'ensemble de données est divisé en sous-ensembles en fonction des valeurs de cet attribut. Cette opération est répétée pour chaque sous-ensemble jusqu'à ce que :

- Tous les éléments du sous-ensemble appartiennent à la même classe (dans le cas de la classification) ;
- Aucun gain d'information supplémentaire ne peut être obtenu ;
- Ou d'autres critères d'arrêt sont atteints (comme une profondeur maximale de l'arbre).

3.3.1.3 Feuille de décision

Une fois la division terminée, chaque feuille de l'arbre représente une décision (c'est-à-dire une classe dans le cas de la classification ou une valeur continue dans le cas de la régression).

3.3.2 Critères de Division

Pour choisir la meilleure division, plusieurs mesures de qualité peuvent être utilisées, notamment :

- Entropie $H(X)$

$$H(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i) \quad (3.2)$$

- Gain d'information IG

$$IG(S, A) = H(S) - \sum_{t \in T} \frac{|S_t|}{|S|} H(S_t) \quad (3.3)$$

3.3.3 Algorithmes de Construction

- **ID3** et **C4.5**: utilisent le gain d'information et le ratio de gain respectivement pour les arbres de classification [55].
- **CART**: utilise l'indice Gini pour la classification et la MSE pour la régression [18].

3.3.3.1 Algorithme ID3 (Iterative Dichotomiser 3)

L'algorithme ID3 est un algorithme classique pour la construction d'arbres de décision, développé par Ross Quinlan en 1986 [54]. L'algorithme peut être décrit comme (algorithme 1).

Algorithm 1 ID3

```
1: Initialisation: Prendre l'ensemble complet  $S$  des données comme racine.
2: while des sous-ensembles non classifiés existent do
3:   Calcul des entropies:
4:     Calculer l'entropie  $\text{Entropie}(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$ 
5:   Sélection de l'attribut:
6:     Choisir l'attribut  $a$  avec le gain d'information maximal:
7:      $\text{Gain}(S, a) = \text{Entropie}(S) - \sum_{v \in \text{Valeurs}(a)} \frac{|S_v|}{|S|} \text{Entropie}(S_v)$ 
8:   Création d'un nœud:
9:     Créer un nouveau nœud de décision pour l'attribut  $a$ 
10:  Décomposition de l'ensemble:
11:    Diviser l'ensemble  $S$  en sous-ensembles  $S_v$  en fonction des valeurs  $v$  de l'attribut  $a$ 
12:  Récursion:
13:    Répéter les étapes 2-7 pour chaque sous-ensemble  $S_v$ 
14:  Terminaison:
15:    Si tous les cas dans  $S$  ou  $S_v$  sont de la même classe, créer une feuille.
16: end while
```

3.3.3.2 L'algorithme C4.5

L'algorithme C4.5 est une extension de l'algorithme ID3 (Iterative Dichotomiser 3) développée par Ross Quinlan [54]. Au lieu du gain d'information utilisé par l'algorithme ID3, C4.5 utilise le ratio de gain. Cela permet de corriger le biais en faveur des attributs avec de nombreuses valeurs distinctes. L'algorithme peut être décrit comme (algorithme 2).

3.3.3.3 L'algorithme CART

Contrairement à ID3 et C4.5, CART utilise un seuil t en plus de l'attribut a pour diviser le dataset. Le seuil et l'attribut sont choisis de manière à minimiser une fonction de coût basée sur une mesure d'impureté. L'ensemble S est divisé en deux sous-ensembles S_{left} et S_{right} en fonction de a et t . L'algorithme peut être décrit comme (algorithme 3).

Remarque : Une feuille est créée si un critère d'arrêt est atteint. Ce critère pourrait être une profondeur maximale de l'arbre, un nombre minimum d'éléments pour diviser un nœud, etc.

3.3.4 Élagage de l'Arbre

L'élagage d'un arbre de décision consiste à retirer des sous-arbres ou des nœuds de l'arbre initial dans le but de le simplifier, de réduire sa taille, et d'éviter le surapprentissage. Cette opération permet souvent d'améliorer les performances de généralisation de l'arbre sur des données inconnues. Il existe plusieurs méthodes d'élagage, et voici quelques-unes des plus courantes :

3.3.4.1 Élagage réduit d'erreur (Reduced Error Pruning)

Cette méthode consiste à supprimer un nœud N et à le remplacer par une de ses feuilles L avec la classe majoritaire. Si la performance P ne se détériore pas ou s'améliore sur un ensemble de

Algorithm 2 C4.5

Initialisation: Prendre l'ensemble complet S des données comme racine.

2: **while** des sous-ensembles non classifiés existent **do**

Calcul des entropies:

4: Calculer l'entropie $\text{Entropie}(S) = -\sum_{i=1}^c p_i \log_2(p_i)$

Sélection de l'attribut:

6: Choisir l'attribut a avec le gain ratio maximal:
 $\text{GainRatio}(S, a) = \frac{\text{Gain}(S, a)}{\text{SplitInformation}(S, a)}$

8: $\text{Gain}(S, a) = \text{Entropie}(S) - \sum_{v \in \text{Valeurs}(a)} \frac{|S_v|}{|S|} \text{Entropie}(S_v)$
 $\text{SplitInformation}(S, a) = -\sum_{v \in \text{Valeurs}(a)} \frac{|S_v|}{|S|} \log_2 \left(\frac{|S_v|}{|S|} \right)$

10: **Création d'un nœud:**
 Créer un nouveau nœud de décision pour l'attribut a

12: **Décomposition de l'ensemble:**
 Diviser l'ensemble S en sous-ensembles S_v en fonction des valeurs v de l'attribut a

14: **Récursion:**
 Répéter les étapes 2-8 pour chaque sous-ensemble S_v

16: **Terminaison:**
 Si tous les cas dans S ou S_v sont de la même classe, créer une feuille.

18: **end while**

validation V , le changement est conservé.

$$P_{\text{après}} \geq P_{\text{avant}} \quad (3.4)$$

3.3.4.2 Élagage de coût-complexité (Cost Complexity Pruning)

Il s'agit d'une méthode d'élagage régularisée où l'on ajoute un terme de pénalité α à la fonction de coût J . Le terme de pénalité est proportionnel à la taille de l'arbre $|T|$. La formule est :

$$J_\alpha(T) = J(T) + \alpha|T| \quad (3.5)$$

où $J(T)$ est le taux d'erreur de l'arbre T , $|T|$ est la taille de l'arbre, et α est un paramètre de complexité.

3.3.4.3 Élagage par validation croisée (Cross-Validation Pruning)

Cette méthode utilise la validation croisée pour évaluer les performances des sous-arbres. On calcule le taux d'erreur moyen $\bar{E}$ sur k ensembles de validation et on choisit le niveau d'élagage qui minimise ce taux :

$$\bar{E} = \frac{1}{k} \sum_{i=1}^k E_i \quad (3.6)$$

où E_i est le taux d'erreur sur le i -ème ensemble de validation.

3.3.5 Exemple

Objectif: Créez un arbre de décision pour diagnostiquer différents types de défauts dans une machine électrique ([table 3.1](#)).

Algorithm 3 CART

Initialisation: Prendre l'ensemble complet S des données comme racine.
while des sous-ensembles non classifiés existent **do**
3: **Sélection de l'attribut et seuil:**
 Choisir l'attribut a et le seuil t qui minimisent la fonction de coût:
 $\text{Cost}(S, a, t) = w_{\text{left}} \text{Impurity}(S_{\text{left}}) + w_{\text{right}} \text{Impurity}(S_{\text{right}})$
6: $w_{\text{left}} = \frac{|S_{\text{left}}|}{|S|}$, $w_{\text{right}} = \frac{|S_{\text{right}}|}{|S|}$
 Création d'un nœud:
 Créer un nouveau nœud de décision pour l'attribut a et le seuil t
9: **Décomposition de l'ensemble:**
 Diviser S en S_{left} et S_{right} en fonction de a et t
 Récursion:
12: Répéter les étapes 2-7 pour chaque sous-ensemble S_{left} et S_{right}
 Terminaison:
 Si un critère d'arrêt est atteint pour S , S_{left} ou S_{right} , créer une feuille.
15: **end while**

Table 3.1: Données pour le diagnostic de machines électriques

Température (°C)	Vibration (mm/s)	Courant (A)	État du Moteur
40	0.5	10	Normal
55	1.2	15	Surchauffe
37	0.4	20	Déséquilibre
50	1.1	12	Usure mécanique
45	0.6	9	Normal
60	1.4	18	Surchauffe
42	0.7	11	Normal
52	1.0	14	Usure mécanique

Solution

Pour construire un arbre de décision, il faut choisir quel attribut utiliser à chaque nœud de l'arbre. Cette décision est généralement prise en utilisant un critère comme le gain d'information, qui est basé sur le concept d'entropie.

Calcul de l'Entropie L'entropie est une mesure de l'incertitude dans un ensemble de données. La formule générale pour calculer l'entropie E pour un ensemble S est:

$$E(S) = - \sum_{i=1}^n p_i \log_2(p_i)$$

L'ensemble de données contient 8 échantillons, dont 3 sont "Normal", 2 sont "Usure mécanique", 2 sont "Surchauffe", et 1 est "Déséquilibre". Ainsi, l'entropie $E(S)$ est:

$$E(S) = - \left(\frac{3}{8} \log_2 \frac{3}{8} + \frac{2}{8} \log_2 \frac{2}{8} + \frac{2}{8} \log_2 \frac{2}{8} + \frac{1}{8} \log_2 \frac{1}{8} \right) \approx 1.906$$

L'étape suivante consiste à calculer l'entropie pour les différents sous-ensembles basés sur chaque attribut. Pour l'attribut "Température", on a deux intervalles: 35 – 45°C et 50 – 60°C.

Interval de Température 35 – 45°C Dans cet intervalle, nous avons 5 échantillons : 3 "Normal"

et 2 "Déséquilibre". L'entropie pour cet intervalle est:

$$E(S)_{35-45^\circ\text{C}} = -\left(\frac{3}{4}\log_2\frac{3}{4} + \frac{1}{4}\log_2\frac{1}{4}\right) = 0.811$$

Interval de Température 50 – 60°C Dans cet intervalle, nous avons 3 échantillons : 2 "Usure mécanique" et 1 "Surchauffe". L'entropie pour cet intervalle est:

$$E(S)_{50-60^\circ\text{C}} = -\left(\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}\right) = 1.0$$

Calcul du Gain d'Information Le gain d'information (IG) est défini comme:

$$IG(S, A) = E(S) - \sum_t \left(\frac{|S_t|}{|S|} \times E(S_t)\right)$$

Pour l'attribut "Température", le gain d'information (IG) peut être calculé en utilisant la formule suivante:

$$IG(S, \text{Température}) = E(S) - \left[\frac{|S_{35-45}|}{|S|} \times E(S_{35-45}) + \frac{|S_{50-60}|}{|S|} \times E(S_{50-60})\right]$$

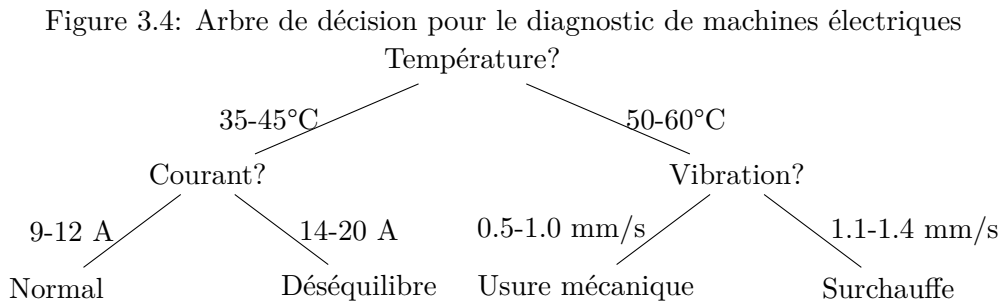
Dans notre exemple, $E(S) = 1.906$, $|S| = 8$, $E(S_{35-45}) = 0.811$, $|S_{35-45}| = 4$, $E(S_{50-60}) = 1.0$, et $|S_{50-60}| = 4$.

En utilisant ces valeurs dans la formule, nous avons :

$$\begin{aligned} IG(S, \text{Température}) &= 1.906 - \left(\frac{4}{8}(0.811) + \frac{4}{8}(1.0)\right) \\ &= 1.005 \end{aligned}$$

Donc, le gain d'information pour l'attribut "Température" est environ 1.005.

l'Arbre de Décision Maintenant que nous avons calculé le gain d'information pour l'attribut "Température", nous pouvons utiliser cette information pour construire l'arbre de décision. Dans cet exemple, l'attribut "Température" a le gain d'information le plus élevé, ce qui en fait le meilleur choix pour le nœud racine de l'arbre (fig. 3.4).



Explications:

1. Le nœud racine de l'arbre est "Température" car il a le gain d'information le plus élevé.
2. Le premier niveau de l'arbre est divisé en deux branches: une pour la plage de température

35 – 45°C et une autre pour la plage de température 50 – 60°C.

3. Pour la branche 35 – 45°C, nous utilisons l'attribut "Courant" pour séparer davantage les données. Dans cette branche, les valeurs de courant de 9-12 A sont associées à un état "Normal", et les valeurs de 14-20 A à un état "Déséquilibre".

4. De même, pour la branche 50 – 60°C, nous utilisons l'attribut "Vibration" pour faire une autre séparation. Les valeurs de vibration de 0.5-1.0 mm/s sont associées à un état "Usure mécanique", et les valeurs de 1.1-1.4 mm/s à un état "Surchauffe".

En suivant cet arbre de décision, on peut facilement diagnostiquer l'état d'un moteur électrique en fonction des valeurs mesurées de température, de courant et de vibration.

3.4 k -NN (k -Nearest Neighbors)

Le K -Nearest Neighbors (k -NN) est un algorithme d'apprentissage supervisé non para-métrique utilisé pour les tâches de classification et de régression. Il fonctionne en calculant la distance entre un nouveau point de données et un ensemble de données d'entraînement existant. Ensuite, les K points d'entraînement les plus proches du nouveau point de données sont utilisés pour prédire la classe ou la valeur du nouveau point de données. La figure (fig. 3.5) illustre le principe de la méthode k -NN.

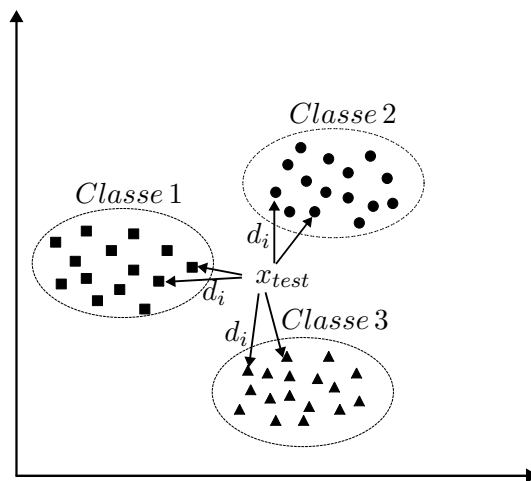


Figure 3.5: Calcul des distances d_i d'un échantillon à classer x_{test}

3.4.1 Principe

La méthode k -NN fonctionne sur un principe simple : la "majorité vote". Pour un nouvel échantillon, l'algorithme trouve les k échantillons les plus proches dans l'ensemble de données d'entraînement et attribue la classe majoritaire (ou la valeur moyenne dans le cas de la régression) parmi ces k voisins. Dans le cas de la classification, la classe du nouveau point de données est prédite en fonction de la classe des K voisins les plus proches. Il existe deux approches principales pour la classification k -NN :

Vote à la Majorité

La classe du nouveau point de données est la classe la plus fréquente parmi les k voisins. L'équation mathématique pour le vote à la majorité est la suivante [9] :

$$f(x) = \arg \max_c |\{x_i \in N_k(x) : y_i = c\}| \quad (3.7)$$

Où :

$f(x)$ est la classe prédite pour le nouveau point de données x

$N_k(x)$ est l'ensemble des k voisins les plus proches du nouveau point de données x

y_i est la classe du point d'entraînement x_i

c est une classe

Probabilité

La classe du nouveau point de données est prédite en fonction de la probabilité de chaque classe parmi les K voisins. L'équation mathématique pour la probabilité est la suivante [48] :

$$p(c|x) = \frac{|\{x_i \in N_k(x) : y_i = c\}|}{|N_k(x)|} \quad (3.8)$$

Où :

$p(c|x)$ est la probabilité de la classe c pour le nouveau point de données x

Dans le cas de la régression, la valeur du nouveau point de données est prédite en fonction des valeurs des K voisins les plus proches. L'équation mathématique pour la régression k -NN est la suivante [9] :

$$y(x) = \frac{\sum_{x_i \in N_k(x)} y_i}{|N_k(x)|} \quad (3.9)$$

Où :

$y(x)$ est la valeur prédite du nouveau point de données x

y_i est la valeur du point d'entraînement x_i

$N_k(x)$ est l'ensemble des K voisins les plus proches de x

3.4.2 Choix du Nombre de Voisins k

Le nombre de voisins K est un paramètre important dans l'algorithme k -NN. Il détermine le nombre de points d'entraînement à utiliser pour prédire la classe ou la valeur du nouveau point de données. Un nombre trop faible de voisins peut conduire à un surajustement (overfitting), où l'algorithme mémorise les données d'entraînement et ne peut pas généraliser à de nouvelles données. Un nombre trop élevé de voisins peut entraîner un sous-ajustement (underfitting), où l'algorithme ne parvient pas à capturer les subtilités des données. Des méthodes comme la validation croisée (Cross-validation) sont souvent utilisées pour choisir une valeur optimale de k [33].

3.4.3 Mesure de la Distance

L'ingrédient essentiel de l'algorithme des k plus proches voisins est la métrique de distance qui détermine quelles observations de l'ensemble d'entraînement sont les plus proches du point à étiqueter. Il existe plusieurs façons de calculer la distance entre deux points de données. Les métriques de distance les plus courantes comprennent :

- **Distance Euclidienne :** Il s'agit de la métrique de distance la plus simple à calculer. Elle est définie comme la racine carrée de la somme des carrés des différences entre les coordonnées des deux points. La distance euclidienne d est calculée comme suit :

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3.10)$$

- **Distance Manhattan :** Cette distance est définie comme la somme des différences absolues entre les coordonnées des deux points. La distance de Manhattan d est calculée comme suit :

$$d = \sum_{i=1}^n |x_i - y_i| \quad (3.11)$$

- **Distance de Minkowski :** Cette distance est une généralisation des distances euclidienne et de Manhattan. La formule de la distance de Minkowski avec le paramètre q est donnée par :

$$d = \left(\sum_{i=1}^n |x_i - y_i|^q \right)^{1/q} \quad (3.12)$$

où q est le paramètre qui détermine la norme (pour la distance euclidienne, $q = 2$, et pour la distance de Manhattan, $q = 1$), et $\sum$ représente la somme des différences absolues élevées à la puissance de q .

- **Distance de Mahalanobis :** Cette distance prend en compte la covariance entre les variables des deux points de données. Elle est définie comme la racine carrée de la différence entre les moyennes des deux points et la distance euclidienne entre eux. La distance de Mahalanobis pour un point x_i à partir d'une distribution avec un vecteur moyen μ et une matrice de covariance Σ est donnée par :

$$D(x_i) = \sqrt{(x_i - \mu)^T \Sigma^{-1} (x_i - \mu)} \quad (3.13)$$

Où :

x_i est le vecteur représentant le point

μ est le vecteur moyen de la distribution

Σ est la matrice de covariance de la distribution

- **Distance de Hamming :** Cette distance est spécifiquement conçue pour des données catégorielles, telles que des chaînes de caractères. Elle mesure le nombre d'endroits où les symboles correspondants diffèrent entre deux chaînes de même longueur. La distance de Hamming d_{Hamming} est calculée comme la proportion d'endroits où les symboles diffèrent :

$$d = \frac{1}{n} \sum_{i=1}^n I(x_i \neq y_i) \quad (3.14)$$

où n est la longueur des chaînes, et $I(x_i \neq y_i)$ est une fonction indicatrice qui vaut 1 si x_i est différent de y_i , et 0 sinon.

Chacune de ces mesures a ses propres avantages et inconvénients, et le choix dépend souvent de la nature du problème.

3.4.4 Apprentissage Paresseux (Lazy Learning)

L'apprentissage paresseux (Lazy Learning), est utilisé pour décrire l'algorithme des k plus proches voisins. Dans ce cas, le processus d'apprentissage consiste uniquement à stocker les données d'entraînement et ne nécessite aucun calcul. Cependant, il est important de noter que cela peut être limitant en termes de mémoire si l'ensemble de données d'entraînement est très volumineux. Inversement, le processus de prédiction nécessite le calcul de la distance entre l'observation à étiqueter et chaque observation dans l'ensemble de données d'entraînement, ce qui peut être intensif en termes de calculs lorsque l'ensemble de données d'entraînement est important. En fait, une prédiction nécessite le calcul de n distances, une opération avec une complexité de $O(np)$ dans p dimensions, puis de trouver les k plus petites de ces distances, une opération avec une complexité de $O(n \log k)$ [9].

3.4.5 Sensibilité au Bruit

Les k plus proches voisins (k -NN) sont sensibles au bruit dans les données d'entraînement, ce qui peut conduire à une classification incorrecte des nouveaux points de données. La sensibilité au bruit peut être réduite en utilisant des techniques de prétraitement des données telles que la normalisation et la suppression des valeurs aberrantes.

3.4.6 Algorithme K plus proches voisins

L'algorithme des K plus proches voisins est décrit dans (algorithm 4).

Algorithm 4 k -NN (K plus proches voisins)

```

1: Entrées:
    $\mathcal{D}$ : ensemble de données d'apprentissage avec des étiquettes
    $\mathbf{x}_{\text{test}}$ : exemple à classifier
    $K$ : nombre de voisins à considérer
2: Sortie:
    $\hat{y}$ : étiquette prédite pour  $\mathbf{x}_{\text{test}}$ 
3: Calculer la distance entre  $\mathbf{x}_{\text{test}}$  et chaque exemple dans  $\mathcal{D}$ 
4: for chaque exemple  $i$  dans  $\mathcal{D}$  do
5:   Calculer la distance euclidienne  $d_i$  entre  $\mathbf{x}_{\text{test}}$  et l'exemple  $i$ 
6:    $d_i \leftarrow \sqrt{\sum_{j=1}^n (\mathbf{x}_{\text{test}}[j] - \mathbf{x}_i[j])^2}$ 
7: end for
8: Trier les exemples dans  $\mathcal{D}$  par ordre croissant de distance à  $\mathbf{x}_{\text{test}}$ 
9: Sélectionner les  $K$  exemples les plus proches
10: Calculer le vote majoritaire  $f(\mathbf{x}_{\text{test}})$  parmi les  $K$  exemples les plus proches
11:  $\hat{y} \leftarrow \arg \max_c |\{x_i \in N_k(\mathbf{x}_{\text{test}}) : y_i = c\}|$  ▷ Détermination de l'étiquette prédite

```

3.4.7 Exemple :

Supposons que nous ayons un ensemble d'entraînement de 3 observations étiquetées comme suit

$$\begin{array}{ll}
 x_1 = (1, 2), & y_1 = 0 \\
 x_2 = (3, 4), & y_2 = 1 \\
 x_3 = (5, 6), & y_3 = 1
 \end{array}$$

Et nous voulons classer la nouvelle observation $x = (4, 5)$.

Solution 1 : (Vote à la Majorité)

Nous choisissons $k = 2$. Tout d'abord, nous calculons les distances euclidiennes entre x et chaque échantillon d'entraînement :

Distance entre x et x_1 :

$$\text{Distance} = \sqrt{(4 - 1)^2 + (5 - 2)^2} = \sqrt{9 + 9} = \sqrt{18}$$

Distance entre x et x_2 :

$$\text{Distance} = \sqrt{(4 - 3)^2 + (5 - 4)^2} = \sqrt{1 + 1} = \sqrt{2}$$

Distance entre x et x_3 :

$$\text{Distance} = \sqrt{(4 - 5)^2 + (5 - 6)^2} = \sqrt{1 + 1} = \sqrt{2}$$

Ensuite, nous identifions les deux voisins les plus proches de x parmi x_1, x_2, x_3 en utilisant les distances calculées. Les deux voisins les plus proches sont x_2 et x_3 car ils ont les distances les plus petites. Les classes associées à x_2 et x_3 sont respectivement $y_2 = 1$ et $y_3 = 1$.

Enfin, en utilisant le vote à la majorité des k -plus proches voisins ($k = 2$), nous pouvons conclure que la classe prédite pour x est la classe qui apparaît le plus fréquemment parmi ses voisins les plus proches. Dans ce cas, les deux voisins appartiennent tous deux à la classe 1, donc la classe prédite pour x est également 1.

Solution 2 : (Probabilité)

Nous avons déjà identifié les deux voisins les plus proches de x parmi x_1, x_2, x_3 dans la première solution.

Ensuite, nous calculons la probabilité conditionnelle que x appartienne à chaque classe c en utilisant la formule fournie précédemment dans la section 3.4.1 :

Pour la classe $c = 0$:

$$p(c = 0|x) = \frac{|x_i \in N_2(x) : y_i = 0|}{|N_2(x)|}$$

Parmi les voisins les plus proches de x (x_2 et x_3), aucun n'appartient à la classe 0. Ainsi, le numérateur est 0. Le dénominateur est 2 car il y a 2 voisins les plus proches. Par conséquent,

$$p(c = 0|x) = \frac{0}{2} = 0$$

Pour la classe $c = 1$:

$$p(c = 1|x) = \frac{|x_i \in N_2(x) : y_i = 1|}{|N_2(x)|}$$

Parmi les voisins les plus proches de x (x_2 et x_3), les deux appartiennent à la classe 1. Ainsi, le numérateur est 2. Le dénominateur est 2 car il y a 2 voisins les plus proches. Par conséquent,

$$p(c = 1|x) = \frac{2}{2} = 1$$

Enfin, en utilisant la probabilité conditionnelle, nous pouvons conclure que la classe prédite pour $x = (4, 5)$ est la classe c pour laquelle la probabilité conditionnelle est maximale. Dans ce cas, la probabilité conditionnelle est maximale pour $c = 1$.

3.5 Machine à Vecteurs de Support (SVM)

Le principe de la machine à vecteurs de support (SVM) repose sur la recherche d'un hyperplan optimal qui peut séparer efficacement les données de deux classes différentes. L'idée est de trouver un hyperplan qui maximise la marge entre les points de données des deux classes. La marge est définie comme la distance entre l'hyperplan et les points les plus proches de chaque classe.

3.5.1 Le cas Linéairement séparable

[21] Considérons un ensemble de données d'entraînement X avec des étiquettes de classe Y . Chaque donnée est représentée par une paire (x_i, y_i) , où x_i est le vecteur de caractéristiques et y_i est l'étiquette de classe (1 ou -1).

L'objectif est de trouver un hyperplan de séparation optimale défini par :

$$w^T \cdot x - b = 0 \quad (3.15)$$

Le vecteur w est le poids de l'hyperplan, b est le terme de biais, et la marge (M) entre les deux classes est :

$$M = \frac{2}{\|w\|} \quad (3.16)$$

L'optimisation de l'algorithme SVM revient à trouver le vecteur de poids w et le terme de biais b qui maximisent la marge tout en respectant les contraintes imposées par les données d'entraînement, comme le montre la figure (fig. 3.6). Ainsi, la fonction objectif traduit l'équilibre entre maximiser la marge et minimiser les erreurs de classification.

3.5.1.1 La fonction de Lagrange

la fonction objectif typique à maximiser est la fonction lagrangienne définie comme suite :

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i [y_i(w \cdot x_i - b) - 1] \quad (3.17)$$

où :

α_i est le vecteur des multiplicateurs de Lagrange associés aux contraintes du problème,

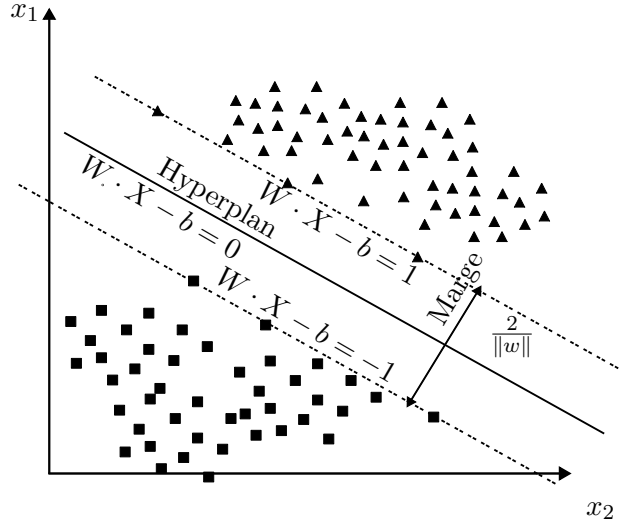


Figure 3.6: Le cas Linéairement séparable

Les deux parties de cette fonction objectif ont des rôles distincts. La première partie, $\frac{1}{2}\|w\|^2$, encourage la maximisation de la marge entre les classes. C'est la partie géométrique de la fonction objectif.

La deuxième partie, $\sum_{i=1}^N \alpha_i [y_i(w \cdot x_i - b) - 1]$, introduit les multiplicateurs de Lagrange et impose que chaque donnée d'entraînement soit classée correctement tout en respectant la marge. L'introduction des multiplicateurs de Lagrange (α_i) simplifie le problème et permet de trouver les valeurs optimales de w et b qui définissent l'hyperplan optimal. Les multiplicateurs de Lagrange (α_i) sont obtenus en résolvant le problème d'optimisation quadratique dual :

$$\mathcal{W}(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(x_i, x_j), \quad (3.18)$$

sous les contraintes $0 \leq \alpha_i \leq C$ et $\sum_{i=1}^N \alpha_i y_i = 0$, où C est le paramètre de régularisation. avec

$$K(x_i, x_j) = x_i^T x_j \quad (3.19)$$

Ici, x_i et x_j sont les vecteurs de caractéristiques des points i et j , respectivement, et T représente la transposition. Le produit scalaire $x_i^T x_j$ mesure la similarité ou la projection des caractéristiques des deux points dans l'espace d'origine.

3.5.1.2 Formules de w et b

les formules de w et b sont obtenues en minimisant la fonction de Lagrange $L(w, b, \alpha)$ par rapport à w et b en prenant les dérivées partielles par rapport à w et b en les égalant à zéro :

$$\frac{\partial \mathcal{L}}{\partial w} = w - \sum_{i=1}^N \alpha_i y_i x_i = 0 \quad (3.20)$$

$$\frac{\partial \mathcal{L}}{\partial b} = - \sum_{i=1}^N \alpha_i y_i = 0 \quad (3.21)$$

À partir des équations 3.20 et 3.21, nous pouvons obtenir les expressions pour calculer w et b :

$$w = \sum_{i=1}^N \alpha_i y_i x_i \quad (3.22)$$

$$b = y_k - \sum_{i=1}^N \alpha_i y_i x_i^T x_k \quad \text{pour tout } 0 < \alpha_k < C \quad (3.23)$$

3.5.1.3 Prédiction

La prédiction pour de nouvelles données x_{new} s'effectue en utilisant :

$$f(x_{\text{new}}) = w^T \cdot x_{\text{new}} - b \quad (3.24)$$

Si $f(x_{\text{new}}) > 0$, la donnée est classée comme 1, sinon elle est classée comme -1 .

3.5.2 Le cas non Linéairement séparable

[21] Dans de nombreux cas réels, les données ne sont pas linéairement séparables dans l'espace d'origine, d'où la nécessité de les transformer vers un espace redéfini où une séparation devient possible, comme illustré dans la figure fig. 3.7. L'*astuce du noyau* (kernel trick) consiste à projeter implicitement les données dans un espace de dimension supérieure, où une séparation linéaire devient possible, en utilisant une fonction noyau $K(\mathbf{x}, \mathbf{x}')$ pour calculer les produits scalaires sans expliciter la transformation. Plusieurs types de noyaux peuvent être utilisés, tels que le noyau polynomial et le noyau radial de base (RBF), chacun adapté à des types spécifiques de données.

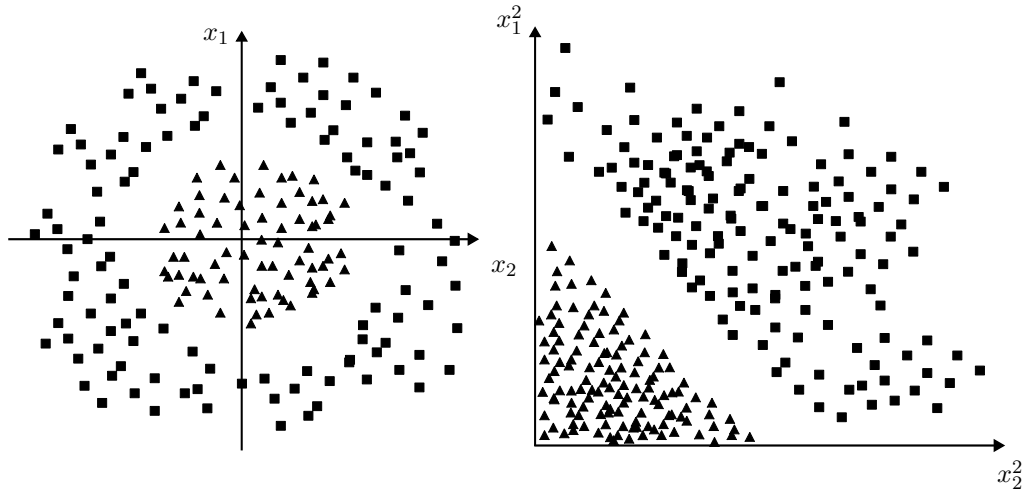


Figure 3.7: Transformation des données dans un espace de redescription.

3.5.2.1 Noyau Polynomial

Le noyau polynomial est une extension du noyau linéaire dans le contexte des Support Vector Machines (SVM). Il offre une flexibilité accrue pour modéliser des relations non linéaires entre les données. La forme générale du noyau polynomial est définie comme suit :

$$K(x_i, x_j) = (x_i^T x_j + c)^d \quad (3.25)$$

où x_i et x_j sont les vecteurs de caractéristiques des points i et j , respectivement. Les paramètres c et d contrôlent la forme et la complexité de la transformation polynomiale.

3.5.2.2 Noyau RBF (Radial Basis Function)

Le noyau RBF, ou Radial Basis Function, est un type de noyau largement utilisé dans les Support Vector Machines (SVM) pour traiter des relations complexes et non linéaires entre les données. La forme générale du noyau RBF est définie comme suit :

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (3.26)$$

où x_i et x_j sont les vecteurs de caractéristiques des points i et j , respectivement, et σ est le paramètre de largeur du noyau.

3.5.2.3 La fonction de Lagrange

La fonction de Lagrange pour les données non linéairement séparables devient :

$$L = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^N \alpha_i \quad (3.27)$$

3.5.2.4 Formules de w et b

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \phi(\mathbf{x}_i) \quad (3.28)$$

où $\phi(\cdot)$ la fonction de transformation vers un espace de dimension supérieure.

Remarque : le vecteur $\mathbf{w}$ n'est pas utilisé explicitement, car les calculs se font uniquement via les produits scalaires $K(\mathbf{x}_i, \mathbf{x}_j)$, évitant ainsi l'évaluation directe de la transformation $\phi(\cdot)$ souvent de dimension infinie.

$$b = y_k - \sum_{i=1}^N \alpha_i y_i K(x_i, x_k) \quad \text{pour tout } 0 < \alpha_k < C \quad (3.29)$$

Il est important de noter que ces formules sont spécifiques à celles déjà vues dans la section 3.5.1.2, sauf que là, les formules incluront des termes liés à la fonction de noyau $K(x_i, x_j)$.

3.5.2.5 Prédiction

La prédiction pour de nouvelles données (x_{new}) s'effectue comme dans le cas linéaire, mais en utilisant la fonction de décision non linéaire

$$f(x_{\text{new}}) = \sum_{i=1}^N \alpha_i y_i K(x_i, x_{\text{new}}) - b \quad (3.30)$$

3.5.3 Exemple

Supposons que nous ayons un ensemble d'entraînement de 3 observations étiquetées comme suit

$$\begin{array}{ll} x_1 = (1, 2), & y_1 = -1 \\ x_2 = (2, 1), & y_2 = -1 \\ x_3 = (4, 4), & y_3 = 1 \end{array}$$

Et nous voulons classer les nouvelles observations $x_4 = (5, 4)$ et $x_5 = (1, 1)$.

Solution

Tout d'abord, nous devons calculer w et b afin de pouvoir tracer l'hyperplan :

$$w^T \cdot x - b = 0$$

Avec :

$$w = \sum_i \alpha_i y_i x_i \text{ et } \sum_i \alpha_i y_i = 0$$

En remplaçant w dans l'équation de l'hyperplan, on obtient :

$$\begin{aligned} \sum_i \alpha_i y_i x_i^T x_j - b &= 1 \quad (\text{Pour le cas où } x_i \text{ appartient à la classe "1"}) \\ \sum_i \alpha_i y_i x_i^T x_j - b &= -1 \quad (\text{Pour le cas où } x_i \text{ appartient à la classe "-1"}) \end{aligned}$$

Pour calculer b et α , nous devons résoudre le système d'équations suivant, en notant que $\beta_i = \alpha_i y_i$

$$\begin{cases} \beta_1 x_1^T \cdot x_1 + \beta_2 x_2^T \cdot x_1 + \beta_3 x_3^T \cdot x_1 - b = -1 \\ \beta_1 x_1^T \cdot x_2 + \beta_2 x_2^T \cdot x_2 + \beta_3 x_3^T \cdot x_2 - b = -1 \\ \beta_1 x_1^T \cdot x_3 + \beta_2 x_2^T \cdot x_3 + \beta_3 x_3^T \cdot x_3 - b = 1 \\ \beta_1 + \beta_2 + \beta_3 = 0 \end{cases}$$

En remplaçant x_1 , x_2 et x_3 par leurs valeurs, on obtient :

$$\begin{cases} 5\beta_1 + 4\beta_2 + 12\beta_3 - b = -1 \\ 4\beta_1 + 5\beta_2 + 12\beta_3 - b = -1 \\ 12\beta_1 + 12\beta_2 + 32\beta_3 - b = 1 \\ \beta_1 + \beta_2 + \beta_3 = 0 \end{cases} \Rightarrow \begin{cases} \beta_1 = -0.08 \\ \beta_2 = -0.08 \\ \beta_3 = 0.16 \\ b = 2.2 \end{cases}$$

Maintenant, calculons w :

$$w = \sum_i \alpha_i y_i x_i = \sum_i \beta_i x_i = -0.08 \begin{pmatrix} 1 \\ 2 \end{pmatrix} - 0.08 \begin{pmatrix} 2 \\ 1 \end{pmatrix} + 0.16 \begin{pmatrix} 4 \\ 4 \end{pmatrix} = \begin{pmatrix} 0.4 \\ 0.4 \end{pmatrix}$$

Pour pouvoir classifier x_4 et x_5 , on utilise l'équation de l'hyperplan :

$$\begin{aligned} f(x_{\text{new}}) &= w^T \cdot x_{\text{new}} - b \\ f(x_4) &= \begin{pmatrix} 0.4 & 0.4 \end{pmatrix} \begin{pmatrix} 5 \\ 4 \end{pmatrix} - 2.2 = 1.4 \rightarrow f(x_4) > 0 \\ &\Rightarrow \text{donc } x_4 \text{ appartient à la classe "1"} \\ f(x_5) &= \begin{pmatrix} 0.4 & 0.4 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - 2.2 = -1.4 \rightarrow f(x_5) < 0 \\ &\Rightarrow \text{donc } x_5 \text{ appartient à la classe "-1"} \end{aligned}$$

3.6 Conclusion

Dans ce chapitre, les fondements de l'apprentissage automatique appliqué au diagnostic des machines électriques ont été présentés. Les principaux paradigmes d'apprentissage ainsi que les algorithmes de classification les plus couramment utilisés, tels que les k plus proches voisins (k -NN), les machines à vecteurs de support (SVM) et les arbres de décision (DT), ont été décrits. Cette étude a permis de mettre en évidence les principes et les caractéristiques de ces approches, qui constituent aujourd'hui des outils largement utilisés pour le développement de systèmes de diagnostic intelligents. Dans la continuité de ces travaux, le chapitre suivant sera consacré aux techniques de diagnostic basées sur l'apprentissage profond (*Deep Learning*), en présentant les concepts fondamentaux ainsi que les architectures DNN, CNN et LSTM, largement exploitées pour l'analyse et la classification des défauts des machines électriques.

Techniques de diagnostic basées sur le *Deep Learning*

4.1 Introduction

L'apprentissage profond (*Deep Learning*) représente une évolution majeure des techniques d'intelligence artificielle appliquées au diagnostic des machines électriques. Grâce à sa capacité à apprendre automatiquement des représentations complexes à partir des données brutes, il permet de réduire la dépendance aux étapes d'extraction manuelle des caractéristiques et d'améliorer les performances des systèmes de diagnostic intelligents. Ce chapitre est consacré à l'étude des principales méthodes de diagnostic basées sur l'apprentissage profond. Après une présentation des concepts fondamentaux de cette approche, les architectures les plus utilisées dans le domaine du diagnostic des défauts seront décrites. Une attention particulière sera portée aux réseaux de neurones profonds (*Deep Neural Networks* - DNN), aux réseaux de neurones convolutifs (*Convolutional Neural Networks* - CNN) et aux réseaux à mémoire longue à court terme (*Long Short-Term Memory* - LSTM), en mettant en évidence leurs principes de fonctionnement, leurs architectures et leur intérêt pour l'analyse et la classification des défauts des machines électriques.

4.2 Réseaux de neurones profonds (DNN)

Les réseaux de neurones profonds (DNN) sont une classe puissante de modèles dans le domaine du machine learning, inspirés par la structure du cerveau humain. Ces réseaux sont capables d'apprendre des représentations complexes et hiérarchiques des données, les rendant particulièrement adaptés à un large éventail de tâches, de la classification d'images à la prédiction de séquences temporelles. Un réseau de neurones profond (DNN) est un type de réseau de neurones artificiels qui comprend plusieurs couches de neurones, souvent appelées couches cachées, entre la couche d'entrée et la couche de sortie. La profondeur du réseau se réfère au nombre de ces couches cachées.

4.2.1 Architectures du réseau de neurones

4.2.1.1 Perceptron

L'architecture d'un perceptron est relativement simple, car il s'agit d'une unité de neurone fondamentale. L'architecture d'un perceptron est représentée dans la figure (fig. 4.1). Cette

unité de neurone fondamentale prend un ensemble d'entrées pondérées $(x_1, x_2, \dots, x_d)$, réalise une somme pondérée, et applique ensuite une fonction d'activation pour produire une sortie. La formule mathématique du perceptron peut être exprimée comme suit [44]:

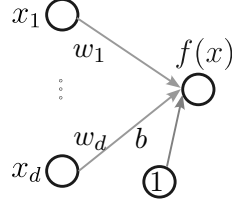


Figure 4.1: Architecture du perceptron

$$a(x) = b + \sum_i w_i x_i = b + W^T X \quad (4.1)$$

$$f(x) = g(a(x)) = g\left(b + \sum_i w_i x_i\right) \quad (4.2)$$

Où:

- $\mathbf{a}(x)$: Vecteur de pré-activation du réseau de neurones pour l'entrée $\mathbf{x}$.
- b est le terme de biais.
- w_i sont les poids associés aux caractéristiques d'entrée x_i .
- $\mathbf{W}$ est le vecteur des poids.
- $\mathbf{X}$ est le vecteur des caractéristiques d'entrée.
- $g(\cdot)$ est la fonction d'activation de la sortie du perceptron.

4.2.1.2 Réseau de neurone à une seule couche cachée

L'architecture d'un réseau de neurones à une seule couche cachée est une configuration classique qui comporte une couche d'entrée, une couche cachée et une couche de sortie voir la figure (fig. 4.2). La formule mathématique d'un réseau de neurones à une seule couche cachée peut être exprimée comme suit [44]:

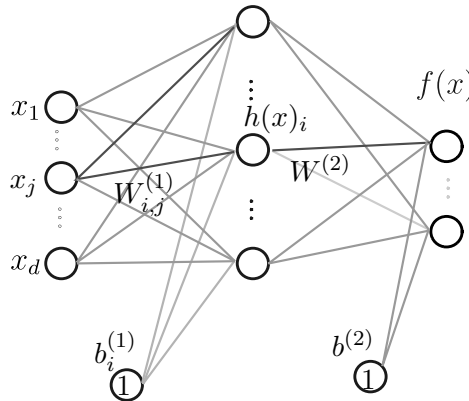


Figure 4.2: Réseau de neurone à une seule couche cachée

$$\begin{aligned} a(x) &= b^{(1)} + W^{(1)} X \\ (a(x))_i &= b_i^{(1)} + \sum_j W_{i,j}^{(1)} x_j \end{aligned} \quad (4.3)$$

Où:

- $\mathbf{a}(x)$: Vecteur vecteur de pré-activation du réseau de neurones pour l'entrée $\mathbf{x}$.
- $\mathbf{b}^{(1)}$: Terme de biais de la couche cachée.
- $\mathbf{W}^{(1)}$: Matrice de poids associée aux connexions entre la couche d'entrée et la couche cachée.
- $\mathbf{X}$: Vecteur d'entrée.
- $\mathbf{a}(x)_i$: Composant i -ème du vecteur de sortie.

Cette équation calcule le i -ème composant de la sortie en prenant le terme de biais ($\mathbf{b}_i^{(1)}$) pour le i -ème neurone de la couche cachée et en ajoutant la somme pondérée des valeurs d'entrée $\mathbf{x}_j$ avec les poids correspondants ($\mathbf{W}_{i,j}^{(1)}$).

$$\begin{aligned} h(x) &= g(a(x)) \\ f(x) &= o(b^{(2)} + W^{(2)}h(x)) \end{aligned} \quad (4.4)$$

Où:

- $h(x)$: Vecteur d'activation de la couche cachée.
- $g(\cdot)$: Fonction d'activation appliquée à chaque composant de $\mathbf{a}(x)$.
- $f(x)$: Vecteur de sortie final du réseau de neurones.
- $o(\cdot)$: Fonction d'activation appliquée à la sortie du réseau de neurone.
- $\mathbf{b}^{(2)}$: Terme de biais de la couche de sortie.
- $\mathbf{W}^{(2)}$: Matrice de poids associée aux connexions entre la couche cachée et la couche de sortie.

4.2.1.3 Réseau de neurone Multicouches

Un réseau de neurones multicouches (MLP, pour l'anglais Multilayer Perceptron) est un type courant de réseau de neurones artificiels. Il se compose de plusieurs couches de neurones, généralement organisées en trois parties : une couche d'entrée, une ou plusieurs couches cachées, et une couche de sortie. La figure (fig. 4.3) illustre un exemple d'un réseau de neurones avec deux couches cachées. Les équations suivantes décrivent le fonctionnement d'un réseau de neurones avec L couches cachées, où $L + 1$ représente la couche de sortie [44]:

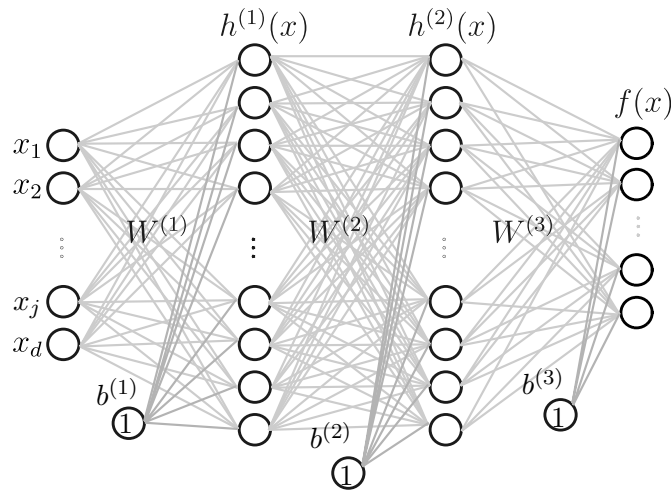


Figure 4.3: Réseau de neurone à deux couches cachées

$$\begin{aligned} & \left(h^{(0)}(X) = X \right) \\ & a^{(k)}(x) = b^{(k)} + W^{(k)}h^{(k-1)}(X) \quad \text{avec } k > 0 \end{aligned} \tag{4.5}$$

$$\begin{aligned} & h^{(k)}(x) = g(a^{(k)}(x)) \quad \text{avec } k \in \{1, \dots, L\} \\ & h^{(L+1)}(x) = o(a^{(L+1)}(x)) = f(x) \quad \text{avec } k = L + 1 \end{aligned}$$

Initialisation de la couche d'entrée ($h^{(0)}(X) = X$):

$h^{(0)}(X)$ représente l'activation de la couche d'entrée, qui est simplement égale au vecteur d'entrée X . Cela correspond à la couche d'entrée ne faisant aucune transformation sur les données d'entrée.

Propagation avant à travers les couches cachées ($a^{(k)}(x) = b^{(k)} + W^{(k)}h^{(k-1)}(X)$):

$a^{(k)}(x)$ est le vecteur de pré-activation de la couche cachée k .

$b^{(k)}$ est le terme de biais pour la couche cachée k .

$W^{(k)}$ est la matrice de poids associée aux connexions entre la couche cachée k et la couche cachée $k - 1$.

$h^{(k-1)}(X)$ est le vecteur d'activation de la couche cachée précédente ($k - 1$).

Cette équation décrit le calcul de l'activation de la couche cachée k à partir de l'activation de la couche cachée précédente $k - 1$.

Application de la fonction d'activation ($h^{(k)}(x) = g(a^{(k)}(x))$):

$h^{(k)}(x)$ est l'activation de la couche cachée k après l'application de la fonction d'activation $g(\cdot)$ à $a^{(k)}(x)$.

$g(\cdot)$ est la fonction d'activation appliquée à chaque composant de $a^{(k)}(x)$. Cette fonction introduit la non-linéarité dans le réseau de neurones.

Couche de sortie ($h^{(L+1)}(x) = o(a^{(L+1)}(x)) = f(x)$):

$h^{(L+1)}(x)$ est l'activation de la couche de sortie ($L + 1$) après l'application de la fonction d'activation $o(\cdot)$ à $a^{(L+1)}(x)$.

$o(\cdot)$ est la fonction d'activation appliquée à la somme pondérée des entrées de la couche de sortie.

$f(x)$ représente la sortie finale du réseau de neurones.

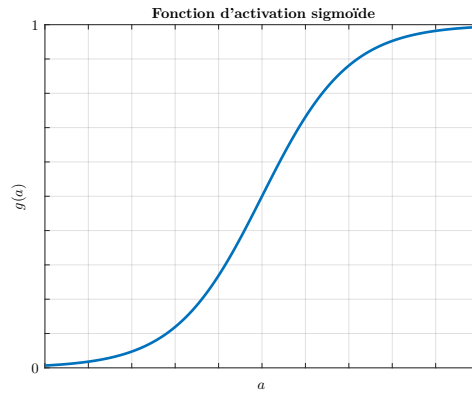
4.2.2 Les Fonctions d'activation

Les fonctions d'activation sont des composants essentiels des neurones artificiels dans un réseau de neurones. Elles introduisent de la non-linéarité dans le modèle, permettant au réseau de capturer des relations complexes dans les données. Voici les fonctions d'activation les plus utilisées dans les réseaux de neurones [44]:

4.2.2.1 Sigmoid (Sigmoid)

La fonction sigmoïde est définie comme :

$$g(a) = \frac{1}{1 + e^{-a}} \tag{4.6}$$



Elle est souvent utilisée dans les couches de sortie pour les problèmes de classification binaire, où la sortie doit être dans l'intervalle $[0, 1]$.

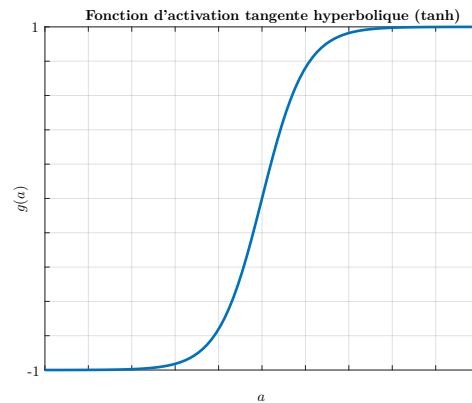
Exemple :

$$g(2) \approx \frac{1}{1 + e^{-2}} \approx 0.88$$

4.2.2.2 Tangente Hyperbolique (Tanh)

La fonction tangente hyperbolique est définie comme :

$$g(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \quad (4.7)$$



Elle produit une sortie dans l'intervalle $[-1, 1]$, et est souvent utilisée dans les couches cachées des réseaux de neurones.

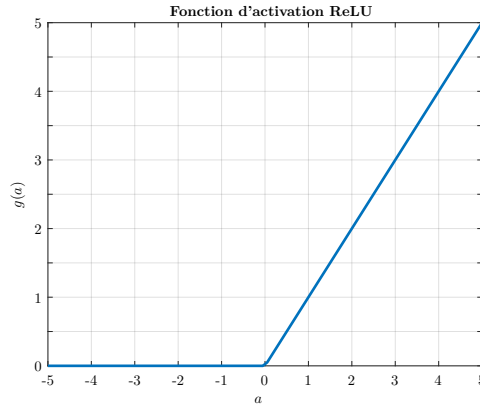
Exemple :

$$g(1) \approx \frac{e - e^{-1}}{e + e^{-1}} \approx 0.76$$

4.2.2.3 Rectified Linear Unit (ReLU)

La fonction ReLU est définie comme :

$$g(a) = \max(0, a) \quad (4.8)$$



Elle est largement utilisée dans les couches cachées, introduisant de la non-linéarité en supprimant les valeurs négatives.

Exemple :

$$g(-1) = \max(0, -1) = 0$$

4.2.2.4 Softmax

La fonction Softmax est définie comme :

$$g(a) = \text{softmax}(a) = \left[\frac{e^{(a_1)}}{\sum_c e^{(a_c)}} \dots \frac{e^{(a_C)}}{\sum_c e^{(a_c)}} \right]^T \quad (4.9)$$

Où:

$a = (a_1, a_2, \dots, a_C)$ est le vecteur pré-activation de la couche de sortie avant l'application de la fonction Softmax. où C est le nombre de classes.

$\left[\frac{e^{(a_1)}}{\sum_c e^{(a_c)}} \dots \frac{e^{(a_C)}}{\sum_c e^{(a_c)}} \right]^T$ crée un vecteur de probabilités normalisé qui représente la distribution de probabilités sur les différentes classes, où c varie de 1 à C

Elle est principalement utilisée dans les couches de sortie pour les problèmes de classification multiclass, normalisant les sorties en probabilités. L'utilisation de la fonction Softmax garantit que les valeurs de sortie sont dans l'intervalle $[0, 1]$ et que leur somme est égale à 1, ce qui les rend interprétables comme des probabilités.

Exemple :

$$g([1, 2, 3]) = \left[\frac{e^1}{e^1 + e^2 + e^3}, \frac{e^2}{e^1 + e^2 + e^3}, \frac{e^3}{e^1 + e^2 + e^3} \right]$$

4.2.3 L'apprentissage du réseau de neurone

L'apprentissage d'un réseau de neurones multicouche implique l'ajustement des poids W et des biais b du réseau afin de minimiser la fonction de coût $J(W, b; X, Y)$. Ici, X représente l'ensemble d'entrée, Y est l'ensemble des étiquettes (labels), et J mesure la disparité entre les prédictions du réseau $f(x)$ et les étiquettes réelles y . Le processus d'optimisation utilise la rétropropagation, une méthode itérative qui propage l'erreur de sortie vers l'arrière du réseau pour ajuster les paramètres. La descente de gradient est ensuite employée pour minimiser la fonction de coût en mettant à jour les poids dans la direction opposée du gradient. En ajustant ces paramètres itérativement, le réseau apprend à mieux représenter les relations complexes dans les données

d'entraînement, améliorant ainsi ses performances et à faire des prédictions précises sur de nouvelles entrées.

4.2.3.1 Fonction de Coût

La fonction de coût mesure la différence entre les prédictions du réseau et les valeurs réelles. Une fonction de coût couramment utilisée est la moyenne de l'erreur quadratique (MSE) pour les problèmes de régression et la perte logistique (Log Loss) pour les problèmes de classification.

Moyenne de l'Erreur Quadratique (MSE)

La MSE est définie comme suit :

$$J(W, b; X, Y) = \frac{1}{2m} \sum_{i=1}^m \|f(x^{(i)}) - y^{(i)}\|^2 \quad (4.10)$$

Où :

- W et b représentent l'ensemble des poids et des biais du réseau.
- X est l'ensemble d'entrée.
- Y est l'ensemble de sortie.
- m est le nombre d'exemples dans l'ensemble d'entraînement.

La fonction de coût MSE mesure l'erreur quadratique moyenne entre les prédictions $f(x^{(i)})$ et les valeurs réelles $y^{(i)}$ pour chaque exemple d'entraînement. Ensuite, cette erreur est moyennée sur l'ensemble de l'entraînement, d'où le terme "Mean Squared Error".

La mise en œuvre de cette fonction de coût dans un contexte de rétropropagation et d'optimisation consiste à minimiser cette quantité. La rétropropagation calcule les gradients par rapport aux poids et aux biais, et la descente de gradient (ou une de ses variantes) ajuste les paramètres pour minimiser la fonction de coût.

L'utilisation de $\frac{1}{2}$ dans la formule simplifie les calculs de dérivées partielles, mais n'affecte pas substantiellement le processus d'optimisation car l'objectif est de trouver les paramètres qui minimisent la fonction de coût.

Perte Logistique (Log Loss)

La fonction de coût Log Loss (ou entropie croisée) mesure l'écart entre les probabilités prédites par le modèle et les classes réelles. Couramment utilisée dans les problèmes de classification, est définie comme suit :

$$J(W, b; X, Y) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log f(x^{(i)}) + (1 - y^{(i)}) \log(1 - f(x^{(i)}))] \quad (4.11)$$

La fonction Log Loss est basée sur le principe de maximisation de la vraisemblance, et la transformation en un problème de minimisation facilite les calculs lors de l'application de la rétropropagation pour ajuster les paramètres W et b .

Démonstration :

Utilisons le modèle logistique pour modéliser la probabilité que Y soit égal à 1 sachant X $P(Y = 1|X)$. La fonction sigmoïde est utilisée pour compresser la sortie dans l'intervalle $(0, 1)$:

$$f(x) = \frac{1}{1 + e^{-(Wx+b)}}$$

Pour l'ensemble d'entraînement $x^{(i)}$ est $f(x^{(i)})$ la probabilité prédite que $y = 1$ est $P(Y = 1|X) = f(x)$, et que $y = 0$ est $P(Y = 0|X) = 1 - f(x)$.

La vraisemblance de l'ensemble d'entraînement Y sachant X est donnée par le produit des probabilités prédites pour chaque exemple d'entraînement est:

$$L(W, b; X, Y) = \prod_{i=1}^m \left(f(x^{(i)})^{y^{(i)}} \cdot (1 - f(x^{(i)}))^{(1-y^{(i)})} \right)$$

Par la suite on prend le logarithme de la vraisemblance pour faciliter les calculs (log-vraisemblance)

$$\log(L(W, b; X, Y)) = \log\left(\prod_{i=1}^m \left(f(x^{(i)})^{y^{(i)}} \cdot (1 - f(x^{(i)}))^{(1-y^{(i)})} \right)\right)$$

$$\log L(W, b; X, Y) = \sum_{i=1}^m \left[y^{(i)} \log f(x^{(i)}) + (1 - y^{(i)}) \log(1 - f(x^{(i)})) \right]$$

Enfin on normalise la fonction par le nombre d'exemples d'entraînement m pour obtenir la fonction de coût :

$$J(W, b; X, Y) = -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} \log f(x^{(i)}) + (1 - y^{(i)}) \log(1 - f(x^{(i)})) \right]$$

L'introduction du signe négatif va nous permettre de minimiser la fonction au lieu de la maximiser.

4.2.3.2 Rétropropagation

La rétropropagation est un processus fondamental dans l'apprentissage des réseaux de neurones multicouches. Elle consiste à propager l'erreur calculée à la sortie du réseau en sens inverse, vers les couches internes. Ce mécanisme permet au réseau d'ajuster les poids et les biais de manière à minimiser la fonction de coût. Des méthodes d'optimisation telles que la descente de gradient stochastique (SGD) et l'algorithme Adam sont utilisées pour guider cette minimisation de manière efficace.

La descente de gradient stochastique (SGD) est une méthode populaire qui optimise les poids du réseau en se basant sur des mini-lots aléatoires d'échantillons d'entraînement. Cette approche présente une efficacité accrue sur de grands ensembles de données, car elle permet des mises à jour fréquentes des poids. Cependant, elle est sensible au choix du taux d'apprentissage, qui doit être ajusté avec soin.

L'algorithme Adam, quant à lui, offre une adaptation des taux d'apprentissage individuels pour chaque paramètre, ce qui le rend moins dépendant d'un choix précis de taux d'apprentissage global. Il s'adapte dynamiquement à la topologie de la fonction de coût et est souvent plus efficace dans des environnements où les données sont bruitées.

Descente de Gradient Stochastique (SGD)

La mise à jour des poids w_{ij} et des biais b_i se fait à partir du gradient. La mise à jour des paramètres se fait pour chaque exemple d'entraînement de chaque itération t :

$$w_{ij} \leftarrow w_{ij} - \alpha \frac{\partial J}{\partial w_{ij}} \quad (4.12)$$

$$b_j \leftarrow b_j - \alpha \frac{\partial J}{\partial b_j} \quad (4.13)$$

où $J(y, f(x))$ est la fonction coût, avec y est l'étiquette (label) et $f(X)$ est la sortie prédite, α est le taux d'apprentissage.

Les gradients de la perte par rapport aux paramètres $\frac{\partial J}{\partial w_{ij}}$ et $\frac{\partial J}{\partial b_i}$ sont calculé en utilisant la règle de dérivée en chaîne :

$$\frac{\partial J}{\partial w_{ij}} = \frac{\partial J}{\partial z_j} \cdot \frac{\partial z_j}{\partial w_{ij}} \quad (4.14)$$

$$\frac{\partial J}{\partial b_j} = \frac{\partial J}{\partial z_j} \cdot \frac{\partial z_j}{\partial b_j} \quad (4.15)$$

où $a_i = g(z_i)$ est la sortie du neurone i après l'application de la fonction d'activation sur la somme pondéré du neurone z_i , $g(\cdot)$ représentant la fonction d'activation.

Par contre, un calcul naïf de toutes ces dérivées serait très inefficace, pour remédier a ce problème on utilise le procédure de la rétropropagation des gradients (où erreurs).

La rétropropagation des gradients utilise le fait que la dérivée pour un neurone à la couche l peut être calculée à partir de la dérivée des neurones connectés à la couche $l + 1$ [37]:

$$\begin{aligned} \frac{\partial J}{\partial z_j} &= \frac{\partial J}{\partial a_j} \cdot \frac{\partial a_j}{\partial z_j} \\ &= \left(\sum_k \frac{\partial J}{\partial z_k} \cdot \frac{\partial z_k}{\partial a_j} \right) \frac{\partial g(z_j)}{\partial z_j} \\ &= \left(\sum_k \frac{\partial J}{\partial z_k} w_{j,k} \right) g(z_j)(1 - g(z_j)) \\ \frac{\partial z_j}{\partial w_{i,j}} &= a_i \\ \frac{\partial z_j}{\partial b_j} &= 1 \end{aligned}$$

où $z_k = b_k + \sum_i w_{i,k} a_i$, et $g(\cdot) = \text{Logistic}(\cdot)$ la fonction d'activation sigmoïde.

En remplaçant les gradients calculés, la forme de mise à jour des poids $w_{i,j}$ et des biais b_j devient [37]:

$$w_{ij} \leftarrow w_{ij} - \alpha a_i g(z_j)(1 - g(z_j)) \sum_k \frac{\partial J}{\partial z_k} w_{j,k} \quad (4.16)$$

$$b_j \leftarrow b_j - \alpha g(z_j)(1 - g(z_j)) \sum_k \frac{\partial J}{\partial z_k} w_{j,k} \quad (4.17)$$

La formulation des équations la plus couramment utilisée est la suivante :

$$w_{i,j} \leftarrow w_{i,j} + \alpha a_i \Delta[j] \quad (4.18)$$

$$b_j \leftarrow b_j + \alpha \Delta[j] \quad (4.19)$$

Où :

$$\Delta[j] = g(z_j)(1 - g(z_j)) \sum_k w_{j,k} \Delta[k]$$

avec $\Delta[j] = -\frac{\partial J}{\partial z_j}$ et $\Delta[k] = -\frac{\partial J}{\partial z_k}$
 Pour la couche de sortie :

$$\Delta[j] = a_j - y_j \quad (4.20)$$

Démonstration

Sachant que :

$$\begin{aligned} a_i &= g(z_i) \quad , \quad \text{où } g(z) = \frac{1}{1 + e^{-z}} \\ J &= -\frac{1}{m} \sum_{i=1}^m [y_i \log(a_i) + (1 - y_i) \log(1 - a_i)] \\ \Delta[i] &= -\frac{\partial J}{\partial z_i} \end{aligned}$$

Il suffit de calculer le gradient $\frac{\partial J}{\partial z_i}$, pour cela on utilise le principe de la dérivation en chaîne :

$$\begin{aligned} \frac{\partial J}{\partial z_i} &= \frac{\partial J}{\partial a_i} \cdot \frac{\partial a_i}{\partial z_i} \\ \frac{\partial J}{\partial a_i} &= -\frac{1}{m} \left[\frac{y_i}{a_i} - \frac{1 - y_i}{1 - a_i} \right] \\ \frac{\partial a_i}{\partial z_i} &= a_i \cdot (1 - a_i) \\ \frac{\partial J}{\partial z_i} &= -\frac{1}{m} \left[\frac{y_i}{a_i} - \frac{1 - y_i}{1 - a_i} \right] \cdot a_i \cdot (1 - a_i) \\ &= \frac{1}{m} [a_i - y_i] \end{aligned}$$

Dans le contexte de la descente de gradient stochastique, la mise à jour s'effectue pour chaque exemple d'entraînement, ce qui implique $m = 1$ (où m représente le nombre total d'exemples d'entraînement).

Au final, on obtient :

$$\Delta[i] = y_i - a_i$$

Variante SGD avec Moment (Momentum SGD)

Introduit un terme de moment pour atténuer les oscillations et accélérer la convergence, pour chaque paramètre θ (poids ou biais) [12] :

$$v_t = \beta v_{t-1} + (1 - \beta) \frac{\partial J}{\partial \theta} \quad (4.21)$$

$$\theta \leftarrow \theta - \alpha v_t \quad (4.22)$$

où β est le coefficient de moment et v_t est le moment à l'itération t .

Variante SGD avec Nesterov Accelerated Gradient (NAG)

Modifie l'ordre d'application du moment pour améliorer la convergence, pour chaque paramètre θ (poids ou biais) [51] :

$$v_t = \beta v_{t-1} + (1 - \beta) \frac{\partial J}{\partial \theta} \quad (4.23)$$

$$\theta \leftarrow \theta - \alpha(\beta v_t + (1 - \beta) \frac{\partial J}{\partial \theta}) \quad (4.24)$$

où β est le coefficient de moment et v_t est le moment à l'itération t .

Variante SGD avec Learning Rate Decay

Diminue le taux d'apprentissage au fil du temps pour permettre une convergence plus fine, pour chaque paramètre θ (poids ou biais) [16] :

$$\alpha_t = \frac{\alpha}{1 + \text{decay} \times t} \quad (4.25)$$

$$\theta \leftarrow \theta - \alpha_t \frac{\partial J}{\partial \theta} \quad (4.26)$$

où decay est le taux de décroissance du taux d'apprentissage.

Variante SGD avec Adagrad

Adapte le taux d'apprentissage pour chaque paramètre θ (poids ou biais) en fonction de son historique de mises à jour [24] :

$$\theta \leftarrow \theta - \frac{\alpha}{\sqrt{G_t + \epsilon}} \frac{\partial J}{\partial \theta} \quad (4.27)$$

où G_t est la somme des carrés des gradients jusqu'à l'itération t et ϵ est une petite constante.

Variante SGD avec RMSprop

Utilise une décroissance exponentielle du taux d'apprentissage en adaptant le taux d'apprentissage pour chaque paramètre θ (poids ou biais) [67] :

$$\theta \leftarrow \theta - \frac{\alpha}{\sqrt{E[G^2]_t + \epsilon}} \frac{\partial J}{\partial \theta} \quad (4.28)$$

où $E[G^2]_t$ est la moyenne exponentielle des carrés des gradients.

Variante SGD avec Adam

Les gradients sont calculés en utilisant la règle de la chaîne comme précédemment, mais Adam introduit des paramètres supplémentaires pour le momentum et les taux d'apprentissage adaptatifs. Les règles de mise à jour pour chaque paramètre θ (poids ou biais) avec Adam sont données par [41] :

$$\begin{aligned}
m_t &\leftarrow \beta_1 m_{t-1} + (1 - \beta_1) \frac{\partial J}{\partial \theta} \\
v_t &\leftarrow \beta_2 v_{t-1} + (1 - \beta_2) \left(\frac{\partial J}{\partial \theta} \right)^2 \\
m_t^{\text{corrected}} &\leftarrow \frac{m_t}{1 - \beta_1^t} \\
v_t^{\text{corrected}} &\leftarrow \frac{v_t}{1 - \beta_2^t} \\
\theta &\leftarrow \theta - \alpha \cdot \frac{m_t^{\text{corrected}}}{\sqrt{v_t^{\text{corrected}} + \epsilon}}
\end{aligned} \tag{4.29}$$

Où β_1 et β_2 sont les taux de décroissance exponentielle pour les estimations des moments, ϵ est une petite constante pour éviter la division par zéro et t est le numéro d'itération.

4.2.4 Algorithme d'apprentissage par rétropropagation

Le processus d'apprentissage par rétropropagation utilisant la méthode de la descente de gradient stochastique (SGD), discuté dans la section 4.2.3.2, est présenté dans l'algorithme (algorithm 5) [37].

4.2.5 Exemple

Soit un échantillon d'entraînement $x = [2, -1]$, $y = 1$ que nous voulons utiliser pour l'entraînement du réseau de neurones illustré dans la figure fig. 4.4.

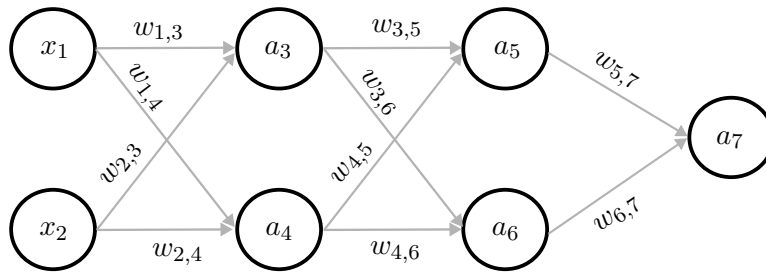


Figure 4.4: Architecture d'exemple

Les valeurs initiales attribuées aux poids w sont les suivantes :

$$\begin{array}{lll}
w_{1,3} = 0.5 & w_{3,5} = 1 & \\
w_{1,4} = -1 & w_{3,6} = -1 & w_{5,7} = 1 \\
w_{2,3} = 1.5 & w_{4,5} = 3 & w_{6,7} = -3 \\
w_{2,4} = -2 & w_{4,6} = -4 &
\end{array}$$

Solution

La première étape consiste à calculer les valeurs pour chaque neurone. Pour ce faire, nous utilisons la **propagation avant** en utilisant la formule que nous avons vue dans la section 4.2.1.3

Algorithm 5 Algorithme d'apprentissage par rétropropagation

```

1: function BACKPROPAGATION(examples, network)
2:   Inputs: examples, vecteur d'entrée X, et vecteur de sortie Y
3:   network, un réseau multicouche avec  $L$  couches, poids  $w_{ij}$ , biais  $b_i$ , fonction
   d'activation  $g$ 
4:   Local Variables:  $\Delta$ , un vecteur d'erreurs
5:   for each poids  $w_{ij}$  et biais  $b_i$  dans network do
6:      $w_{ij} \leftarrow$  un petit nombre aléatoire
7:      $b_i \leftarrow$  un petit nombre aléatoire
8:   end for
9:   repeat
10:    for each exemple  $(x, y)$  dans examples do
11:       $\triangleright$  Propager les entrées vers l'avant pour calculer les sorties
12:      for each neurone  $i$  dans la couche d'entrée do
13:         $a_i \leftarrow x_i$ 
14:      end for
15:      for  $l = 2$  à  $L$  do
16:        for each neurone  $j$  dans la couche  $l$  do
17:           $z_j \leftarrow b_j + \sum_i w_{i,j} a_i$ 
18:           $a_j \leftarrow g(z_j)$ 
19:        end for
20:      end for
21:       $\triangleright$  Propager les deltas en arrière depuis la couche de sortie jusqu'à la couche d'entrée
22:      for each neurone  $j$  dans la couche de sortie do
23:         $\Delta[j] \leftarrow y_j - a_j$   $\triangleright (\Delta[j] = -\frac{\partial J}{\partial z_j})$ 
24:      end for
25:      for  $l = L - 1$  à  $1$  do
26:        for each neurone  $i$  dans la couche  $l$  do
27:           $\Delta[i] = g(z_i)(1 - g(z_i)) \sum_j w_{i,j} \Delta[j]$ 
28:        end for
29:      end for
30:       $\triangleright$  Mettre à jour chaque poids dans le réseau en utilisant les deltas
31:      for each poids  $w_{ij}$  et biais  $b_i$  dans network do
32:         $w_{i,j} \leftarrow w_{i,j} + \alpha a_i \Delta[j]$ 
33:         $b_j \leftarrow b_j + \alpha \Delta[j]$ 
34:      end for
35:    end for
36:  until certain critère d'arrêt est atteint
37:  return network
38: end function

```

:

$$a_k = g \left(\sum_j w_{j,k} a_j \right)$$

$$\begin{aligned}
a_3 &= g(w_{1,3} \cdot x_1 + w_{2,3} \cdot x_2) = \frac{1}{1 + e^{-0.5}} = 0.378 \\
a_4 &= g(w_{1,4} \cdot x_1 + w_{2,4} \cdot x_2) = \frac{1}{1 + e^0} = 0.5 \\
a_5 &= g(w_{3,5} \cdot a_3 + w_{4,5} \cdot a_4) = \frac{1}{1 + e^{1.878}} = 0.867 \\
a_6 &= g(w_{3,6} \cdot a_3 + w_{4,6} \cdot a_4) = \frac{1}{1 + e^{-2.378}} = 0.085 \\
a_7 &= g(w_{5,7} \cdot a_5 + w_{6,7} \cdot a_6) = \frac{1}{1 + e^{0.612}} = 0.648
\end{aligned}$$

Avec $a_1 = x_1 = 2$ et $a_2 = x_2 = -1$.

La deuxième étape consiste à calculer les Δ . Pour ce faire, nous utilisons la **rétropropagation** en appliquant les formules que nous avons vues dans la section 4.2.3.2 :

$$\begin{aligned}
\Delta[j] &= g(z_j)(1 - g(z_j)) \sum_k w_{j,k} \Delta[k] \\
\Delta[i] &= y_i - a_i \quad (\text{Pour la couche de sortie})
\end{aligned}$$

$$\begin{aligned}
\Delta[7] &= y - a_7 = 1 - 0.648 = 0.352 \\
\Delta[6] &= a_6(1 - a_6)(w_{6,7} \Delta[7]) = -0.082 \\
\Delta[5] &= a_5(1 - a_5)(w_{5,7} \Delta[7]) = 0.041 \\
\Delta[4] &= a_4(1 - a_4)(w_{4,6} \Delta[6] + w_{4,5} \Delta[5]) = 0.113 \\
\Delta[3] &= a_3(1 - a_3)(w_{3,6} \Delta[6] + w_{3,5} \Delta[5]) = 0.029
\end{aligned}$$

Enfin, pour la **mise à jour** des poids w , nous utilisons la formule que nous avons vue dans la section 4.2.3.2 :

$$w_{i,j} \leftarrow w_{i,j} + \alpha a_i \Delta[j]$$

pour $\alpha = 0.1$:

$$\begin{aligned}
w_{1,3} &\leftarrow w_{1,3} + \alpha a_1 \Delta[3] = 0.506 & w_{3,5} &\leftarrow w_{3,5} + \alpha a_3 \Delta[5] = 1.002 \\
w_{1,4} &\leftarrow w_{1,4} + \alpha a_1 \Delta[4] = -0.977 & w_{3,6} &\leftarrow w_{3,6} + \alpha a_3 \Delta[6] = -1.003 \\
w_{2,3} &\leftarrow w_{2,3} + \alpha a_2 \Delta[3] = 1.497 & w_{4,5} &\leftarrow w_{4,5} + \alpha a_4 \Delta[5] = 3.002 \\
w_{2,4} &\leftarrow w_{2,4} + \alpha a_2 \Delta[4] = -2.011 & w_{4,6} &\leftarrow w_{4,6} + \alpha a_4 \Delta[6] = -4.004
\end{aligned}$$

$$\begin{aligned}
w_{5,7} &\leftarrow w_{5,7} + \alpha a_5 \Delta[7] = 1.031 \\
w_{6,7} &\leftarrow w_{6,7} + \alpha a_6 \Delta[7] = -2.997
\end{aligned}$$

4.3 Réseaux de Neurones Convolutifs (CNN)

Les réseaux de neurones convolutifs (*Convolutional Neural Networks* – *CNN*) constituent une catégorie de modèles d'apprentissage profond (*Deep Learning*) particulièrement adaptée au

traitement des données multidimensionnelles telles que les images, les spectres et les signaux industriels [30].

Contrairement aux méthodes classiques nécessitant une extraction manuelle des caractéristiques, les CNN permettent l'extraction automatique des informations importantes à partir des données d'entrée [34].

Le principe des CNN repose sur l'application d'opérations de convolution afin d'extraire automatiquement les caractéristiques importantes des données [45].

4.3.1 Architecture générale d'un CNN

Un CNN est généralement constitué des couches suivantes [30], comme illustré dans la Figure 4.5 :

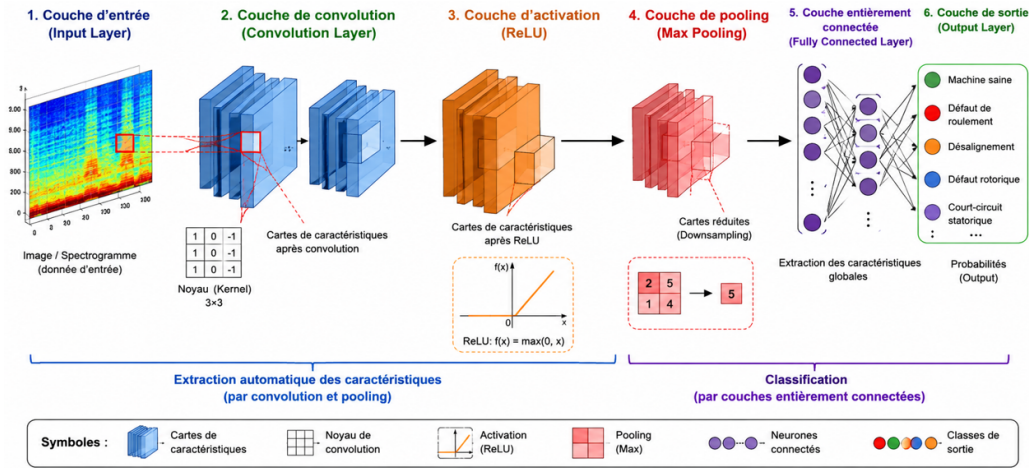


Figure 4.5: Architecture générale d'un réseau de neurones convolutif (CNN)

4.3.1.1 Couche d'entrée (*Input Layer*)

Cette couche reçoit les données brutes sous forme matricielle.

Exemple :

$$X \in \mathbb{R}^{m \times n}$$

où :

- m représente le nombre de lignes ;
- n représente le nombre de colonnes.

4.3.1.2 Couche de convolution (*Convolution Layer*)

La couche de convolution applique un noyau (*Kernel*) sur les données d'entrée afin d'extraire les caractéristiques locales [45].

L'opération de convolution est donnée par :

$$S(i, j) = \sum_m \sum_n I(m, n) K(i - m, j - n)$$

où :

- I : image ou signal d'entrée ;
- K : noyau de convolution ;
- S : carte de caractéristiques obtenue.

Exemple de noyau de convolution :

$$K = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

Ce filtre permet la détection des contours verticaux.

4.3.1.3 Fonction d'activation

Après la convolution, une fonction d'activation est appliquée afin d'introduire une non-linéarité dans le modèle.

La fonction la plus utilisée est ReLU [30] :

$$f(x) = \max(0, x)$$

Cette fonction conserve uniquement les valeurs positives.

4.3.1.4 Couche de Pooling

Le *Pooling* permet de réduire la dimension des données tout en conservant les caractéristiques essentielles [28].

Le *Max Pooling* est la méthode la plus utilisée.

Exemple :

$$\begin{bmatrix} 2 & 5 \\ 1 & 4 \end{bmatrix} \rightarrow 5$$

Cette opération réduit :

- la complexité du modèle ;
- le temps de calcul ;
- le risque de surapprentissage.

4.3.1.5 Couche entièrement connectée (*Fully Connected Layer*)

Les caractéristiques extraites sont transformées en vecteur puis envoyées vers des couches entièrement connectées afin d'effectuer la classification finale [43].

4.3.2 Exemple

Considérons un système de surveillance vibratoire d'un moteur asynchrone où les signaux vibratoires sont transformés en spectrogrammes puis utilisés comme images d'entrée du CNN.

La matrice d'entrée est donnée par :

$$I = \begin{bmatrix} 3 & 1 & 2 \\ 4 & 0 & 1 \\ 2 & 3 & 1 \end{bmatrix}$$

Le noyau de convolution utilisé est :

$$K = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

Étape 1 : Convolution Le calcul de convolution donne :

$$\begin{aligned} S &= (3 \times 1) + (1 \times 0) + (2 \times (-1)) \\ &\quad + (4 \times 1) + (0 \times 0) + (1 \times (-1)) \\ &\quad + (2 \times 1) + (3 \times 0) + (1 \times (-1)) \\ S &= 3 + 0 - 2 + 4 + 0 - 1 + 2 + 0 - 1 \\ S &= 5 \end{aligned}$$

La valeur obtenue dans la carte de caractéristiques est :

$$S = 5$$

Étape 2 : Fonction ReLU

$$f(5) = \max(0, 5) = 5$$

La sortie reste positive après activation.

Étape 3 : Classification Après les différentes couches de traitement, le CNN fournit les probabilités suivantes :

Table 4.1: Résultats de classification du CNN

Classe	Probabilité
Machine saine	0.08
Défaut de roulement	0.87
Désalignement	0.05

Le système classe donc la machine dans la catégorie :

Défaut de roulement

avec une probabilité de :

$$87\%$$

4.4 Réseaux de neurones LSTM

Les réseaux de neurones récurrents (Recurrent Neural Networks – RNN) sont des architectures particulièrement adaptées au traitement des données séquentielles et temporelles. Cependant, les RNN classiques présentent des limitations importantes lors de l'apprentissage de dépendances à long terme, notamment à cause du problème de disparition et d'explosion du gradient lors de l'entraînement par rétropropagation temporelle (*Backpropagation Through Time – BPTT*). Pour remédier à cette limitation, le modèle Long Short-Term Memory (LSTM) a été proposé par Hochreiter et Schmidhuber [36] afin d'améliorer la mémorisation des dépendances temporelles sur de longues séquences.

Le réseau LSTM est une version améliorée des RNN, capable de conserver des informations importantes pendant de longues périodes grâce à une structure interne composée de cellules mémoire et de portes de contrôle. Cette architecture permet au modèle de sélectionner les informations pertinentes à conserver ou à oublier au cours du temps. Les modèles LSTM sont largement utilisés dans les applications de séries temporelles, notamment la reconnaissance vocale, la prévision, le traitement du langage naturel ainsi que le diagnostic des défauts industriels [73].

4.4.1 Structure d'une cellule LSTM

Une cellule LSTM est principalement constituée de trois portes :

- la porte d'oubli (*Forget Gate*) ;
- la porte d'entrée (*Input Gate*) ;
- la porte de sortie (*Output Gate*).

Comme illustré dans la figure 4.6, une cellule LSTM contient plusieurs mécanismes de contrôle permettant de gérer le flux d'information au sein du réseau. Les portes d'oubli, d'entrée et de sortie assurent respectivement la suppression, l'ajout et la transmission des informations pertinentes dans la mémoire de la cellule.

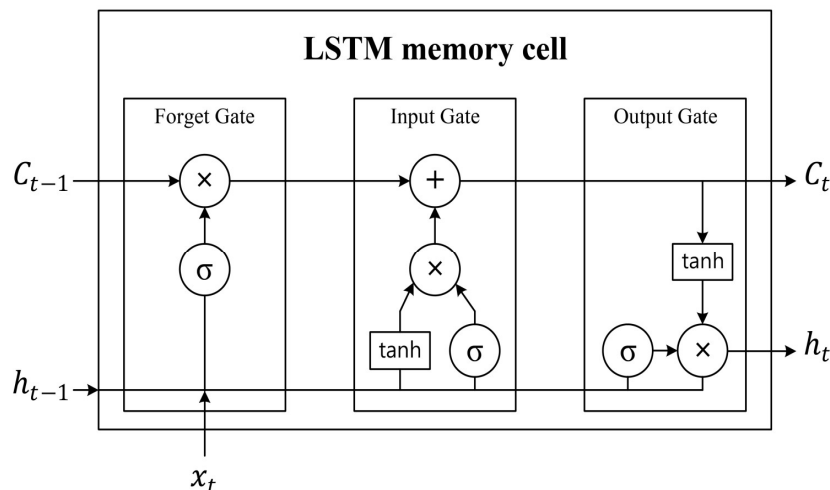


Figure 4.6: Structure interne d'une cellule LSTM

Ces portes contrôlent le flux d'information dans la cellule mémoire afin d'assurer un apprentissage efficace des dépendances temporelles.

Porte d'oubli La porte d'oubli permet de déterminer quelles informations de l'état précédent doivent être supprimées. Elle est exprimée par :

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (4.30)$$

où :

- x_t représente l'entrée au temps t ;
- h_{t-1} représente l'état caché précédent ;
- W_f est la matrice des poids ;
- b_f est le biais ;
- σ est la fonction d'activation sigmoïde.

La sortie de cette porte varie entre 0 et 1. Une valeur proche de 0 indique que l'information doit être oubliée, tandis qu'une valeur proche de 1 signifie qu'elle doit être conservée.

Porte d'entrée La porte d'entrée détermine quelles nouvelles informations seront ajoutées à la mémoire de la cellule :

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (4.31)$$

Une nouvelle information candidate est calculée par :

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (4.32)$$

L'état de la cellule est ensuite mis à jour selon :

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (4.33)$$

où C_t représente l'état mémoire de la cellule au temps t .

Porte de sortie La porte de sortie détermine l'information qui sera transmise à la sortie du réseau :

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (4.34)$$

L'état caché final est donné par :

$$h_t = o_t \times \tanh(C_t) \quad (4.35)$$

Cette architecture permet aux réseaux LSTM d'extraire efficacement les caractéristiques temporelles importantes dans les signaux industriels complexes.

4.5 Conclusion

Dans ce chapitre, les principaux concepts de l'apprentissage profond appliqué au diagnostic des machines électriques ont été présentés. Les architectures les plus couramment utilisées, notamment les réseaux de neurones profonds (DNN), les réseaux de neurones convolutifs (CNN) et les réseaux à mémoire longue à court terme (LSTM), ont été décrites en mettant en évidence leurs principes de fonctionnement et leurs caractéristiques. Cette étude a permis de présenter les fondements théoriques des approches d'apprentissage profond ainsi que leur intérêt pour

le développement de systèmes de diagnostic intelligents capables d'exploiter efficacement les données issues des machines électriques. Dans la continuité de ce travail, le chapitre suivant sera consacré à l'application des différentes méthodes étudiées au cours de cette thèse, incluant les techniques de traitement du signal, d'apprentissage automatique et d'apprentissage profond. Les performances de ces approches seront analysées et comparées sur des systèmes réels afin d'identifier leurs avantages, leurs limites et leur pertinence dans le cadre du diagnostic des défauts des machines électriques.

Application et discussion

5.1 Introduction

Après avoir présenté les fondements théoriques des techniques de traitement du signal, d'apprentissage automatique et d'apprentissage profond appliquées au diagnostic des machines électriques, ce chapitre est consacré à leur validation expérimentale sur des systèmes réels au sein d'un laboratoire. L'objectif est d'évaluer l'efficacité de ces approches pour la détection et l'identification de différents défauts mécaniques et électriques dans des conditions de fonctionnement représentatives. Les travaux présentés s'appuient sur plusieurs études de cas ayant fait l'objet de publications scientifiques, témoignant de leur intérêt et de leur contribution au domaine du diagnostic des machines électriques. Une analyse comparative des différentes méthodes étudiées sera également menée afin de mettre en évidence leurs performances respectives, leurs avantages ainsi que leurs limites, dans la perspective du développement de systèmes de surveillance et de maintenance prédictive plus performants.

5.2 Application de l'analyse vibratoire à la détection des défauts mécaniques combinés d'une éolienne

5.2.1 Méthodologie

La figure ([fig. 5.1](#)) présente la configuration expérimentale utilisée pour mener notre étude sur la détection de défauts combinés affectant le système éolien basé sur un générateur couplé à un moteur à courant continu à vitesse variable. Son énergie cinétique est transmise à l'alternateur et convertie en énergie électrique. Les détails de la plaque signalétique du moteur et du générateur utilisés sont donnés dans le tableau ([table 5.1](#)).

L'analyse des vibrations caractéristiques pour la surveillance et le diagnostic des générateurs comprend plusieurs étapes principales, à savoir l'acquisition de données, qui consiste à obtenir les échantillons de vibration au moment des conditions saines et défectueuses mises en œuvre par la création délibérée d'un défaut combiné consistant en un déséquilibre dans l'arbre et un défaut de desserrage dans l'alternateur. Ensuite, le signal de vibration est traité par MATLAB pour tester l'algorithme, où il est nécessaire de transformer les signaux du domaine temporel en domaine fréquentiel ou en temps-fréquence. À cette fin, plusieurs techniques de traitement du signal ont été utilisées, telles que la transformée de Fourier rapide (FFT) et la transformée en ondelettes discrètes (DWT), donc la dernière étape est la reconnaissance du défaut affectant les systèmes éoliens.

Type de machine	Moteur DC	Générateur AC
Fabricant	Leroy-Somer	Leroy-Somer
Pays	France	France
Puissance	3 kW	3 KVA
Tension	220 V	220 V
Courant	10 A	7.5 A
Vitesse	2400 tr/min	2400 tr/min

Table 5.1: Détails des machines



Figure 5.1: Configuration expérimentale pour la détection de défauts combinés

5.2.2 Signatures fréquentes des défaillances dans un système éolien

Quelle que soit l'attention portée à la construction et à l'assemblage des différentes parties du système de turbine éolienne, il n'est pas possible de faire coïncider l'axe de rotation avec le centre de gravité de chaque section élémentaire du rotor, ce qui caractérise le défaut de déséquilibre. En conséquence, l'arbre rotatif est soumis à des forces centrifuges, qui le déforment. Ces forces entraînent des vibrations liées à la fréquence de rotation f_r . Les déséquilibres proviennent généralement de défauts d'usinage et de l'assemblage du rotor avec le système de turbine éolienne. En fonctionnement, les rotors des générateurs peuvent également se déformer sous l'effet d'un chauffage asymétrique ou suite à la rupture et au départ d'une pièce du rotor ou d'une pale. Dans le cas d'un générateur dégradé, une évolution instantanée des vibrations est observée. Pour une meilleure exploitation du signal obtenu, la transformée de Fourier rapide (FFT) est une technique fiable, précise et largement utilisée pour la détection de défauts dans le domaine des machines tournantes ; ensuite, l'évolution de la signature vibratoire est observée lors du suivi de l'amplitude et de la phase de la fréquence de vibration $f_{\text{déséquilibre}}$ déterminée par l'équation suivante :

$$f_{\text{déséquilibre}} = 1 \times f_r. \quad (5.1)$$

On observe également divers phénomènes vibratoires suite à un mauvais serrage de la structure de la ferme éolienne causé par une mauvaise fixation des machines et des composants du système sur leurs supports ou massifs. Si l'assemblage est défectueux ou s'il y a du jeu dû à un déséquilibre ou à un autre défaut, le mouvement vibratoire sinusoïdal sera transformé en un mouvement

périodique d'autant plus riche en harmoniques que le signal sera déformé. Dans ce cas, l'analyse du spectre du signal de vibration révèle des harmoniques et des sous-harmoniques d'ordre demi-fréquence de rotation et ses multiples :

$$f_{\text{desserrage}} = \frac{1}{2} \times f_r. \quad (5.2)$$

5.2.3 Résultats et discussion

Approches FFT

La figure (fig. 5.2) présente le signal de vibration dans le domaine fréquentiel des différentes conditions du générateur. Nous constatons uniquement qu'en présence de défauts combinés, les amplitudes FFT sont plus élevées que dans un état sain. On peut également remarquer qu'il y a deux pics principaux aux fréquences de 20 Hz et 40 Hz.

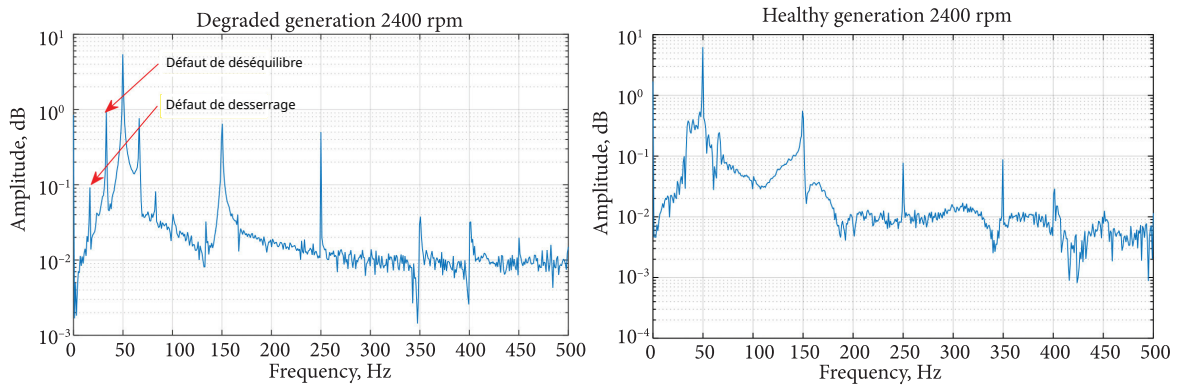


Figure 5.2: Signaux dans le domaine fréquentiel de différentes conditions du générateur

Approches DWT

Dans ce travail, la transformée en ondelettes discrète (DWT) a été appliquée pour traiter les signaux de vibration collectés sur le générateur dans le domaine temps-fréquence. Pour cela, six niveaux de décomposition du signal ont été utilisés. La figure (fig. 5.3) illustre la décomposition du signal de vibration ; nous avons remarqué un lissage dans le cas d'une génération saine.

La figure (fig. 5.4) présente le résultat du traitement du signal de vibration par la transformée en ondelettes discrète (DWT) sous le défaut de déséquilibre et de desserrage du rotor avec une vitesse de 2400 tr/min. Contrairement au cas du générateur sain, nous avons clairement remarqué des comportements très différents des signaux indiquant les effets des défauts mécaniques combinés sur le comportement vibratoire du générateur, traduits par l'apparition d'oscillations dans, D3 et D4 confirmant l'état défectueux.

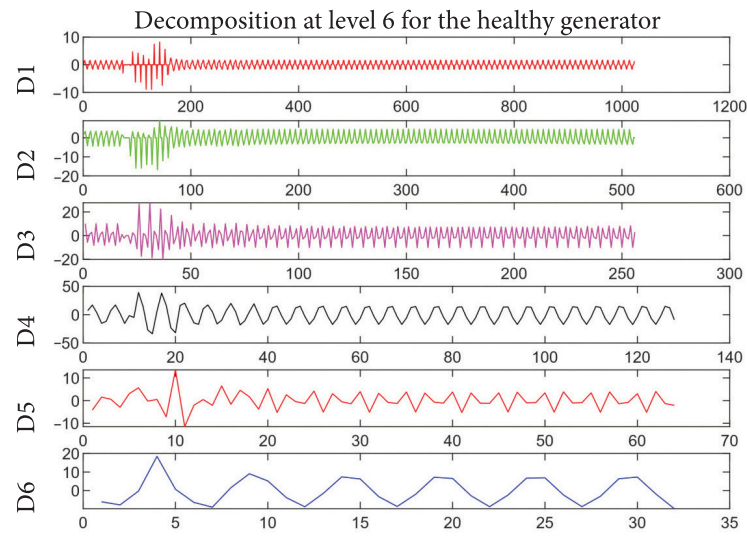


Figure 5.3: Décomposition du signal de vibration du générateur sain à une vitesse de 2400 tr/min

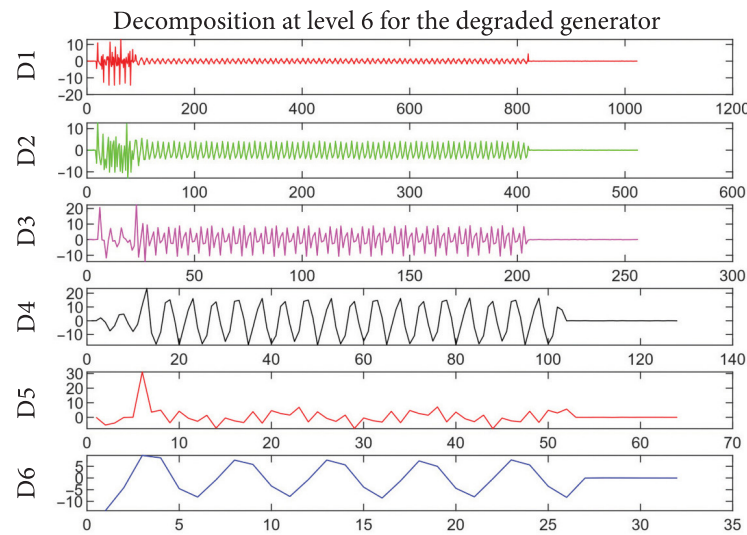


Figure 5.4: Décomposition du signal de vibration de déséquilibre du rotor et d'un défaut de desserrage à une vitesse de 2400 tr/min

5.3 Application de CWT et CNN au diagnostic des défauts de roulements des moteurs à induction

5.3.1 Approche proposée

Dans ce manuscrit, nous proposons une nouvelle méthode qui nécessite moins de temps de calcul et présente une meilleure précision que les méthodes de l'état de l'art lors des phases d'apprentissage et de test en apprentissage profond. Dans notre implémentation, nous avons utilisé des scalogrammes et le réseau *SqueezeNet* basé sur une architecture CNN (*Convolutional Neural Network*) afin d'étudier la classification de trois états de roulements dans les moteurs à induction : état sain (normal), défaut de bague intérieure (*inner race*) et défaut de bague extérieure (*outer race*). L'organigramme de la méthodologie proposée pour la classification des défauts est présenté à la Figure 5.5.

Tous les signaux vibratoires bruts utilisés dans cette étude ont d'abord été segmentés en cycles. Ces signaux ont ensuite été analysés à l'aide de la transformée en ondelettes continue (CWT, *Continuous Wavelet Transform*). Les résultats obtenus ont été convertis en une représentation bidimensionnelle sous forme de scalogrammes, générés à partir de trois types différents de transformées en ondelettes continues. Ces images ont ensuite été redimensionnées à une résolution de 227×227 pixels au format RGB, puis utilisées comme entrée du réseau de neurones profond *SqueezeNet*.

Dans une seconde étape, les performances des différents types de CWT ont été évaluées en classifiant les signaux vibratoires correspondant aux trois conditions de fonctionnement (état sain, défaut de bague intérieure et défaut de bague extérieure) à l'aide du modèle *SqueezeNet*. L'objectif est de comparer la précision de classification obtenue avec chaque type de transformée en ondelettes continue afin d'identifier la méthode la plus performante pour le diagnostic des défauts de roulements.

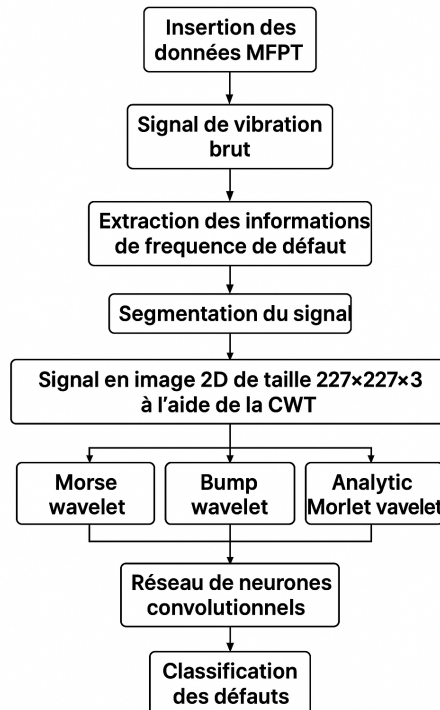


Figure 5.5: Organigramme de la méthodologie proposée pour la classification des défauts.

5.3.1.1 Architecture CNN

Dans cette étude, les signaux vibratoires du domaine temporel sont convertis en représentations temps–fréquence bidimensionnelles (2D) sous forme de spectrogrammes. La méthode proposée repose sur le modèle CNN utilisant le réseau *SqueezeNet*, introduit dans [38], et se déroule en trois phases. Dans un premier temps, tous les segments des signaux vibratoires sont transformés en cartes de caractéristiques sous forme de scalogrammes. Étant donné que le réseau *SqueezeNet* requiert des images d’entrée de taille $227 \times 227 \times 3$, les images générées sont redimensionnées en conséquence.

La phase finale du processus de classification de *SqueezeNet* repose sur les couches *conv10* et *Classification Layer predictions*, qui permettent respectivement l’extraction des caractéristiques de l’image et la classification de l’image d’entrée. Ces couches combinent les caractéristiques extraites afin de produire les probabilités d’appartenance aux différentes classes, les étiquettes prédites ainsi qu’une valeur de perte (*loss*).

Pour adapter *SqueezeNet* à une nouvelle tâche de classification d’images, telle que la classification d’images représentant différents états de roulements, il est nécessaire de remplacer ces deux couches par de nouvelles couches adaptées au domaine d’application considéré. La couche de classification est chargée de déterminer les classes de sortie du réseau ; elle doit donc être remplacée par une couche ne contenant pas de classes prédéfinies afin de permettre l’apprentissage de nouvelles catégories. Lors de la phase d’entraînement, la fonction *trainNetwork* détermine automatiquement les classes de sortie appropriées en fonction des données d’apprentissage fournies.

5.3.1.2 Approche proposée pour le diagnostic des défauts de roulements

Dans cette étude, une nouvelle méthode de détection des défauts des roulements à éléments roulants est proposée. Cette approche repose sur l’utilisation des réseaux de neurones convolutifs (CNN) afin d’améliorer les performances de diagnostic. Ce choix est motivé par la disponibilité limitée des données de défauts de roulements dans les applications industrielles réelles ainsi que par la complexité des méthodes conventionnelles de diagnostic intelligent.

La méthode proposée est illustrée par l’organigramme présenté à la Figure 5.5 et se déroule selon les étapes suivantes :

- **Étape 1 : Acquisition des données vibratoires**
Les signaux vibratoires sont acquis à partir de roulements présentant différents états de fonctionnement, notamment des conditions normales et des défauts.
- **Étape 2 : Prétraitement et génération des images temps–fréquence**
Les signaux vibratoires sont d’abord segmentés, puis transformés en images temps–fréquence de dimensions $227 \times 227 \times 3$ à l’aide de la transformée en ondelettes continue (CWT, *Continuous Wavelet Transform*). Les images obtenues sont ensuite réparties en ensembles d’apprentissage et de test. Chaque image est associée à une étiquette correspondant à la classe du défaut étudié.
- **Étape 3 : Extraction des caractéristiques et apprentissage du modèle**
Le logiciel MATLAB est utilisé pour charger le réseau pré-entraîné *SqueezeNet* via la bibliothèque *Add-Ons*. Les images préparées à l’étape précédente sont ensuite introduites dans le réseau. La couche entièrement connectée (*Fully Connected Layer*) est exploitée pour extraire les caractéristiques de haut niveau à partir des images d’apprentissage et de test. Les caractéristiques extraites des données d’apprentissage sont ensuite utilisées comme variables prédictives pour l’entraînement du classificateur basé sur la fonction *SoftMax*.
- **Étape 4 : Évaluation des performances du diagnostic**

Les échantillons de test sont introduits dans le modèle entraîné afin d'évaluer les performances du système de diagnostic. La précision de classification obtenue permet de mesurer l'efficacité de la méthode proposée pour l'identification des différents défauts de roulements.

5.3.2 Description de la base de données MFPT

Cette étude évalue un modèle de diagnostic des défauts de roulements à l'aide de jeux de données open source, en mettant particulièrement l'accent sur la base de données *MFPT*. L'utilisation de jeux de données librement accessibles permet de réaliser une évaluation approfondie des performances du modèle tout en améliorant sa reproductibilité, sa polyvalence et sa transparence dans le cadre de la recherche scientifique.

L'exploitation des données MFPT dans cette étude renforce la crédibilité des résultats obtenus, car elle permet d'évaluer rigoureusement le modèle dans différents scénarios de défaillance des roulements [23].

5.3.2.1 Description des données

Les travaux de recherche et d'analyse des défaillances des roulements à éléments roulants ont été réalisés à l'aide des données du projet *MFPT* (*Machinery Failure Prevention Technology*). La base de données MFPT comprend trois ensembles de données de défauts réels ainsi que trois ensembles de données expérimentales de vibrations de roulements. Les expériences portent sur trois conditions de fonctionnement : l'état sain (*baseline*), le défaut de la bague intérieure (*inner race fault*) et le défaut de la bague extérieure (*outer race fault*). Les données de référence (état sain) sont constituées de trois fichiers. Chaque fichier contient des signaux échantillonnés à une fréquence de 97 656 Hz pendant 6 secondes, ce qui représente un total de 585 936 échantillons, sous une charge constante de 270 lb. L'ensemble de données correspondant aux défauts de la bague intérieure est composé de sept fichiers. Les signaux ont été échantillonnés à une fréquence de 48 828 Hz pendant 3 secondes, produisant 146 484 échantillons par fichier. Les essais ont été réalisés sous différentes charges variant de 0 à 300 lb. De même, l'ensemble de données relatif aux défauts de la bague extérieure comprend sept fichiers enregistrés à une fréquence d'échantillonnage de 48 828 Hz pendant 3 secondes. Les mesures ont été effectuées sous sept niveaux de charge différents, compris entre 25 et 300 lb. Les roulements utilisés dans les expériences de la base de données MFPT sont des roulements à billes à gorge profonde comportant huit éléments roulants, un diamètre primitif de 31,62 mm, un diamètre de bille de 5,97 mm ainsi qu'un angle de contact spécifique. Dans cette étude, les données vibratoires issues des trois conditions de fonctionnement (état sain, défaut de bague intérieure et défaut de bague extérieure) sont exploitées pour l'évaluation des performances du système de diagnostic.

5.3.2.2 Traitement des données

Dans cette expérimentation, l'ensemble des données issues des trois catégories de défauts a été utilisé. Les trois fichiers correspondant à l'état de référence (état sain) ont été segmentés en 117 segments, chacun contenant 5 000 échantillons. Pour les données relatives aux défauts de la bague intérieure, chacun des sept fichiers a été divisé en 58 segments, tandis que chacun des sept fichiers associés aux défauts de la bague extérieure a été segmenté en 117 segments.

Pour chaque condition de fonctionnement, un nombre correspondant d'images temps-fréquence a été généré en appliquant les différents types de transformée en ondelettes continue (CWT). Après combinaison des différentes conditions de charge pour les trois types de défauts, un total de :

Table 5.2: Caractéristiques des signaux, longueurs d’acquisition et nombre de segments des ensembles d’apprentissage et de test pour les différentes conditions de défaut.

	Signals types	Signal length	Number of segments
Training	baseline_1	585936	117
	baseline_2	585936	117
	InnerRaceFault_various load 1	146484	58
	InnerRaceFault_various load 2	146484	58
	InnerRaceFault_various load 3	146484	58
	InnerRaceFault_various load 4	146484	58
	InnerRaceFault_various load 5	146484	58
	OuterRaceFault_1	585936	117
	OuterRaceFault_2	585936	117
	OuterRaceFault_various load 1	146484	58
	OuterRaceFault_various load 1	146484	58
	OuterRaceFault_various load 1	146484	58
	OuterRaceFault_various load 1	146484	58
	OuterRaceFault_various load 1	146484	58
	OuterRaceFault_various load 1	146484	58
Testing	baseline_3	585936	117
	InnerRaceFault_various load 6	146484	58
	InnerRaceFault_various load 7	146484	58
	OuterRaceFault_3	585936	117
	OuterRaceFault_various load 6	146484	58
	OuterRaceFault_various load 7	146484	58

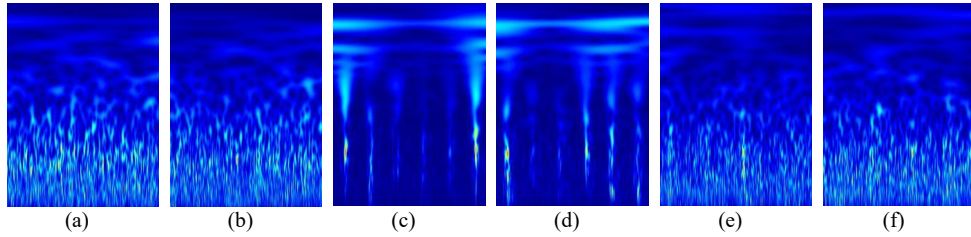


Figure 5.6: Scalogrammes obtenus à l’aide de l’ondelette Morse : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.

- 234 images temps–fréquence a été obtenu pour les deux premiers ensembles de données de référence ;
- 290 images temps–fréquence a été généré à partir des cinq premiers signaux de défaut de la bague intérieure ;
- 524 images temps–fréquence a été obtenu pour les signaux de défaut de la bague extérieure.

Enfin, comme indiqué dans le Tableau 5.2, 80 % des images ont été sélectionnées pour constituer l’ensemble d’apprentissage, tandis que les 20 % restants ont été réservés à l’évaluation du modèle. Les Figures 5.6, 5.7 et 5.8 présentent respectivement les images temps–fréquence obtenues à l’aide des trois types de CWT considérés. La Figure 5.6 illustre les scalogrammes générés par l’ondelette Morse, la Figure 5.7 présente ceux obtenus avec l’ondelette Morlet, tandis que la Figure 5.8 montre les résultats issus de l’ondelette Bump. Pour chacune de ces figures, les sous-figures (a) et (b) correspondent à l’état sain, les sous-figures (c) et (d) représentent les défauts de la bague intérieure, tandis que les sous-figures (e) et (f) illustrent les défauts de la bague extérieure.

5.3.3 Résultats et discussion

Après la génération des images et leur répartition selon un ratio de 80 % pour l’apprentissage et 20 % pour le test, un total de 1048 images a été utilisé pour l’entraînement du modèle, tandis que 466 images ont été réservées à l’évaluation pour chacun des types d’images obtenues par transformée en ondelettes. Les performances du modèle ont été évaluées à l’aide de différents

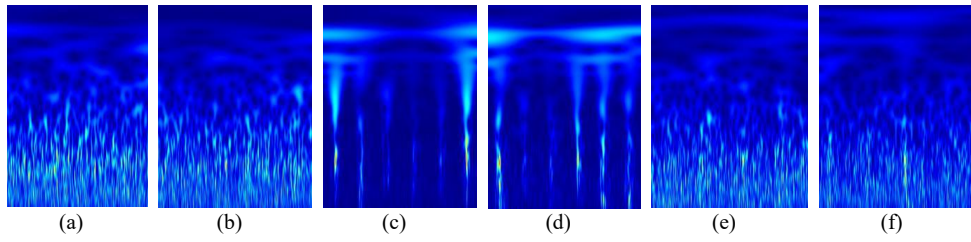


Figure 5.7: Scalogrammes obtenus à l'aide de l'ondelette Morlet : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.

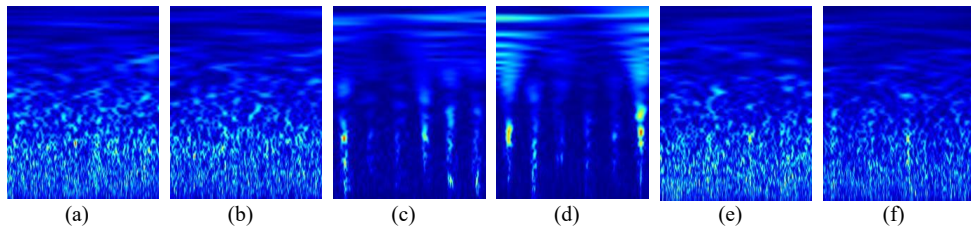


Figure 5.8: Scalogrammes obtenus à l'aide de l'ondelette Bump : (a)-(b) état sain, (c)-(d) défaut de la bague intérieure et (e)-(f) défaut de la bague extérieure.

indicateurs, notamment la précision (*Accuracy*) et la fonction de perte (*Loss*). Les résultats correspondants sont présentés dans les Figures 5.9, 5.10 et 5.11 pour les trois types d'ondelettes considérés. Par ailleurs, afin d'obtenir une meilleure compréhension des échantillons correctement et incorrectement classifiés, des matrices de confusion ont été utilisées. Celles-ci sont présentées à la Figure 5.12. Cette figure comporte trois sous-figures illustrant les résultats obtenus pour les différents types d'ondelettes : la sous-figure (a) correspond à l'ondelette Morse, la sous-figure (b) à l'ondelette Morlet, tandis que la sous-figure (c) présente les résultats obtenus avec l'ondelette Bump. Ces représentations fournissent une analyse détaillée des performances du modèle en mettant en évidence les classifications correctes ainsi que les erreurs de classification pour chaque type d'image temps-fréquence. Dans le domaine de l'apprentissage profond, la précision (*Accuracy*) et la perte (*Loss*) constituent deux indicateurs essentiels pour évaluer les performances d'un modèle au cours des phases d'entraînement et de test. La fonction de perte mesure l'écart entre les valeurs prédites par le modèle et les valeurs réelles, tandis que la précision indique la proportion d'échantillons correctement classifiés. La précision d'un modèle d'apprentissage profond est généralement exprimée sous forme de pourcentage. Une valeur élevée de précision traduit une meilleure capacité du modèle à reconnaître correctement les différentes classes. Par exemple, une précision de 90 % signifie que le modèle a correctement identifié la classe de 9 échantillons sur 10.

D'autre part, la fonction de perte (*Loss*) d'un modèle d'apprentissage profond indique dans quelle mesure les valeurs prédites par le modèle correspondent aux valeurs réelles. Elle est généralement exprimée sous la forme d'une valeur numérique, où des valeurs plus faibles traduisent de meilleures performances. La fonction de perte, qui calcule l'écart entre les valeurs prédites et les valeurs réelles, est définie en fonction de la tâche considérée et de la nature des données. L'objectif principal de l'entraînement d'un modèle d'apprentissage profond est de minimiser cette fonction de perte.

Les courbes de précision (*Accuracy*) et de perte (*Loss*) sont utilisées pour suivre l'évolution des performances du modèle au cours de l'apprentissage et pour déterminer si des ajustements sont nécessaires. Par exemple, lorsque la précision est faible et que la perte est élevée, il peut

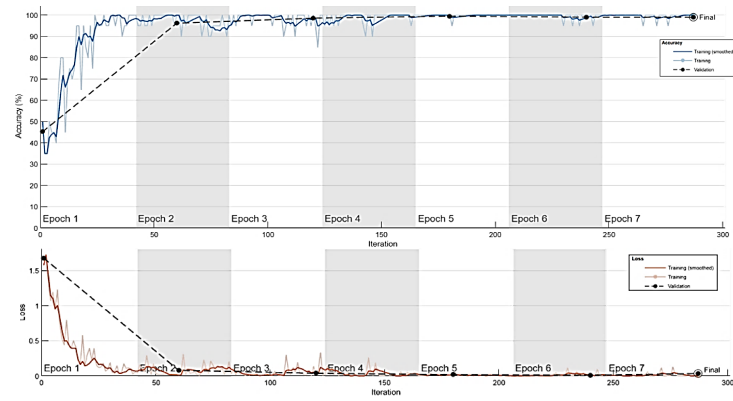


Figure 5.9: Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Morse.

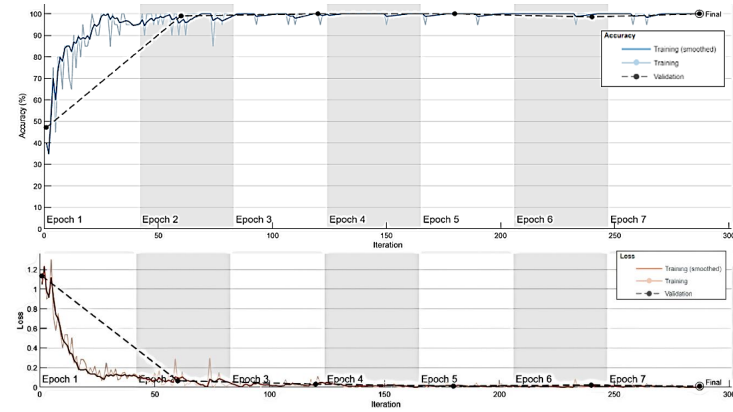


Figure 5.10: Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Morlet.

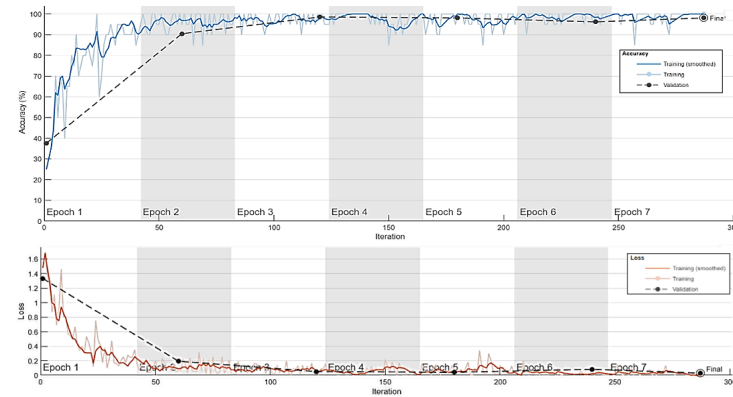


Figure 5.11: Évaluation des performances du modèle à l'aide de la précision (Accuracy) et de la fonction de perte (Loss) pour les scalogrammes obtenus avec l'ondelette Bump.

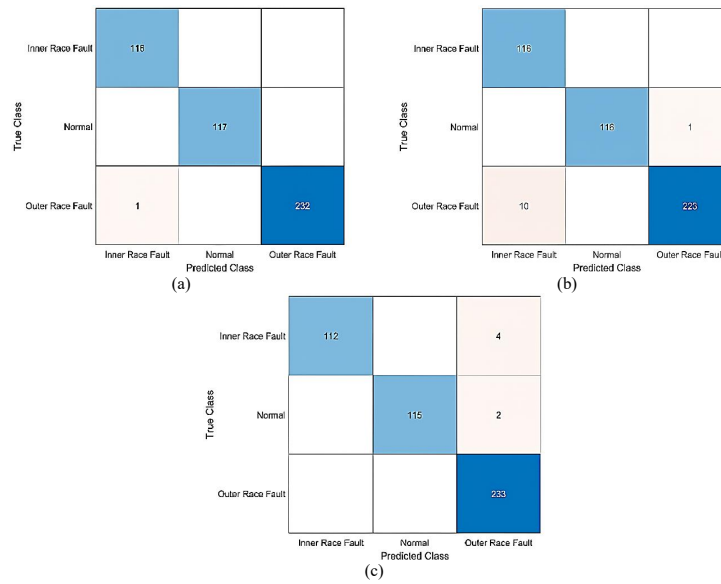


Figure 5.12: Matrices de confusion du modèle CNN pour les différents types d'ondelettes. (a) Ondelette Morse ; (b) ondelette Morlet ; (c) ondelette Bump.

être nécessaire de modifier l'architecture du réseau ou d'ajuster certains hyperparamètres. À l'inverse, lorsque la précision est élevée et que la perte est faible, le modèle présente de bonnes performances et peut être utilisé pour effectuer des prédictions sur de nouvelles données. Ainsi, les indicateurs de précision et de perte fournissent des informations essentielles sur la qualité d'un modèle d'apprentissage profond et contribuent à orienter son développement vers une meilleure efficacité et une plus grande précision.

Une matrice de confusion est un tableau permettant d'évaluer les performances d'un modèle de classification en comparant les prédictions du modèle aux classes réelles des données de test. Elle indique non seulement le nombre de prédictions correctes et incorrectes réalisées par le modèle, mais également la répartition des échantillons appartenant à chaque classe réelle.

La précision (*Accuracy*) est définie comme le rapport entre le nombre de prédictions correctes et le nombre total de prédictions effectuées par le modèle. Elle mesure donc la fréquence à laquelle le modèle attribue correctement la classe correspondante aux données d'entrée.

À partir des valeurs de précision obtenues, il est possible de comparer les performances des trois modèles de classification utilisant les ondelettes Morse, Morlet et Bump, et d'en tirer les conclusions suivantes :

- Le modèle basé sur l'ondelette **Morse** présente la meilleure performance avec une précision de **99,79 %**.
- Le modèle utilisant l'ondelette **Morlet** obtient une précision de **97,64 %**.
- Le modèle fondé sur l'ondelette **Bump** atteint une précision de **98,71 %**.
- Par conséquent, le modèle utilisant l'ondelette **Morse** apparaît comme le plus performant parmi les trois méthodes évaluées.

5.4 Application de la MCSA et DNN au diagnostic des défauts de roulements

5.4.1 Méthodologie

Dans cette étude expérimentale, nous effectuons le diagnostic d'un moteur asynchrone afin de détecter d'éventuelles défaillances. Notre approche repose sur l'analyse du courant monophasé du moteur pour constituer une base de données qui sera ensuite utilisée pour l'entraînement de notre modèle de réseau de neurones profond (DNN). La figure 5.13 présente une vue d'ensemble de la méthodologie adoptée.

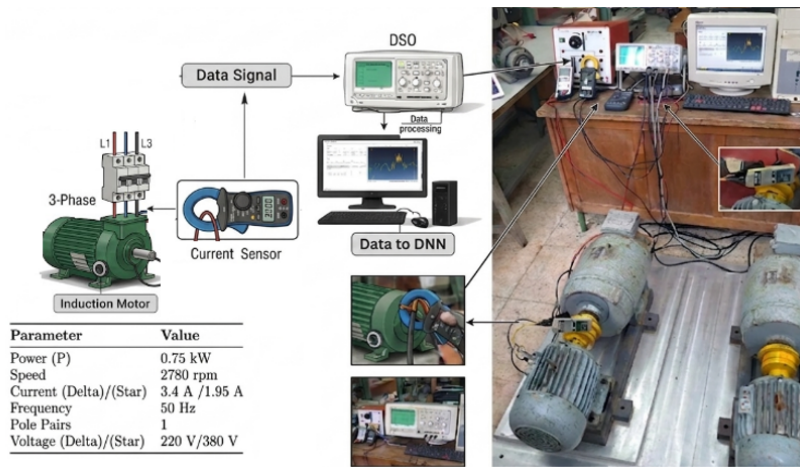


Figure 5.13: Schéma du banc d'essai expérimental

Afin d'acquérir des données précises et représentatives du fonctionnement d'un moteur électrique dans une large gamme de conditions d'exploitation, des mesures en temps réel du courant électrique sur une phase du moteur ont été effectuées. Les signaux électriques ont été enregistrés pour différents états de fonctionnement du moteur, comprenant un état sain ainsi que plusieurs états défectueux correspondant à des défauts de roulements volontairement introduits. La base de données est constituée de cinq classes, comme illustré dans le tableau 5.3, correspondant aux signaux de courant acquis sur une seule phase d'un moteur asynchrone. Chaque signal a été échantillonné à une fréquence de 12 kHz puis segmenté en séquences de longueur fixe de 4000 échantillons. Ces segments représentent cinq conditions de roulement : *état sain*, *défaut de bille*, *défaut de bague intérieure*, *défaut de bague extérieure* et *défaut de cage*. Les données ont été collectées sous différentes conditions de charge, notamment à vide, sous une charge de 100 W, 200 W et 300 W, comme illustré respectivement par les figures 5.14, 5.15, 5.16 et 5.17.

Les données brutes ont ensuite été prétraitées et subdivisées en 400 segments pour chaque classe.

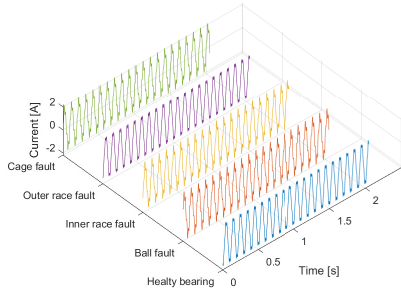


Figure 5.14: Sans charge

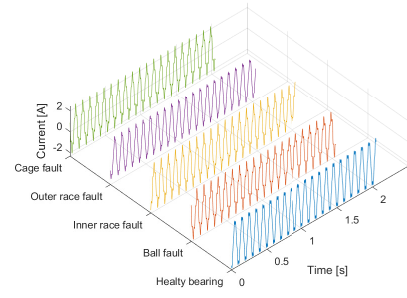


Figure 5.15: Charge de 100 W

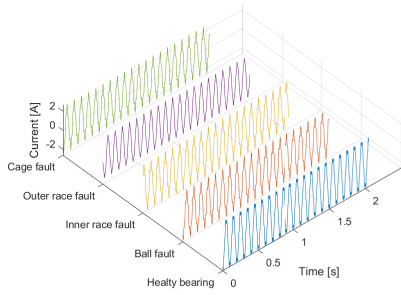


Figure 5.16: Charge de 200 W

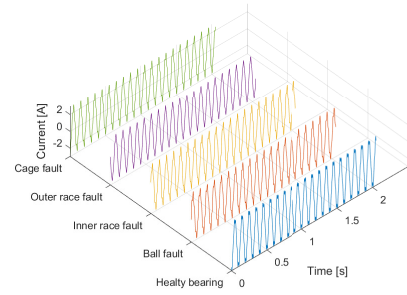


Figure 5.17: Charge de 300 W

Table 5.3: Structure de la base de données

Condition de charge	À vide	100 W	200 W	300 W
Classe 1	État sain	État sain	État sain	État sain
Classe 2	Défaut de bague intérieure	Défaut de bague intérieure	Défaut de bague intérieure	Défaut de bague intérieure
Classe 3	Défaut de bague extérieure	Défaut de bague extérieure	Défaut de bague extérieure	Défaut de bague extérieure
Classe 4	Défaut de cage	Défaut de cage	Défaut de cage	Défaut de cage
Classe 5	Défaut de bille	Défaut de bille	Défaut de bille	Défaut de bille

L'application **Deep Network Designer** de MATLAB R2020b a été utilisée pour concevoir et implémenter l'architecture DNN proposée. La configuration du réseau a été définie à la suite d'un ajustement empirique approfondi de ses hyperparamètres. Plus précisément, le nombre de neurones dans les couches cachées a été fixé à 70, le taux d'apprentissage à 0,001, le nombre d'époques à 300 et la taille des mini-lots (*batch size*) à 256.

La sélection de ces hyperparamètres s'est appuyée sur une procédure systématique par essais et erreurs, au cours de laquelle plusieurs combinaisons ont été évaluées et comparées en termes de précision de classification et de capacité de généralisation sur les données de validation. La configuration finale a été retenue car elle a permis d'obtenir le meilleur compromis entre une précision élevée et une convergence stable, tout en évitant le surapprentissage (*overfitting*).

Les données d'entrée du modèle DNN sont constituées de dix caractéristiques temporelles extraites des segments à l'aide de l'application **Diagnostic Feature Designer**. Ces caractéristiques comprennent la moyenne (*Mean*), la valeur efficace (*Root Mean Square, RMS*), le facteur de forme (*Shape Factor*), le kurtosis, l'asymétrie (*Skewness*), la valeur de crête (*Peak Value*), le facteur d'impulsion (*Impulse Factor*), le facteur de crête (*Crest Factor*), le facteur de jeu (*Clearance Factor*) et l'écart-type (*Standard Deviation, Std*).

Afin d'améliorer l'interprétabilité des résultats, les diagrammes en boîtes (*boxplots*) de trois caractéristiques représentatives, à savoir la valeur efficace (RMS), l'écart-type (Std) et la moyenne,

sont présentés dans la figure 5.18. Ces caractéristiques ont été sélectionnées à titre d'exemple parmi l'ensemble des caractéristiques extraites.

Comme l'illustre la figure 5.18, les caractéristiques RMS et Std présentent une séparation nette entre les conditions saines et défectueuses, ce qui témoigne de leur forte capacité discriminante. En revanche, la caractéristique de moyenne montre une différence plus modérée entre les différentes classes.

Les données ont été réparties en trois ensembles : 70 % pour l'apprentissage, 15 % pour la

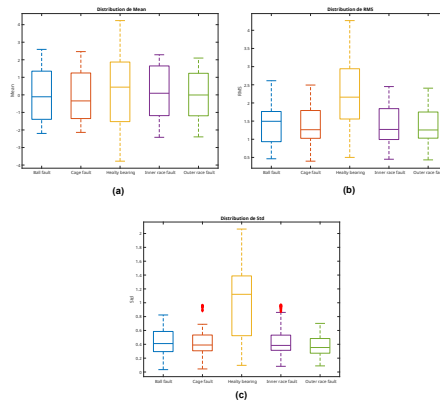


Figure 5.18: Distribution des caractéristiques : (a) moyenne, (b) écart-type (Std) et (c) valeur efficace (RMS) pour les conditions saines et défectueuses.

validation et 15 % pour les tests. Le modèle DNN a été conçu avec cinq neurones de sortie correspondant aux différentes conditions de fonctionnement des roulements. L'architecture du modèle ainsi que les paramètres retenus sont présentés dans le tableau 5.4. L'algorithme d'entraînement sélectionné est "Adam".

La figure 5.19 illustre l'ensemble des étapes précédemment décrites. Elle montre le processus de

Table 5.4: Architecture du réseau de neurones profond (DNN)

Nom	Type	Taille de sortie	Paramètres
Entrée	Couche d'entrée de caractéristiques	10	-
FC.1	Couche entièrement connectée	70	Poids 70×10 , biais 70×1
BatchNorm.1	Normalisation par lots	70	Décalage 70×1 , échelle 70×1
ReLU.1	Unité linéaire rectifiée (ReLU)	70	-
FC.2	Couche entièrement connectée	70	Poids 70×70 , biais 70×1
BatchNorm.2	Normalisation par lots	70	Décalage 70×1 , échelle 70×1
ReLU.2	Unité linéaire rectifiée (ReLU)	70	-
FC.3	Couche entièrement connectée	5	Poids 5×70 , biais 5×1
Softmax	Couche Softmax	5	-
Sortie	Couche de classification	-	-

division des données, leur prétraitement ainsi que leur utilisation comme données d'entrée du modèle DNN pour la phase d'apprentissage.

Afin d'évaluer les performances réelles du modèle proposé, les données de test ont été soumises au modèle DNN entraîné. Les résultats obtenus ont montré une précision de classification de 96,1 %. La figure 5.20 présente la matrice de confusion correspondante, confirmant la robustesse du modèle pour la classification des défauts de roulements. Afin d'effectuer une analyse comparative, le même jeu de données a également été utilisé pour entraîner d'autres modèles

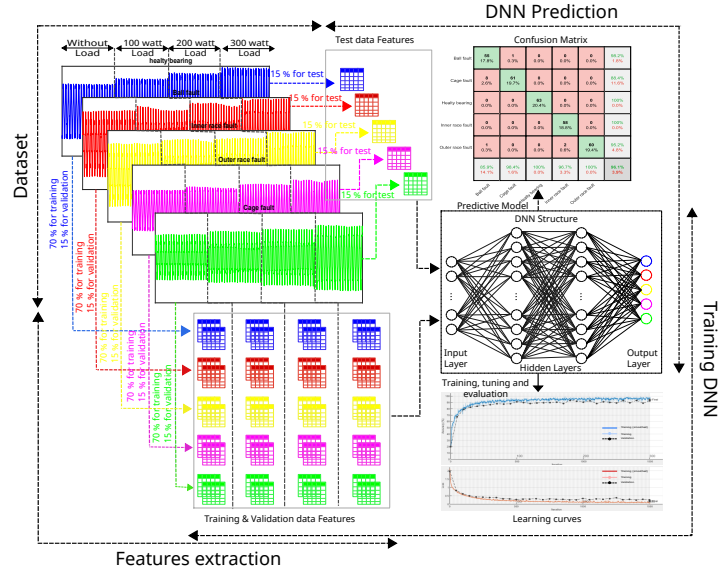


Figure 5.19: Système de diagnostic des défauts de roulements basé sur un réseau DNN.

d'apprentissage automatique et d'apprentissage profond, notamment les k plus proches voisins (KNN), les machines à vecteurs de support (SVM), les arbres de décision (DT), les réseaux de neurones convolutifs (CNN) et les réseaux à mémoire longue à court terme (LSTM).

Les modèles KNN, SVM et DT ont été évalués à l'aide du même ensemble de test afin de comparer leurs performances prédictives à celles du modèle DNN. En revanche, pour les modèles CNN et LSTM, chaque signal a été subdivisé en 160 segments par classe.

Par ailleurs, les segments de signaux unidimensionnels (1D) ont été transformés en représentations bidimensionnelles (2D) sous forme de scalogrammes à l'aide de la transformée en ondelettes continue (*Continuous Wavelet Transform, CWT*) avec l'ondelette de Morse, conformément à l'approche proposée dans [17]. Ces représentations ont ensuite été utilisées comme données d'entrée du modèle CNN. Enfin, les segments de signaux bruts ont été directement injectés dans le réseau LSTM afin d'exploiter les dépendances temporelles intrinsèques aux données.

Les figures 5.21, 5.22, 5.23, 5.26 et 5.27 présentent respectivement les matrices de confusion obtenues pour les modèles KNN, SVM, DT, CNN et LSTM. Les résultats quantitatifs correspondants sont résumés dans le tableau 5.5.

Les résultats de cette étude comparative montrent que l'architecture DNN est plus efficace

Table 5.5: Comparaison détaillée des différentes méthodes

Méthode	Précision globale (%)	Précision (%)	Rappel (%)	Score F1 (%)
DNN	96,12	96,37	96,20	96,13
k -NN	93,20	93,82	92,94	92,62
SVM	91,59	91,66	91,34	91,29
DT	89,32	88,94	89,01	88,95
CNN	93,75	95,24	93,75	93,59
LSTM	94,38	95,61	94,38	94,26

pour capturer les caractéristiques complexes associées à l'état de santé des roulements. Cette

Ball fault	55 17.8%	1 0.3%	0 0.0%	0 0.0%	0 0.0%	98.2% 1.8%
Cage fault	8 2.6%	61 19.7%	0 0.0%	0 0.0%	0 0.0%	88.4% 11.6%
Healthy bearing	0 0.0%	0 0.0%	63 20.4%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	58 18.8%	0 0.0%	100% 0.0%
Outer race fault	1 0.3%	0 0.0%	0 0.0%	2 0.6%	60 19.4%	95.2% 4.8%
	85.9% 14.1%	98.4% 1.6%	100% 0.0%	96.7% 3.3%	100% 0.0%	96.1% 3.9%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.20: Matrice de confusion de la méthode DNN.

Ball fault	41 13.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	14 4.5%	54 17.5%	0 0.0%	0 0.0%	0 0.0%	79.4% 20.6%
Healthy bearing	0 0.0%	0 0.0%	61 19.7%	0 0.0%	1 0.3%	98.4% 1.6%
Inner race fault	2 0.6%	0 0.0%	0 0.0%	67 21.7%	0 0.0%	97.1% 2.9%
Outer race fault	2 0.6%	1 0.3%	0 0.0%	1 0.3%	65 21.0%	94.2% 5.8%
	69.5% 30.5%	98.2% 1.8%	100% 0.0%	98.5% 1.5%	98.5% 1.5%	93.2% 6.8%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.21: Matrice de confusion de la méthode KNN.

Ball fault	48 15.5%	3 1.0%	0 0.0%	2 0.6%	0 0.0%	90.6% 9.4%
Cage fault	11 3.6%	51 16.5%	0 0.0%	0 0.0%	0 0.0%	82.3% 17.7%
Healthy bearing	0 0.0%	1 0.3%	54 17.5%	0 0.0%	1 0.3%	96.4% 3.6%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	65 21.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	7 2.3%	1 0.3%	65 21.0%	89.0% 11.0%
	81.4% 18.6%	92.7% 7.3%	88.5% 11.5%	95.6% 4.4%	98.5% 1.5%	91.6% 8.4%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.22: Matrice de confusion de la méthode SVM.

Ball fault	48 15.5%	5 1.9%	1 0.3%	1 0.3%	2 0.6%	82.8% 17.2%
Cage fault	8 2.6%	45 14.6%	0 0.0%	1 0.3%	2 0.6%	80.4% 19.6%
Healthy bearing	2 0.6%	0 0.0%	60 19.4%	0 0.0%	2 0.6%	93.8% 6.3%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	64 20.7%	1 0.3%	98.5% 1.5%
Outer race fault	1 0.3%	4 1.3%	0 0.0%	2 0.6%	59 19.1%	89.4% 10.6%
	81.4% 18.6%	81.8% 18.2%	98.4% 1.6%	94.1% 5.9%	89.4% 10.6%	89.3% 10.7%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.23: Matrice de confusion de la méthode DT.

Ball fault	32 20.0%	10 6.3%	0 0.0%	0 0.0%	0 0.0%	76.2% 23.8%
Cage fault	0 0.0%	22 13.8%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Healthy bearing	0 0.0%	0 0.0%	32 20.0%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	32 20.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	0 0.0%	0 0.0%	32 20.0%	100% 0.0%
	100% 0.0%	68.8% 31.3%	100% 0.0%	100% 0.0%	100% 0.0%	93.8% 6.3%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.24: Matrice de confusion de la méthode CNN.

Ball fault	23 14.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	9 5.6%	32 20.0%	0 0.0%	0 0.0%	0 0.0%	78.0% 22.0%
Healthy bearing	0 0.0%	0 0.0%	32 20.0%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	32 20.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	0 0.0%	0 0.0%	32 20.0%	100% 0.0%
	71.9% 28.1%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	94.4% 5.6%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.25: Matrice de confusion de la méthode LSTM.

capacité lui permet d'atteindre des performances supérieures en matière de diagnostic, comme en témoignent les valeurs les plus élevées obtenues pour l'ensemble des indicateurs d'évaluation considérés. Le modèle DNN présente notamment une précision globale de 96,12 %, surpassant les

approches KNN, SVM, DT, CNN et LSTM, ce qui confirme son aptitude à fournir un diagnostic fiable et précis des défauts de roulements.

5.4.2 Méthode proposée

Lorsqu'un défaut survient dans le moteur, une manifestation courante est l'augmentation de l'amplitude du courant de phase. Cependant, des variations similaires peuvent également résulter de changements normaux de charge, ce qui peut rendre la caractéristique du courant maximal potentiellement trompeuse pour l'identification des défauts.

Afin d'améliorer la robustesse du modèle DNN, une normalisation basée sur des coefficients est introduite pour prendre en compte les différents niveaux de charge en régime permanent. Les coefficients C_1 , C_2 , C_3 et C_4 sont respectivement attribués aux conditions de fonctionnement à vide, sous 100 W, 200 W et 300 W, permettant ainsi au modèle de distinguer les signatures liées aux défauts des variations induites par la charge.

En supposant l'existence d'une relation monotone entre la charge et le courant maximal dans des conditions de fonctionnement saines, une normalisation de type Min-Max est appliquée [35] afin de définir les coefficients comme suit :

$$C_i = 1 + (k - 1) \frac{I_{\max,i} - I_{\min}}{I_{\max} - I_{\min}} \quad (5.3)$$

où $I_{\min}$ et $I_{\max}$ représentent respectivement les valeurs minimale et maximale du courant dans des conditions de fonctionnement sans défaut. Cette formulation permet de contraindre les coefficients dans l'intervalle $[1, k]$ et garantit une mise à l'échelle cohérente entre les différents niveaux de charge.

Le paramètre k contrôle la sensibilité du modèle : de faibles valeurs réduisent l'effet de la normalisation, tandis que des valeurs plus élevées améliorent la séparation entre les niveaux de charge, mais peuvent également amplifier le bruit. Une analyse de sensibilité montre que des valeurs modérées de k offrent le meilleur compromis entre robustesse et capacité de discrimination. Une étude d'ablation confirme davantage l'efficacité de l'approche proposée en démontrant une amélioration des performances de classification lorsque les coefficients sont intégrés au modèle. La figure 5.26 illustre la méthodologie globale proposée.

Après l'entraînement hors ligne du modèle DNN, une stratégie de prédiction en ligne a été développée. Lors de l'acquisition de nouveaux échantillons, ceux-ci sont multipliés par le coefficient correspondant à la charge de fonctionnement actuelle. Cette étape permet d'assurer la cohérence entre les données d'entrée et les conditions de charge prises en compte lors de l'apprentissage du modèle. La figure 5.27 présente les différentes étapes du modèle de prédiction en ligne proposé.

5.4.3 Résultats expérimentaux

Les résultats obtenus à l'aide de la méthode proposée sont très satisfaisants. La figure 5.28 présente la matrice de confusion obtenue, confirmant l'efficacité de l'approche proposée, avec une identification correcte des différentes classes. La précision globale du modèle est passée de 96,1 % à 99,0 % en utilisant les mêmes données d'apprentissage et de test, garantissant ainsi une comparaison équitable des performances avant et après l'application de la méthode proposée. De plus, la courbe d'apprentissage a connu une amélioration significative, témoignant d'une meilleure adaptation du modèle aux données d'apprentissage prétraitées.

La méthode proposée a également été appliquée aux modèles KNN, SVM, DT, CNN et LSTM. Les résultats obtenus, comparés à ceux des méthodes conventionnelles, se sont révélés nettement plus satisfaisants.

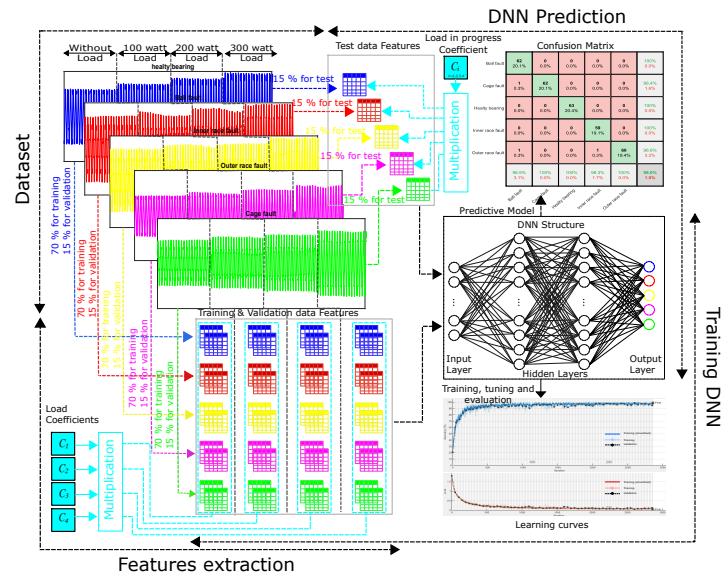


Figure 5.26: Architecture de la méthode proposée

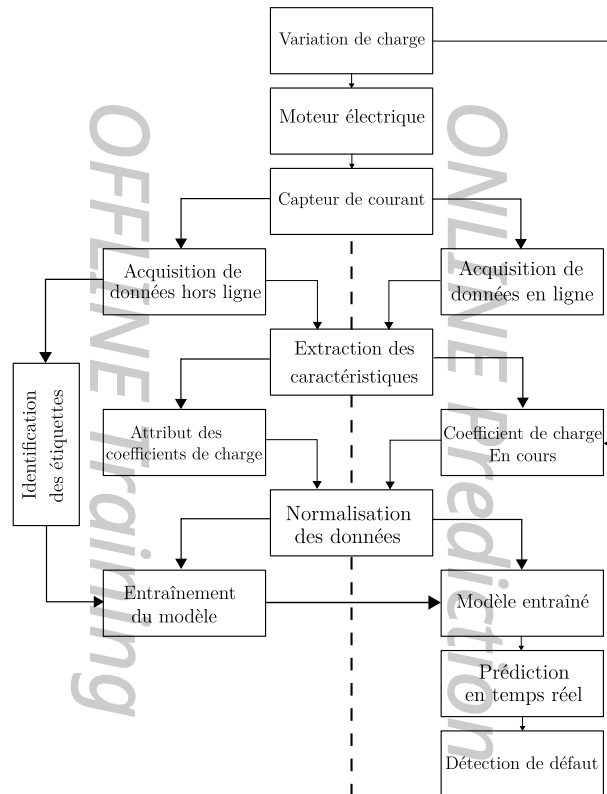


Figure 5.27: Organigramme de la méthode proposée pour le diagnostic en ligne d'un moteur électrique

Pour la méthode KNN (figure 5.29), la précision est passée de 93,2 % à 98,7 %. Concernant la méthode SVM (figure 5.30), la précision a été améliorée de 91,6 % à 96,8 %. De même, pour l'arbre de décision (DT) (figure 5.31), la précision a augmenté de 89,3 % à 97,7 %.

Ball fault	62 20.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	1 0.3%	62 20.1%	0 0.0%	0 0.0%	0 0.0%	96.4% 1.6%
Healthy bearing	0 0.0%	0 0.0%	63 20.4%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	59 19.1%	0 0.0%	100% 0.0%
Outer race fault	1 0.3%	0 0.0%	0 0.0%	1 0.3%	60 19.4%	96.8% 3.2%
	96.9% 3.1%	100% 0.0%	100% 0.0%	96.3% 1.7%	100% 0.0%	99.0% 1.0%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.28: Matrice de confusion de la méthode DNN proposé.

Ball fault	57 18.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	1 0.3%	55 17.8%	1 0.3%	0 0.0%	0 0.0%	96.5% 3.5%
Healthy bearing	1 0.3%	0 0.0%	60 19.4%	0 0.0%	1 0.3%	96.8% 3.2%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	68 22.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	0 0.0%	0 0.0%	65 21.0%	100% 0.0%
	96.6% 3.4%	100% 0.0%	98.4% 1.6%	100% 0.0%	98.5% 1.5%	98.7% 1.3%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.29: Matrice de confusion de la méthode KNN proposé.

Ball fault	53 17.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	0 0.0%	54 17.5%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Healthy bearing	4 1.3%	1 0.3%	59 19.1%	0 0.0%	1 0.3%	90.8% 9.2%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	68 22.0%	0 0.0%	100% 0.0%
Outer race fault	2 0.6%	0 0.0%	2 0.6%	0 0.0%	65 21.0%	94.2% 5.8%
	89.8% 10.2%	98.2% 1.8%	95.7% 3.3%	100% 0.0%	98.5% 1.5%	96.8% 3.2%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.30: Matrice de confusion de la méthode SVM proposé.

Ball fault	57 18.4%	1 0.3%	2 0.6%	0 0.0%	0 0.0%	95.0% 5.0%
Cage fault	0 0.0%	53 17.2%	0 0.0%	0 0.0%	1 0.3%	98.1% 1.9%
Healthy bearing	2 0.6%	0 0.0%	59 19.1%	0 0.0%	0 0.0%	96.7% 3.3%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	68 22.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	1 0.3%	0 0.0%	0 0.0%	65 21.0%	98.5% 1.5%
	96.6% 3.4%	96.4% 3.6%	96.7% 3.3%	100% 0.0%	98.5% 1.5%	97.7% 2.3%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.31: Matrice de confusion de la méthode DT proposé.

Ball fault	29 18.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	3 1.9%	32 20.0%	0 0.0%	0 0.0%	0 0.0%	91.4% 8.6%
Healthy bearing	0 0.0%	0 0.0%	32 20.0%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	32 20.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	0 0.0%	0 0.0%	32 20.0%	100% 0.0%
	90.6% 9.4%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	98.1% 1.9%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.32: Matrice de confusion de la méthode CNN proposé.

Ball fault	30 18.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Cage fault	2 1.3%	32 20.0%	0 0.0%	0 0.0%	0 0.0%	94.1% 5.9%
Healthy bearing	0 0.0%	0 0.0%	32 20.0%	0 0.0%	0 0.0%	100% 0.0%
Inner race fault	0 0.0%	0 0.0%	0 0.0%	32 20.0%	0 0.0%	100% 0.0%
Outer race fault	0 0.0%	0 0.0%	0 0.0%	0 0.0%	32 20.0%	100% 0.0%
	93.8% 6.3%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	98.8% 1.2%
	Ball fault	Cage fault	Healthy bearing	Inner race fault	Outer race fault	

Figure 5.33: Matrice de confusion de la méthode LSTM proposé.

Pour le modèle CNN (figure 5.32), la précision est passée de 93,8 % à 98,1 %, tandis que pour le modèle LSTM (figure 5.33), elle a progressé de 94,4 % à 98,8 %.

La figure 5.34 présente une comparaison des performances des différentes méthodes avant et

après l'application de l'approche proposée. Le tableau 5.6 fournit une analyse plus approfondie ainsi qu'une évaluation comparative détaillée des performances obtenues.

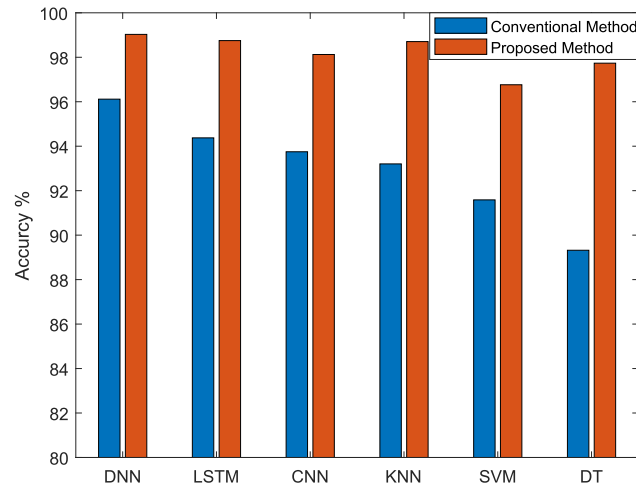


Figure 5.34: Comparaison des méthodes conventionnelles et les méthodes proposé

Table 5.6: Évaluation comparative entre les différentes méthodes

Méthode	Précision globale (%)	Précision (%)	Rappel (%)	Score F1 (%)
DNN	96,12	96,37	96,20	96,13
DNN (Proposée)	99,03	99,04	99,04	99,03
k -NN	93,20	93,82	92,94	92,62
k -NN (Proposée)	98,71	98,65	98,69	98,66
SVM	91,59	91,66	91,34	91,29
SVM (Proposée)	96,76	96,99	96,64	96,73
DT	89,32	88,94	89,01	88,95
DT (Proposée)	97,73	97,67	97,64	97,65
CNN	93,75	95,24	93,75	93,59
CNN (Proposée)	98,13	98,29	98,13	98,12
LSTM	94,38	95,61	94,38	94,26
LSTM (Proposée)	98,75	98,82	98,75	98,75

5.5 Conclusion

Dans ce chapitre, les différentes méthodes de diagnostic présentées au cours de cette thèse ont été appliquées et validées expérimentalement sur des systèmes réels dans un environnement de laboratoire. Les résultats obtenus ont permis d'évaluer leurs performances dans la détection et l'identification des défauts mécaniques des moteurs électriques. L'étude basée sur l'analyse vibratoire a démontré l'intérêt des techniques conventionnelles de traitement du signal pour la détection de défauts mécaniques combinés de déséquilibre et de desserrage, grâce à leur simplicité de mise en œuvre et à leur caractère interprétable. Les approches fondées sur l'intelligence

artificielle, notamment les architectures CNN et DNN associées respectivement à la transformée en ondelettes continue (CWT) et à l'analyse de la signature du courant moteur (MCSA) pour le diagnostic des défauts de roulements, ont quant à elles montré une capacité supérieure à exploiter des signaux complexes et à améliorer les performances de classification des défauts.

L'analyse comparative réalisée met en évidence que les méthodes de traitement du signal constituent des outils robustes et efficaces lorsque les caractéristiques des défauts sont clairement identifiables, tout en nécessitant une expertise pour l'interprétation des résultats. À l'inverse, les approches basées sur l'apprentissage automatique et l'apprentissage profond offrent un niveau d'automatisation plus élevé et une meilleure capacité de généralisation, au prix d'un besoin accru en données d'apprentissage et en ressources de calcul. Parmi les méthodes étudiées, les réseaux de neurones profonds (DNN) se sont révélés les plus performants pour le diagnostic des défauts de roulements à partir des signaux de courant.

Ces travaux confirment ainsi l'intérêt de combiner les techniques avancées de traitement du signal et les méthodes d'intelligence artificielle pour le développement de systèmes de diagnostic fiables et performants. Ils ouvrent également des perspectives prometteuses pour la mise en œuvre de solutions de surveillance conditionnelle et de maintenance prédictive dans les systèmes industriels modernes.

Conclusion générale et Perspectives

Cette thèse s'inscrit dans le cadre du diagnostic des défauts des machines électriques, un domaine de recherche essentiel pour améliorer la fiabilité, la disponibilité et la sécurité des systèmes industriels. Face aux enjeux croissants de la maintenance prédictive, l'objectif principal de ce travail a été d'étudier, développer et évaluer différentes approches de diagnostic reposant sur le traitement du signal et l'intelligence artificielle.

Dans un premier temps, les principaux défauts susceptibles d'affecter les machines électriques ainsi que leurs mécanismes d'apparition et leurs conséquences sur le fonctionnement des systèmes ont été présentés. Une étude approfondie des techniques de diagnostic basées sur le traitement du signal a ensuite été réalisée, en s'intéressant aux méthodes d'analyse temporelle, fréquentielle et temps-fréquence. Ces approches permettent d'extraire des informations pertinentes à partir des signaux de fonctionnement et constituent une étape fondamentale dans la détection des anomalies.

Par la suite, les méthodes de diagnostic fondées sur l'apprentissage automatique ont été étudiées. Les principes de l'intelligence artificielle ainsi que les algorithmes de classification les plus utilisés dans le diagnostic des machines électriques ont été présentés. Cette étude a été complétée par l'analyse des techniques d'apprentissage profond, notamment les réseaux de neurones profonds (DNN), les réseaux de neurones convolutifs (CNN) et les réseaux à mémoire longue à court terme (LSTM), qui offrent de nouvelles possibilités pour l'exploitation automatique des données issues des systèmes surveillés.

La dernière partie de cette thèse a été consacrée à la validation expérimentale des différentes approches étudiées. Les travaux réalisés ont permis d'appliquer les méthodes de traitement du signal et d'intelligence artificielle à plusieurs cas d'étude réels. Les résultats obtenus ont montré l'efficacité de l'analyse vibratoire pour la détection de défauts mécaniques combinés dans les systèmes éoliens. Ils ont également mis en évidence les performances des approches basées sur la transformée en ondelettes continue associée à un réseau CNN ainsi que celles fondées sur l'analyse de la signature du courant moteur couplée à un réseau DNN pour le diagnostic des défauts de roulements des moteurs à induction. L'analyse comparative effectuée a permis de souligner les avantages et les limites de chaque famille de méthodes. Les techniques de traitement du signal se distinguent par leur simplicité, leur interprétabilité et leur robustesse, tandis que les approches basées sur l'intelligence artificielle offrent des capacités supérieures d'automatisation et de classification des défauts complexes. Parmi les méthodes étudiées, les réseaux de neurones profonds ont montré les meilleures performances pour le diagnostic des défauts de roulements à partir des signaux de courant.

Les contributions de cette thèse ont donné lieu à plusieurs publications scientifiques, confirmant l'intérêt des approches proposées et leur apport au domaine du diagnostic des machines électriques.

Les résultats obtenus démontrent que la combinaison des techniques avancées de traitement du signal et des méthodes d'intelligence artificielle constitue une voie prometteuse pour le développement de systèmes de surveillance conditionnelle performants.

Comme perspectives de recherche, il serait intéressant d'étendre les travaux réalisés à d'autres types de défauts et à des conditions de fonctionnement plus variées. L'intégration de modèles d'apprentissage profond plus récents, l'exploitation de données multi-capteurs ainsi que le développement de systèmes de diagnostic embarqués capables de fonctionner en temps réel constituent également des pistes prometteuses pour renforcer les performances et la fiabilité des systèmes de maintenance prédictive dans les environnements industriels du futur.

Bibliographie

- [1] B. Abdelkarim. Techniques de detection des defaillances. Cours. COURS DE TECHNIQUES DE SURVEILLANCE.
- [2] B. Abdelkarim. PhD thesis, Université Badji Mokhtar Annaba, 2015.
- [3] S. Altug, M.-Y. Chen, and H. J. Trussell. Fuzzy inference systems implemented on neural architectures for motor fault detection and diagnosis. *IEEE transactions on industrial electronics*, 46(6):1069–1079, 1999.
- [4] G. Andria, E. D’ambrosio, M. Savino, and A. Trotta. Application of wigner-ville distribution to measurements on transient signals. In *1993 IEEE Instrumentation and Measurement Technology Conference*, pages 612–617. IEEE, 1993.
- [5] J. Antoni. The spectral kurtosis: a useful tool for characterising non-stationary signals. *Mechanical systems and signal processing*, 20(2):282–307, 2006.
- [6] J. Antonino-Daviu, P. Jover, M. Riera, A. Arkkio, and J. Roger-Folch. Dwt analysis of numerical and experimental data for the diagnosis of dynamic eccentricities in induction motors. *Mechanical Systems and Signal Processing*, 21(6):2575–2589, 2007.
- [7] P. Arquès. *Diagnostic prédictif et défaillances des machines: théorie, traitement, analyse, reconnaissance, prédiction*. Editions Technip, 2009.
- [8] M. A. Awadallah and M. M. Morcos. Application of ai tools in fault diagnosis of electrical machines and drives-an overview. *IEEE Transactions on energy conversion*, 18(2):245–251, 2003.
- [9] C.-A. Azencott. *Introduction au Machine Learning-2e éd.* Dunod, 2019.
- [10] N. Baydar and A. Ball. A comparative study of acoustic and vibration signals in detection of gear failures using wigner-ville distribution. *Mechanical systems and signal processing*, 15(6):1091–1107, 2001.
- [11] A. Bellini, F. Filippetti, C. Tassoni, and G.-A. Capolino. Advances in diagnostic techniques for induction machines. *IEEE Transactions on industrial electronics*, 55(12):4109–4126, 2008.
- [12] Y. Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade: Second Edition*, pages 437–478. Springer, 2012.

- [13] M. Blodt, P. Granjon, B. Raison, and G. Rostaing. Models for bearing damage detection in induction motors using stator current monitoring. *IEEE transactions on industrial electronics*, 55(4):1813–1822, 2008.
- [14] M. Blodt, J. Regnier, and J. Faucher. Distinguishing load torque oscillations and eccentricity faults in induction motors using stator current wigner distributions. *IEEE Transactions on Industry Applications*, 45(6):1991–2000, 2009.
- [15] W. Bolton. *Mechatronics: electronic control systems in mechanical and electrical engineering*. Pearson Education, 2003.
- [16] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM review*, 60(2):223–311, 2018.
- [17] R. Boudiaf, B. Abdelkarim, and H. Issam. Bearing fault diagnosis in induction motor using continuous wavelet transform and convolutional neural networks. *International Journal of Power Electronics and Drive Systems*, 15(1):591–602, 2024.
- [18] L. Breiman et al. Classification and regression trees, wadsworth, inc., belmont, california (21) quinlan. *JR (1986). Induction of Decision Trees, Mar chine Learning*, 1:81–106, 1986.
- [19] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and regression trees*. CRC press, 1984.
- [20] M. F. Cabanas, M. G. Melero, J. G. Aleixandre, and J. Solares. Shaft misalignment diagnosis of induction motors using current spectral analysis: A theoretical approach. In *International conference on electric machines, ICEM*, volume 96, 1996.
- [21] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.
- [22] J. CusidÓCusido, L. Romeral, J. A. Ortega, J. A. Rosero, and A. G. Espinosa. Fault detection in induction machines using power spectral density in wavelet decomposition. *IEEE Transactions on Industrial Electronics*, 55(2):633–643, 2008.
- [23] M. Dataset. Society for machinery failure prevention technology. *Accessed: Mar, 28, 2020*.
- [24] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.
- [25] A. Fitzgerald. Ck jr., and sd umans, electric machinery, 2003.
- [26] L. Foued. Etude des principaux défauts. Cours, 2017. COURS DE TECHNIQUES DE SURVEILLANCE.
- [27] K. S. Gaeid and H. W. Ping. Wavelet fault diagnosis and tolerant of induction motor: A review. *International Journal of the Physical Sciences*, 6(3):358–376, 2011.
- [28] A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. ” O’Reilly Media, Inc.”, 2022.
- [29] A. Ghods and H.-h. Lee. A frequency-based approach to detect bearing faults in induction motors using discrete wavelet transform. In *2014 IEEE International Conference on Industrial Technology (ICIT)*, pages 121–125. IEEE, 2014.

- [30] I. Goodfellow. Deep learning, 2016.
- [31] M. Habyarimana and A. A. Adebiyi. A review of artificial intelligence applications in predicting faults in electrical machines. *Energies*, 18(7):1616, 2025.
- [32] K. Hamani, M. Kuchar, M. Kubatko, and S. Kirschner. Advancements in induction motor fault diagnosis and condition monitoring: A comprehensive review. *Sensors (Basel, Switzerland)*, 25(19):5942, 2025.
- [33] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.
- [34] S. Haykin. *Neural networks and learning machines, 3/E*. Pearson education india, 2009.
- [35] H. Henderi, T. Wahyuningsih, and E. Rahwanto. Comparison of min-max normalization and z-score normalization in the k-nearest neighbor (knn) algorithm to test the accuracy of types of breast cancer. *International Journal of Informatics and Information Systems*, 4(1):13–20, 2021.
- [36] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [37] F. K. Hugo Larochelle. Artificial intelligence class - universit  de sherbrooke. 2013.
- [38] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.
- [39] M. E. Iglesias Mart nez, J. A. Antonino-Daviu, L. Dunai, J. A. Conejero, and P. Fern ndez de C rdoba. Higher-order spectral analysis and artificial intelligence for diagnosing faults in electrical machines: An overview. *Mathematics*, 12(24):4032, 2024.
- [40] M. Kang and J.-M. Kim. Reliable fault diagnosis of multiple induction motor defects using a 2-d representation of shannon wavelets. *IEEE Transactions on Magnetics*, 50(10):1–13, 2014.
- [41] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [42] M. P. Kostenko and L. Piotrovski. *Machines  lectriques*. 1979.
- [43] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.
- [44] H. Larochelle. Neural networks class - universit  de sherbrooke. 2013.
- [45] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [46] J. Lee, B. Bagheri, and H.-A. Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing letters*, 3:18–23, 2015.
- [47] R. K. Mobley. *An introduction to predictive maintenance*. Elsevier, 2002.
- [48] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

- [49] S. Nandi, H. A. Toliyat, and X. Li. Condition monitoring and fault diagnosis of electrical motors—a review. *IEEE transactions on energy conversion*, 20(4):719–729, 2005.
- [50] S. Nandi, H. A. Toliyat, and X. Li. Condition monitoring and fault diagnosis of electrical motors—a review. *IEEE transactions on energy conversion*, 20(4):719–729, 2005.
- [51] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. In *Dokl. Akad. Nauk. SSSR*, volume 269, page 543, 1983.
- [52] C. Piché. La machine à courant continu. Cours, 2022. Classe de première SI.
- [53] V. F. Pires, M. Kadivonga, J. Martins, and A. Pires. Motor square current signature analysis for induction motor rotor diagnosis. *Measurement*, 46(2):942–948, 2013.
- [54] J. R. Quinlan. Induction of decision trees. *Machine learning*, 1:81–106, 1986.
- [55] J. R. Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.
- [56] R. B. Randall. *Vibration-based condition monitoring: industrial, automotive and aerospace applications*. John Wiley & Sons, 2021.
- [57] M. Riera-Guasp, J. A. Antonino-Daviu, and G.-A. Capolino. Advances in electrical machine, power electronic, and drive condition monitoring and fault detection: State of the art. *IEEE Transactions on Industrial Electronics*, 62(3):1746–1759, 2014.
- [58] M. Riera-Guasp, M. Pineda-Sánchez, J. Pérez-Cruz, R. Puche-Panadero, J. Roger-Folch, and J. A. Antonino-Daviu. Diagnosis of induction motor faults via gabor analysis of the current in transient regime. *IEEE Transactions on Instrumentation and Measurement*, 61(6):1583–1596, 2012.
- [59] G. Saint-Cirgue. Apprendre le machine learning en une semaine. *Machine Learning, Londres*, page 16, 2019.
- [60] G. Salles. *Surveillance et diagnostic des défauts de la charge d’un entraînement par machine asynchrone*. PhD thesis, Lyon 1, 1997.
- [61] R. R. Schoen, T. G. Habetler, F. Kamran, and R. Bartfield. Motor bearing damage detection using stator current monitoring. *IEEE transactions on industry applications*, 31(6):1274–1279, 1995.
- [62] K. M. Siddiqui and V. Giri. Broken rotor bar fault detection in induction motors using transient current analysis. *International Journal on Electronics & Communication Technology (IJECT)*, 2(4):114–119, 2011.
- [63] J. R. Stack, T. G. Habetler, and R. G. Harley. Experimentally generating faults in rolling element bearings via shaft current. *IEEE transactions on industry applications*, 41(1):25–29, 2005.
- [64] W. Thomson. A review of on-line condition monitoring techniques for three-phase squirrel-cage induction motors-past present and future. In *Keynote address at IEEE symposium on diagnostics for electrical machines, power electronics and drives, Gijon, Spain*, pages 3–18, 1999.
- [65] W. T. Thomson and M. Fenger. Current signature analysis to detect induction motor faults. *IEEE Industry Applications Magazine*, 7(4):26–34, 2001.

- [66] W. T. Thomson and M. Fenger. Case histories of current signature analysis to detect faults in induction motor drives. In *IEEE International Electric Machines and Drives Conference, 2003. IEMDC'03.*, volume 3, pages 1459–1465. IEEE, 2003.
- [67] T. Tieleman, G. Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.
- [68] B. Trajin. *Analyse et traitement de grandeurs électriques pour la détection et le diagnostic de défauts mécaniques dans les entraînements asynchrones. Application à la surveillance des roulements à billes*. PhD thesis, Institut National Polytechnique de Toulouse-INPT, 2009.
- [69] J.-C. Trigeassou. *Diagnostic des machines électriques*. Lavoisier, 2011.
- [70] P. Vas. *Parameter estimation, condition monitoring, and diagnosis of electrical machines*. Oxford University Press, 1992.
- [71] A. Widodo and B.-S. Yang. Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical systems and signal processing*, 21(6):2560–2574, 2007.
- [72] C. Zachariades and V. Xavier. A review of artificial intelligence techniques in fault diagnosis of electric machines. *Sensors (Basel, Switzerland)*, 25(16):5128, 2025.
- [73] W. Zhang et al. A review of the application of deep learning in intelligent fault diagnosis of rotating machinery. *Measurement*, 220:113324, 2023.