



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed

معهد الصيانة والأمن الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département : MAINTENANCE EN ELECTROMECHANIQUE

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : Génie Industriel

Spécialité : Maintenance-Fiabilité-Qualité

Thème

**Gestion de maintenance assistée par ordinateur :
Maintenance préventive d'un processus industriel**

Présenté et soutenu publiquement par :

MEKIDECHE Rami et BERRALEM Abdelkarim

Devant le jury composé de :

Nom et Prénom	Grade	Etablissement	Qualité
HEMMAMI ZINEB	MCB	Université d'Oran 2	Président
ROUAN SERIK Mehdi	MCB	Université d'Oran 2	Encadreur
REGUIEG YSSAAD Sadek	MAA	Université d'Oran 2	Examineur

Année 2019/2020

Résumé

Le travail présenté dans ce mémoire a pour but d'ouvrir les portes sur ce vaste domaine qui est la gestion de la maintenance assistée par ordinateur et son développement en utilisant les réseaux de neurones artificiels, pour prédire le moment de la panne et pouvoir faire la maintenance préventive prédictive

Et faire Implémentation et testé sur des données de moteurs d'avions du centre d'excellence en pronostic de la National Aeronautics and Space Administration (NASA).

Mots clé : gestion maintenance assisté par ordinateur, Réseau de neurone artificiel, maintenance préventive.

Abstract

The work presented in this thesis aims to open the doors to this vast topic which is the computer-assisted maintenance management and its development using artificial neural networks, to predict the moment of failure and to be able to do preventive maintenance and implemented and tested on aircraft engine data from the National Aeronautics and Space Administration's Center of Excellence for Prognostics (NASA)..

Keywords : computer-assisted maintenance management, artificial neural networks, preventive maintenance.

Remerciements

Nous remercions, en premier lieu, notre Dieu le plus Puissant qui a bien Voulu nous donner la force et le courage Pour effectuer Le présent travail.

Nous exprimons nos profondes gratitudes à nos parents pour leurs Encouragements, leur soutien et pour les sacrifices Qu'ils ont enduré.

En deuxième lieu, nous tenons à remercier notre encadreur " Mehdi Rouan Serik " pour son sérieux, sa compétence et ses orientations.

Nous remercions vont aussi à tous les enseignants qui ont contribué à notre formation.

Aussi nous tenons à exprimer notre reconnaissance à tous nos amis et Qui par leur amitié et leur bonne humeur, ont créé une ambiance de travail parfaite.

Dédicaces

Je dédie ce travail

A ma famille, elle qui m'a doté d'une éducation digne, son amour a fait de moi ce que je suis aujourd'hui :

Particulièrement à mon père, pour le gout à l'effort qu'il suscité en moi, de par rigueur.

A toi ma mère, ceci est ma profonde gratitude pour ton éternel amour, que ce rapport soit le meilleur cadeau que je puisse t'offrir.

A mon très cher frère Chouaib qui je le souhaite tout le bonheur de la vie.

A ma petite sœur sabrine qui je l'aime beaucoup, que dieu te protège pour moi.

A toutes les personnes que j'aime et surtout celles qui font toute choses pour mon bien être ;

JE VOUS AIME.

Rami

Dédicaces

A mon père à qui aucune dédicace ne saurait exprimer l'amour, l'estime et le respect que j'ai pour lui.

A ma mère, mon bonheur, rien au monde ne vaut les efforts fournis jours et nuits pour mon éducation et mon bien être, tu es la meilleure maman au monde et à qui je dois toute réussite.

A mon très cher frère Abdeljalil qui je le souhaite une vie pleine de joie et de réussite.

A mes grands-parents, mes tantes et oncles qui ont toujours cru en moi.

A ma deuxième famille avec qui j'ai partagé mes plus beaux jours et que je garderai les meilleurs souvenirs. Je n'oublierai jamais nos chagrins, nos rires, nos aventures, nos folies.

A toutes les personnes que j'aime et surtout celles qui font toute choses pour mon bien être ;

JE VOUS AIME.

Abdelkarim

Table des matières

Abréviations	i
Liste de Figures	ii
Liste des tableaux	iii
Introduction Générale.....	1
Chapitre1 :	Généralité sur la maintenance
<hr/>	
I.1. Introduction.....	2
I.2. Historique.....	2
I.3. Définitions AFNOR et CEN de la maintenance.....	2
I.3.1. Définitions de la maintenance selon l'AFNOR par la norme X 60-000.....	3
I.3.2. Missions et objectifs.....	3
I.4. La fonction maintenance	4
I.4.1. La fonction préparation	5
I.4.2. La fonction réalisation	5
I.5. La communication au sein du service maintenance	6
I.6. Les cinq niveaux de maintenance	6
I.7 Diffèrent types de maintenance.....	8
I.7.1.La Maintenance Corrective	8
I.7.2. La maintenance préventive	10
I.7.3. La préparation et l'ordonnancement en maintenance préventive	12
I.8. Mise en place et contrôles	12
I.9. Gestion de maintenance	13
I.9.1. Principe de la gestion	13
I.9.2. Les objectifs de la maintenance	13
I.9.3. Les cinq zéros.....	14
I.9.4. Aspects de la maintenance	15
I.10. Les formes organisationnelles de la maintenance	15
I.10.1. La maintenance centralisée	16
I.10.2. La maintenance répartie ou décentralisée	16
I.10.3. Organisation mixte	16
I.10.4. La maintenance externalisée (sous-traitance)	16
I.11. CONCLUSION.....	17

II.1. Introduction.....	18
II.2. Historique de l'évolution de l'informatique industrielle	18
II.3. Un progiciel de GMAO	19
II.4. La GMAO un outil incontournable.....	19
II.5. La réussite d'une GMAO	20
II.6. Les conditions de réussite d'un GMAO	20
II.6.1. La convivialité de l'outil informatique.....	20
II.6.2. Volonté de réussite de consensus.....	20
II.6.3. Facteur temps.....	20
II.7. Structure de la base de données maintenance	21
II.8. Les fonctionnalités du système GMAO.....	21
II.9. Les progiciels de GMAO (analyse des différents modules fonctionnels)	22
II.9.1 MODULE « GESTION DES EQUIPEMENTS ».....	23
II.9.2 MODULE « GESTION DU SUIVI OPÉRATIONNEL DES ÉQUIPEMENTS »	24
II.9.3 MODULE « GESTION DES INTERVENTIONS »	25
II.9.4 MODULE « GESTION DU PRÉVENTIF ».....	27
II.9.5 MODULE « GESTION DES STOCKS »	27
II.9.6 MODULE « GESTION DES APPROVISIONNEMENTS ET DES ACHATS »	28
II.9.7 MODULE « ANALYSES DES DÉFAILLANCES »	29
II.9.8 MODULE « BUDGET ET LE SUIVI DES DÉPENSES »	29
II.9.9 MODULE « GESTION DES RESSOURCES HUMAINES »	30
II.9.10 MODULE « TABLEAUX DE BORD ET STATISTIQUES »	30
II.9.11 MODULE COMPLÉMENTAIRES OU INTERFAÇAGES UTILES.....	30
II.10. LE CHOIX D'UN OUTIL GMAO BIEN ADAPTÉ	31
II.10.1 Intégration de la GMAO dans le système d'information de l'entreprise	31
II.10.2 Importance du paramétrage : la « flexibilité » d'une GMAO	32
II.10.3 Importance du cahier des charges.....	32
II.11. LA CONDUITE D'UN PROJET GMAO.....	32
II.11.1 IMPORTANCE DE L'ASPECT HUMAIN DANS LA RÉUSSITE DU PROJET	32
II.11.2 ÉTAPES DU PROJET.....	33
II.11.3. QUELQUES CAUSES D'ÉCHEC	35
II.12. Conclusion.....	37

III.1. Introduction	38
III.2. Machine Learning (ML)	38
III.2.1. Définition.....	38
III.2.2. Les principales méthodes de machine learning	39
III.2.3. Champs d'application du Machine Learning.....	40
III.2.4. L'avantage du Machine Learning.....	41
III.2.5. Les limites du Machine Learning.....	41
III.3. Machine Deep Learning (MDL).....	41
III.3.1 Définition.....	41
III.3.2. Fonctionnement	42
III.3.3. L'évolution des réseaux de neurones	42
III.3.4. Champs d'application de Machine Deep Learning [23].....	43
III.3.5. Limites de machine deep Learning.....	43
III.4. Les différences entre les Machine Learning et les Machine Deep Learning.....	44
III.5. Synthèse des différences entre Machine Learning et Deep Learning.....	45
III.6. Les réseaux de neurones	45
III.6.1. Introduction	45
III.6.2. Historique	46
III.6.3. Le neurone biologique.....	47
III.6.4. Le neurone artificiel	48
III.6.5. Fonction d'activation.....	50
III.7. Les réseaux de neurones artificiels.....	51
III.7.1. Différentes architectures de réseaux de neurones.....	52
III.7.2. Apprentissage	54
III.8. Conclusion.....	56

Chapitre IV

conception et implémentation

IV.1. Introduction	58
IV.2. Problématique (Maintenance assistée par ordinateur : maintenance prédictive des pannes d'un turboréacteur)	58
IV.3. Conception	59
IV.3. 1.Environment & outils de développement	59
IV.3.1.1.Environment matériel	59
IV.3. 1.2.Outil de développement	59
IV.3. 2. Modèle d'apprentissage.....	60
IV.3. 2.1.Préparation des données	60

IV.3. 2.2.Création du modèle	65
IV.3. 2.3.Test du modèle	67
IV.4. Résultats obtenus.....	68
IV.4.1. Analyse des résultats	71
IV.4.2. Définition de Pareto	71
IV.4.3. Principe de Pareto (METHODE ABC)	72
IV.5.Conclusion.....	75
Conclusion General	76
Bibliographie	77
Annexes	79

Abréviations

GMAO : gestion assisté par ordinateur

AFNOR : Association Française de Normalisation

CEN : Comité Européenne de Normalisation

L’A.M.D.E.C. : Analyse des Méthodes de Défaillance de Leur Effets et de leur Criticité

TRS : taux de rendement synectique

TPM : total prédictive maintenance

DT : demande de travail

OT : ordre de travail

DA : demande d’approvisionnement

DTE : dossier technique Équipement

BSM : bon de sortie magasin

DTE : dossier technique Équipement

TTR : temps technique de réparation

IA : apprentissage artificiel

ML : Machine Learning

MDL : Machine Deep Learning

RNA : réseaux de neurones artificiels

NASA : The National Aeronautics and Space Administration

Liste de Figures

Figure I.1: Les organismes de normalisation	2
Figure I.2:les différentes taches contenues dans la fonction maintenance.....	4
Figure I.3:: Activités contenues dans les taches de préparation.....	5
Figure I.4:Quelques flux de communication interne en maintenance.....	6
Figure I.5:Les différents types de maintenance	8
Figure I.6:Les cinq zéros olympique.....	14
Figure I.7: Les formes organisationnelles de la maintenance	16
Figure II.1:Structure de SGBD.....	21
Figure II.2:Exemple de structure modulaire d'une GMAO	22
Figure II.3:Exemple de structure modulaire d'une GMAO.....	24
Figure II.4:fonction de la maintenance.....	34
Figure III.1:schéma simplifié d'un neurone du cerveau humain	48
Figure III.2: Le neurone artificiel générique.....	49
Figure III.3:Un neurone biologique et Une synapse	49
Figure III.4: impulsions électrique entre les neurones	50
Figure III.5:fonction d'activation.....	51
Figure III.6: Réseau multicouche	52
Figure III.7:Structure multicouche à réseau de connexion locale.....	53
Figure III.8:Réseau à connexion récurrente	53
Figure III.9:Réseau de connexion complète.....	54
Figure III.10: La séparation non linéaire (rétro-propagation : cas général).....	56
Figure IV.1:Schéma simplifié d'un moteur à turboréacteur	58
Figure IV.2:Les bibliothèques requises.....	61
Figure IV.3:importer le fichier 'train_FD001'	62
Figure IV.4:importer le fichier 'test_FD001'	62
Figure IV.5: importer le fichier 'RUL_FD001'	63
Figure IV.6:ajouter une nouvelle colonne 'rtf'(run to failure)	63
Figure IV.7:ajouter ttf et rtf au tableau des données de test.....	64
Figure IV.8: ajouter une nouvelle colonne ' ttf' dans le tableau des données de train.....	64
Figure IV.9:Créez deux nouvelles tableaux et nommez-les df_train et df_test	64
Figure IV.10: générer x-train et y-train	65
Figure IV.11:générer x-test et y-test.....	66
Figure IV.12:Création de model avec plusieurs couches	66
Figure IV.13:donner x-train et y-train et démarrer l'entraînement de modèle.....	67
Figure IV.14: Evaluation de la précision du modèle dans la phase d'entraînement.....	67
Figure IV.15:comparaison entre y-test et y-pred.....	67
Figure IV.16:Créer une fonction 'proba failure' pour faire le test	68
Figure IV.17:calculer probabilité des pannes de cent machine.....	68
Figure IV.18: Créer le graphe de probabilité des pannes pour cent machines	70
Figure IV.19: ID machines vs Probabilité	70
Figure IV.20: créer l'histogramme de probabilité des pannes de cent machines	70
Figure IV.21: Histogramme des probabilités de pannes.....	71
Figure IV.22:Probabilité des pannes et zones Pareto.....	74

Liste des tableaux

Tableau I.1:Les cinq niveaux de maintenance.....	7
Tableau III.1:Différences entre Machine Learning et Deep LearningIII.6. Les réseaux de neurones ..	45
Tableau IV.1: Probabilité de pannes de cent machines.	69
Tableau IV.2: Probabilité des pannes et classification selon la méthode pareto.	73

Introduction Générale

Aujourd'hui, le développement rapide de l'informatique, en particulier les Micro- ordinateurs avec des systèmes d'exploitation de plus en plus conviviaux Permet une interface homme-machine très simple : Plus besoin de solides connaissances informatiques pour l'utiliser la GMAO.

La GMAO Doit être un outil puissant pour que le personnel de maintenance puisse gagner du temps et Avec un accès facile à l'historique et Autre analyse. L'informatique en tant que technologie de gestion et d'organisation est devenue Un outil indispensable pour toute organisation qui demande beaucoup de management information. Dans notre travail, on va parler de l'influence de l'intelligence artificiel dans la gestion maintenance assisté par ordinateur, en utilisent les réseaux de neurone artificiels a la prédiction des pannes dans les machines à l'aide des paramètres et l'historique des panne dans la Machine Le premier chapitre a été consacré pour rappeler les généralités de maintenance et ces différents types.

Dans le chapitre 2, nous discuterons de la gestion maintenance assisté par ordinateur (GMAO) et sont rôle important dans le domaine industriel.

Le chapitre 3 sera dédié au machine Learning et machine deep learning ainsi que la différence entre eux et à définir les réseaux de neurones.

Dans le chapitre 4 Nous nous concentrerons sur une problématique de GMAO : maintenance prédictive des pannes d'un turboréacteur et on va expliquer comment la propagation des dommages peut être modélisée dans le turboréacteur des avions, puis on va faire une simulation avec python pour détecter un défaut après une période étudié.

Chapitre I :

Généralité sur la

maintenance

I.1. Introduction

Pour être et demeurer compétitive, une entreprise doit produire toujours mieux (qualité) et au coût le plus bas. Pour minimiser ce coût, on fabrique plus vite et sans interruption des produits sans défaut afin d'atteindre la production maximale par unité de temps. Cet objectif est un des buts de la fonction maintenance d'une entreprise. Il s'agit de maintenir un bien dans un état lui permettant de répondre de façon optimale à sa fonction.

Ce chapitre examine les définitions fondamentales concernant la maintenance et le rôle de cette dernière dans l'industrie. Ensuite l'accent est mis sur l'intégration d'éléments mathématiques dans la maintenance pour évaluer le comportement du matériel en service.

I.2. Historique

Le terme « maintenance », forgé sur les racines latines Manus et Ténéré, est apparu dans la langue française au XII^e siècle. L'étymologiste Wace a trouvé la forme maintenir (celui qui soutient), utilisée en 1169 : c'est une forme archaïque de « mainteneur ». Anecdotiquement, c'est avec plaisir que j'ai retrouvé l'usage du mot « maintenance » sous la plume de François Rabelais, qui, vers 1533, parlait de la « maintenance de la LOY » dans Pantagruel. Les utilisations anglo-saxonnes du terme sont donc postérieures. À l'époque moderne, le mot est réapparu dans le vocabulaire militaire : « maintien dans des unités de combat, de l'effectif et du matériel à un niveau constant ». Définition intéressante, puisque l'industrie l'a reprise à son compte en l'adaptant aux unités de production affectées à un « combat économique » [1].

I.3. Définitions AFNOR et CEN de la maintenance

Remarque préalable : L'AFNOR faisant partie du CEN, les définitions AFNOR seront supplantées par les définitions CEN lorsque les normes « projet » deviendront définitives.

Il n'existe pas actuellement de normes internationales ISO (International Standardization Organization) relatives à la fonction maintenance.

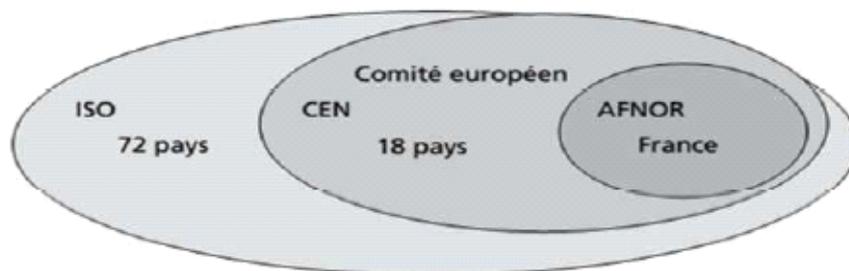


Figure I.1 – Les organismes de normalisation [1]

I.3.1. Définitions de la maintenance selon l'AFNOR par la norme X 60-000

La maintenance est l'ensemble de toutes les actions techniques, administratives et de Management durant le cycle de vie d'un bien, destinées à le maintenir ou à le rétablir dans Un état dans lequel il peut accomplir la fonction requise [2]. Bien maintenir, c'est assurer l'ensemble de ces opérations au coût optimal [1].

La définition de la maintenance fait donc apparaître 4 notions :

- Maintenir : qui suppose un suivi et une surveillance.
- Rétablir : qui sous-entend l'idée d'une correction de défaut.
- État spécifié et service déterminé : qui précise le niveau de compétences et les objectifs Attendus de la maintenance.
- Coût optimal : qui conditionne l'ensemble des opérations dans un souci d'efficacité économique.

I.3.2. Missions et objectifs

La maintenance est appelée à remplir les missions suivantes :

- Assurer la continuité de marche de l'outil de production (diagnostic, réparation, révisions Et prévention).
- Faire les modifications demandées en accord avec la production pour améliorer la productivité.
- Mettre en place et vérifier les dispositifs de sécurité tant pour le personnel que pour les installations.

Ses principaux objectifs peuvent être résumés en :

- Améliorer la disponibilité de l'équipement de production par :
 - Une diminution des pannes et défaillances et augmenter ainsi la fiabilité des équipements.
 - La mise en place d'un système de suivi des indicateurs de performances, notamment la disponibilité des équipements.
- Améliorer la qualité du service par la mise en place de relation de type client / fournisseur entre la production et la maintenance.
- Diminuer les coûts de maintenance par un suivi rigoureux de tous les paramètres qui y entrent (main d'œuvre, matières...).

I.4. La fonction maintenance

La fonction maintenance peut être considérée comme un ensemble d'activités regroupées en deux sous-ensembles : les activités à dominante technique et les activités à dominante de gestion (figure I.2).

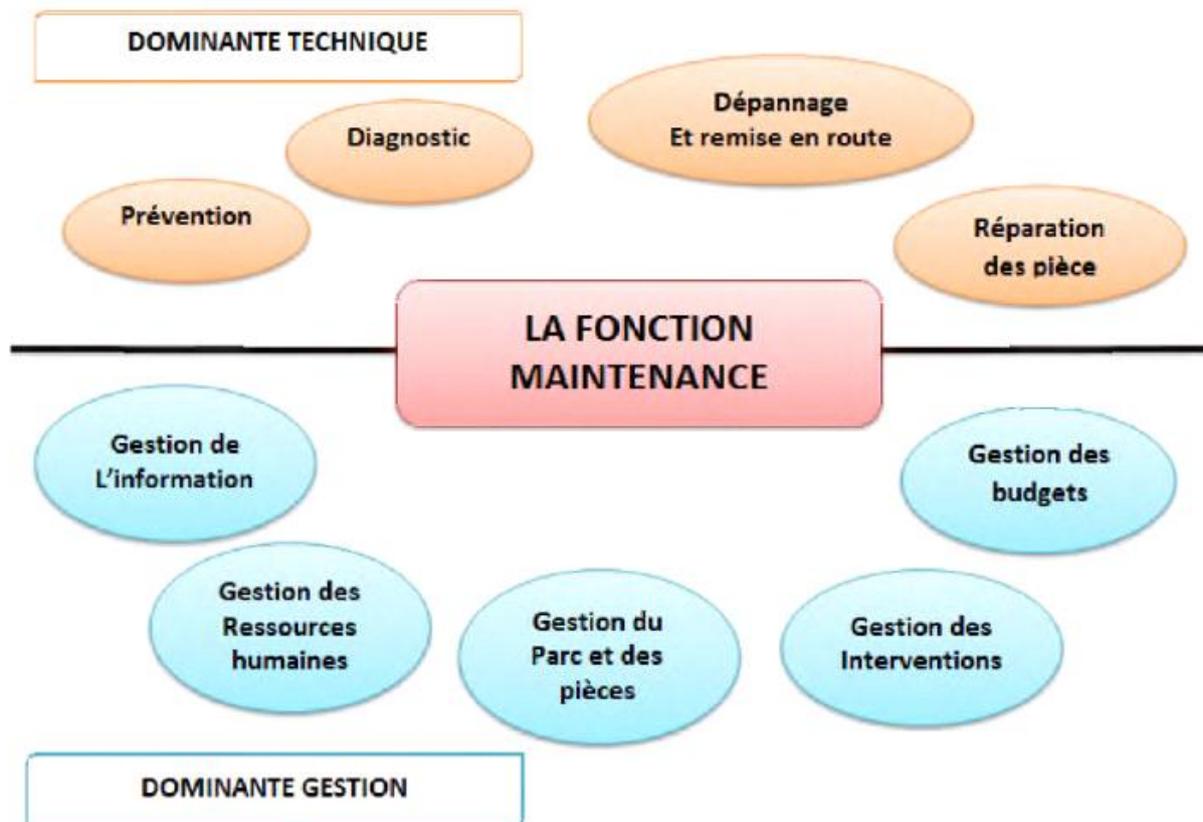


Figure I.2 : les différentes taches contenues dans la fonction maintenance [3]

Chacune de ces composantes est constituée de taches bien précises dont certaines sont assurées par plusieurs sous-fonctions. Pour simplifier nous désignerons les sous-fonctions par fonctions.

C'est ainsi que nous pouvons distinguer les fonctions ci-après :

- La fonction préparation.
- La fonction ordonnancement.
- La fonction réalisation
- La fonction gestion du service maintenance

I.4.1. La fonction préparation

C'est la fonction qui est chargée de prévoir, définir et réaliser les conditions optimales d'exécution d'un travail. La figure I.3 nous montre que, pour une tâche donnée, la préparation consiste à définir des moyens humains et matériels et à éditer des documents opérationnels [3].

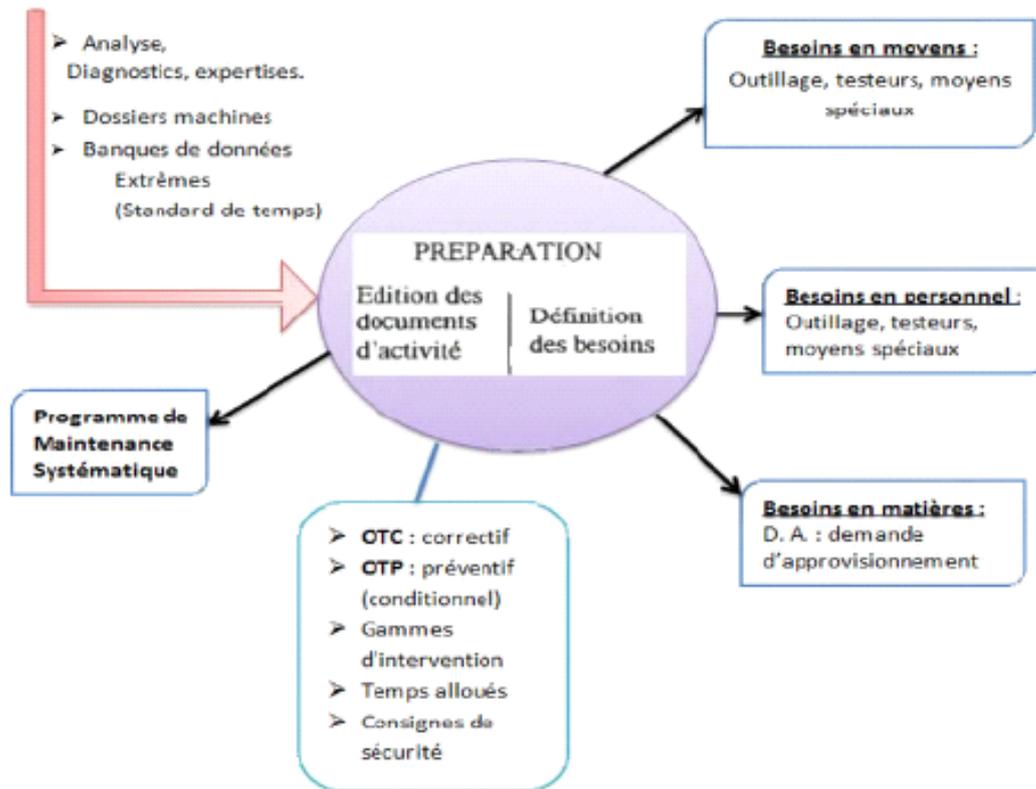


Figure I.3 : Activités contenues dans les tâches de préparation [3]

I.4.2. La fonction réalisation

Cette fonction assure des missions importantes qui peuvent être classées en cinq niveaux comme suit :

I.4.2.1. Mission et activités

La mission de cette fonction est d'utiliser les moyens mis à disposition, suivant les Procédures définies, pour remettre le matériel dans l'état spécifié. A ce titre cette fonction peut être appelé à effectuer :

- Des actions sur du matériel non en service comme participer à l'installation, la mise en service, réglage, etc...
- Des opérations de maintenance corrective telle que des tests, diagnostic, dépannage, réparation, remplacement ;
- Des opérations de maintenance préventive : ronde (de graissage par

exemple), révision, rénovation et reconstruction.

Notons aussi qu'en plus de ces actions traditionnelles, la réalisation s'occupe d'autres activités non moins importantes que sont : veiller à la sécurité des intervenants et le nettoyage des ateliers, des chantiers et des équipements.

I.5. La communication au sein du service maintenance

Nous allons décrire brièvement un système de communication assez traditionnel dans les services maintenance, relatif à une intervention corrective « lourde » prise entre la demande d'intervention et sa clôture. Nous utiliserons les abréviations suivantes :

- DT : demande de travail, ou DI, demande d'intervention, ouvrant un n° de référence, provenant du « client interne »,
- OT : ordre de travail, géré par l'ordonnancement,
- BT : bon de travail, accompagnant la préparation et retourné complété après intervention,
- DA : demande d'approvisionnement,
- BSM : bon de sortie magasin.

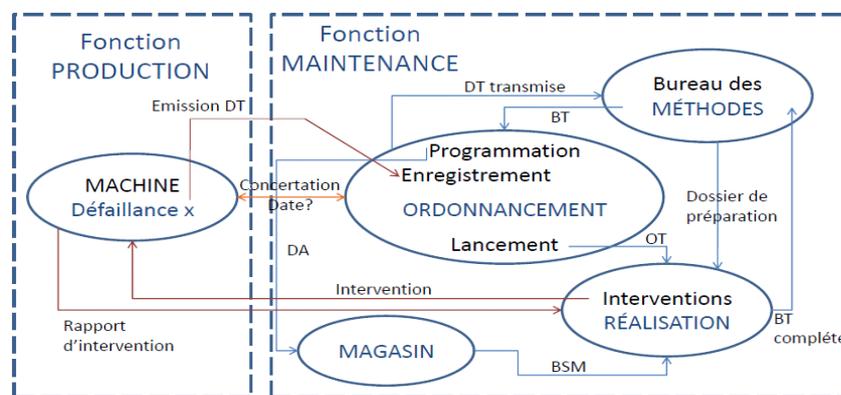


Figure I.4 : Quelques flux de communication interne en maintenance [1]

I.6. Les cinq niveaux de maintenance

La maintenance est caractérisée par une très grande variabilité des tâches, en natures comme en durées. D'où l'utilité de jeter les bases de son organisation à partir d'une mise en familles à cinq niveaux (tableau I.1), suivant la norme AFNOR X 60-000. La tendance actuelle est de se ramener à trois niveaux seulement, dans une logique de TPM. À savoir : I= niveau 1 + niveau 2 c'est la maintenance « de première ligne » transférée

progressivement aux opérateurs de production, assistés si nécessaire par les techniciens de maintenance de l'antenne sectorisée.

II = niveau 3 + niveau 4 : domaine d'action privilégié des équipes polyvalentes de techniciens de maintenance. Diagnostics, interventions techniquement évoluées, mise en œuvre d'améliorations, etc.

III = niveau 5 : travaux spécialisés souvent sous-traités pour que la maintenance puisse recentrer ses moyens sur son savoir-faire (le niveau II). (Voir exemple sur le tableau I.1)

Niveau	Personnel d'intervention	Nature de l'intervention	Moyens requis
1	Exploitant, sur place	Réglage simple d'organes accessibles sans aucun démontage, ou échanges d'éléments accessibles en dans les consignes de toute sécurité conduite.	Outillage léger défini dans les consignes de conduite
2	Technicien habilité (dépanneur) sur place	Dépannage par échange standard d'éléments prévus à cet effet, ou opérations mineures de maintenance préventive	Outillage standard et rechanges situés à proximité
3	Technicien spécialisé, sur place ou en atelier de maintenance	Identification et diagnostics de pannes, réparations par échange de composants fonctionnels, réparations mécaniques mineures	Outillage prévu plus appareils de mesure, banc d'essai, de contrôle
4	Équipe encadrée par un technicien spécialisé, en atelier central	Travaux importants de maintenance corrective ou préventive. Révisions	Outillage général et spécialisé
5	Équipe complète polyvalente, en atelier central	Travaux de rénovation, de reconstruction, réparations importantes confiées à un atelier central souvent externalisés	Moyens proches de ceux de la fabrication par le constructeur

Tableau I.1 : Les cinq niveaux de maintenance [1]

I.7 Différents types de maintenance

Nous avons vu que la maintenance peut être subdivisée en deux principales composantes qui sont : la maintenance corrective et la maintenance préventive. La figure I.5 nous en donne une image. Certains spécialistes parlent d'une troisième, la maintenance améliorative, dont on parle peu car dépendante des deux autres.

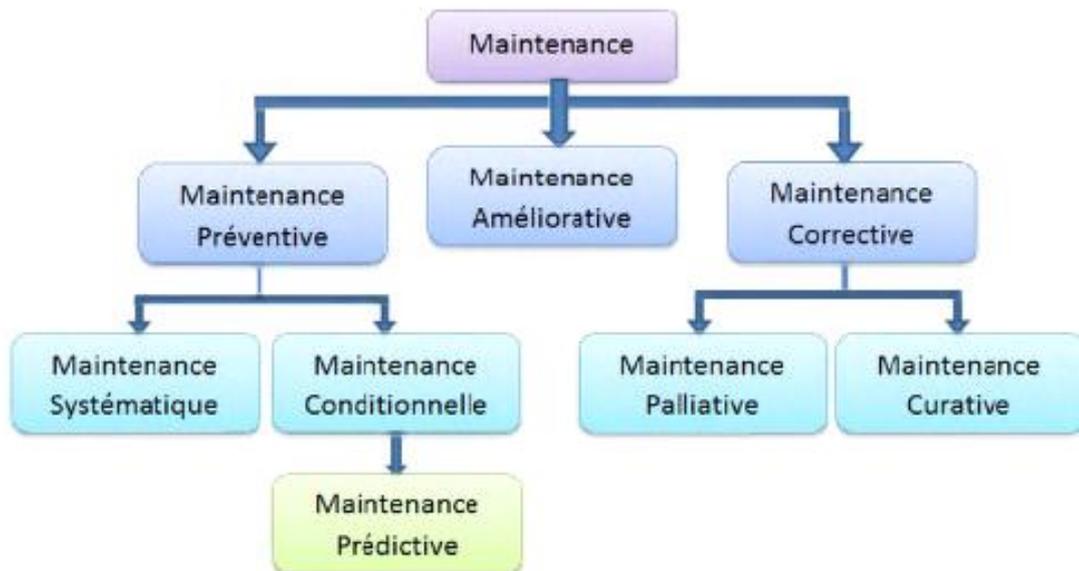


Figure I.5 : Les différents types de maintenance. [3]

I.7.1. La Maintenance Corrective

C'est l'ensemble des activités réalisées après la défaillance d'un bien ou la dégradation de sa fonction, pour lui permettre d'accomplir une fonction requise, au moins provisoirement.

Note : la maintenance corrective comprend :

- La remise en état avec ou sans modification.
- Le contrôle du bon fonctionnement.

I.7.1.1. Maintenance palliative

C'est la maintenance qui permet de remettre en état de fonctionnement un équipement de façon provisoire. Elle est effectuée dans des conditions extrêmes et imposée par l'une des situations suivantes :

- Un manque de pièces de rechange pour effectuer les travaux de réparation nécessaires.
- Des contraintes de production à satisfaire ne permettant pas d'avoir suffisamment de temps pour intervenir.

- Un manque de compétences capables d'exécuter les travaux. C'est une maintenance dans laquelle on tente seulement d'agir sur les effets sans se préoccuper des causes qui les produisent. Par conséquent elle ne permet pas d'éviter une répétition de certains types de pannes.

I.7.1.2. Maintenance curative

C'est la maintenance réalisée suite à un dysfonctionnement de l'équipement. Elle consiste à le remettre en état de fonctionnement en procédant à des réparations complètes. Elle conduit à des actions de diagnostic permettant d'identifier les causes de la panne ou Défaillance et de préciser les opérations de maintenance nécessaires pour la remise en état. Ces opérations peuvent être : une rénovation ou une révision (dans certains cas). [3]

I.7.1.3. La préparation et l'ordonnancement en maintenance corrective

I.7.1.3.1. La préparation

Il faut noter que ce ne sont pas tous les travaux de maintenance qui doivent faire objet de préparation. Les principaux travaux à préparer sont :

- Les travaux importants qui sont déterminés par la méthode Pareto. Ce sont les 20 % à 30 % des interventions qui prennent les 70 % à 80 % du temps passé.
- Les travaux répétitifs : est de même que les précédents sauf qu'ici le critère déterminant est le nombre d'interventions.
- Les travaux de révision : qui sont à la fois importants (en heures) et répétitifs ;
- Les travaux de haute qualité : ce sont des interventions délicates et coûteuses ;
- Les travaux liés à la sécurité : la préparation sera axée sur la sécurité (habilitation, Consignation...).

Selon le cas, la préparation comportera :

- Un diagnostic : il s'agit du premier constat qui orientera les investigations à mener lors de l'expertise.
- Une expertise : étape la plus importante de la préparation, elle consiste en une « Auscultation » complète de l'organe défaillant afin de déterminer les actions de réparation à entreprendre.
- Une gamme d'intervention : Établie à partir du rapport d'expertise. Elle contient toutes les actions et tous les moyens à déployer et à mettre en œuvre pour l'équipement défaillant.

I.7.1.3.2. L'ordonnancement

L'ordonnancement s'occupera, dans ce cas, du traitement administratif des interventions. Notamment la gestion des temps d'intervention, le suivi du plan des charges des différentes équipes d'exécution des travaux, la vérification de la disponibilité des pièces de rechange avant d'envoyer les demandes d'intervention à la réalisation.[3]

I.7.2. La maintenance préventive

C'est une « maintenance effectuée dans l'intention de réduire la probabilité de défaillance d'un bien ou d'un service rendu ».

Elle vise les principaux objectifs suivants :

- Augmenter la fiabilité d'un équipement, donc réduire les défaillances en service :
Réduction des coûts de défaillance, amélioration de la disponibilité.
- Augmenter la durée de vie d'un équipement.
- Améliorer l'ordonnancement des travaux, donc les relations avec la production.
- Réduire et régulariser la charge de travail.
- Faciliter la gestion des stocks (consommations prévues à l'avance)
- Assurer la sécurité (moins d'improvisations dangereuses)
- Plus globalement, réduire la part du « fortuit », améliorer le climat des relations Humaines (une panne imprévue est toujours génératrice de tensions).

Elle regroupe la maintenance systématique, la maintenance conditionnelle et la maintenance Prévisionnelle ou prédictive.

I.7.2.1. Maintenance systématique

C'est une maintenance planifiée selon une fréquence de temps fixe (échancier établi. jour, semaine, mois, année) ou de temps de fonctionnement ou de nombre d'unités d'usage (Valeur compteur : heures de marche, pièces fabriquées, distance parcourue,...), l'objectif étant de remplacer les pièces d'usure avant l'apparition d'un dysfonctionnement. Elle peut, selon les cas, être décomposée en deux sous composantes :

I.7.2.1.1. La maintenance systématique de type Age

C'est un remplacement systématique des rechanges préalablement déterminées, des lors que l'échéance ou l'usage est atteint, sans se préoccuper de l'état de ces rechanges (dégrade ou pas). Ceci se rencontre, le plus souvent, dans le cas de contraintes de production très sévères ne permettant pas de prendre le moindre risque.

I.7.2.1.2. La maintenance préventive systématique :

Les rechanges font l'objet d'une inspection et seules celles présentant un niveau d'usure Avancé sont remplacées. En plus des opérations périodiques, on peut citer comme actions de maintenance systématique les opérations suivantes : la révision et modification.

I.7.2.2. Maintenances conditionnelle et prévisionnelle (ou prédictive)

Elle peut être subdivisée en deux composantes :

I.7.2.2.1. Maintenance conditionnelle

C'est une maintenance préventive consistant en une surveillance du fonctionnement du Bien et des paramètres significatifs de ce fonctionnement intégrant les actions qui en découlent. Ces actions de maintenance sont alors déclenchées suivant des critères prédéterminés significatifs de l'état de dégradation du bien ou du service. Les remplacements ou les remises en état des pièces, les remplacements ou les appoints des huiles ont lieu après une analyse de leur état de dégradation. Une décision volontaire est alors prise d'effectuer les remplacements ou les émises en état nécessaire c'est une méthode moderne qui permet un suivi continu du matériel en service dans le but de prévenir une défaillance.

I.7.2.2.2. Maintenance prévisionnelle ou prédictive

Elle est basée sur l'analyse des mesures de certains paramètres de l'équipement : Température, vibration, qualité des huiles, pour tenter d'éviter un dysfonctionnement. Elle est aussi définie comme « maintenance préventive exécutée en suivant les prévisions extrapolées de l'analyse et de l'évaluation de paramètres significatifs de la dégradation du bien. ».

Ces deux types de maintenances préventives sont souvent confondus. Elles sont déclenchées par les résultats du contrôle de l'état de l'équipement ou de la mesure de ses paramètres de fonctionnement.

La mise en place d'un tel type de maintenance nécessite des choix préalables qui sont, Généralement faits de deux manières différentes :

- Les organes importants et vitaux pour le fonctionnement des installations.
- Les organes ayant présente des défaillances dont la connaissance par retour d'expérience aura permis d'évaluer les risques et leur gravite. Pour cela il est utilisé certaines techniques comme l'A.M.D.E.C. (Analyse des Méthodes de

Défaillance, de Leur Effets et de leur Criticité), qui est une méthode d'analyse de la fiabilité qui permet de recenser les défaillances dont les conséquences affectent le fonctionnement d'un système donné, pour déterminer les organes à suivre.

I.7.3. La préparation et l'ordonnancement en maintenance préventive

La vocation de la maintenance préventive est de prévenir un certain nombre de défaillances et de mieux connaître le comportement de l'équipement afin de lui appliquer une maintenance systématique et/ou conditionnelle.

I.7.3.1. La Préparation

Pour la maintenance préventive la préparation a plutôt lieu en amont à travers :

- Le choix des parties sensibles des équipements et/ou les paramètres à visiter ou remplacer (maintenance systématique) ou à suivre (maintenance conditionnelle ou prédictive).
- L'établissement des fiches de visites et de contrôle, respectivement pour les systématiques et les conditionnelles.
- La détermination de l'échéancier des visites ou contrôles.
- La détermination des seuils (maintenance conditionnelle ou prévisionnelle) qu'une fois atteints, doivent conduire au déclenchement des interventions.

Après la visite ou le contrôle, nous retrouvons les mêmes travaux de préparations et les mêmes actions que dans le cas de la maintenance corrective.

I.7.3.2. L'ordonnancement

Il s'agit, ici, de veiller au lancement à temps des différents processus de visites et Contrôles des équipements concernés par la maintenance préventive. C'est ainsi que nous avons :

Pour la maintenance systématique : le lancement est fait suivant l'état présumé de l'équipement, à partir d'un échéancier qui déclenche les actions à entreprendre.

Pour la maintenance conditionnelle : le déclenchement est fait suivant l'état constaté du Matériel, lors du contrôle.

I.8. Mise en place et contrôles

Contrairement à la maintenance corrective et à la maintenance préventive systématique qui sont déclenchées, respectivement, par l'apparition de la défaillance ou de la

panne et l'atteinte d'une échéance, le déclenchement de la maintenance préventive conditionnelle et prévisionnelle dépend de l'atteinte d'un seuil déterminé d'un paramètre physique (température, vibrations, ...). Selon les paramètres physiques à suivre il existe des moyens (ou techniques) de contrôle bien précis. Nous pouvons citer quelques-unes de ces techniques : [3]

- Les mesures et analyses vibratoires qui peuvent être faites de manière permanente ou périodique.
- Les analyses d'huile qui sont effectuées aussi bien sur les équipements mécaniques (analyses physico-chimiques) que pour les équipements électriques (rigidité diélectrique des huiles) ;
- Les contrôles plus spécifiques tels que l'endoscopie, la thermographie infrarouge et les C.N.D. (contrôle non destructif : radiographie par rayon γ et X, ressuage, mesures d'épaisseur par ultrasons)

I.9. Gestion de maintenance

I.9.1. Principe de la gestion

Gérer c'est administrer, diriger, gouverner, exercer des fonctions de direction est de contrôle pour son propre compte ou pour le compte d'un autre.

La gestion de la maintenance dans une installation industrielle c'est :

- Lui définir des objectifs chiffrés est mesurable.
- Définir les moyens à mettre en œuvre pour atteindre ses objectifs.

Le gestionnaire de maintenance est responsable de la mise en classe d'un système de gestion adapté à son entreprise, il doit tenir compte :

- Des spécificités de l'entreprise.
- De sa taille.
- De l'importance de la maintenance.
- Du degré d'information.
- Mesurer les résultats, les comparer avec les objectifs, analyser les écarts et décider des moyens à mettre en œuvre pour corriger la déviation.

1.9.2. Les objectifs de la maintenance

Les objectifs de la gestion de maintenance seront atteints si le gestionnaire maîtrise parfaitement les paramètres et les conditions de fonctionnement de l'entreprise.

Le rôle de la maintenance et donc de traiter des défaillances afin de réduire est si possible d'éviter les arrêts de production.

La maintenance est indissociable des poursuites des objectifs conduisant à la maîtrise de la qualité, les cinq zéros symbolisant les objectifs, concernent en effet la maintenance, est un fonctionnement avec : [5]

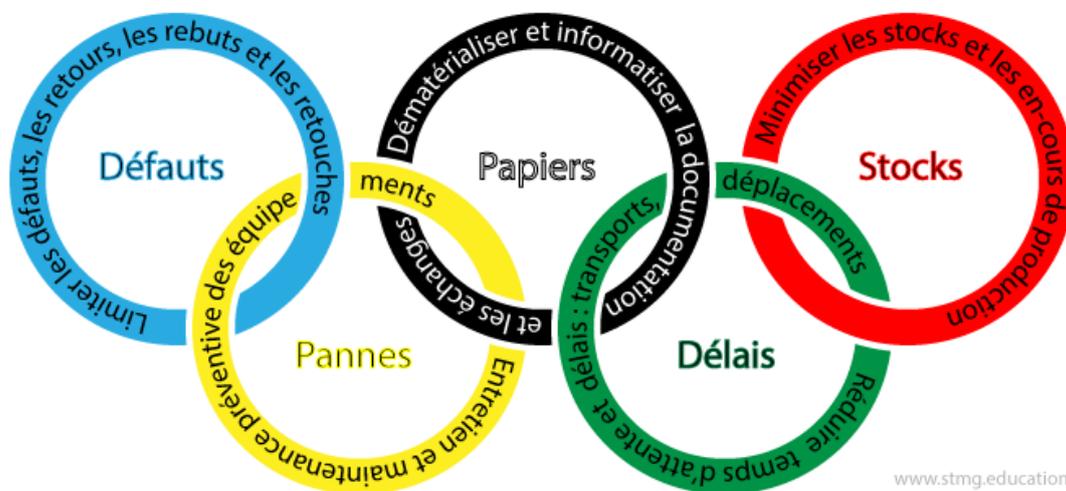


Figure I.6 : Les cinq zéros olympique [6].

1.9.3. Les cinq zéros

Le concept des 5 zéros est quelque part le catalyseur des méthodes du toyotisme, telles que le Juste à temps et Kanban et, par extension des démarches qualité.

1.9.3.1. Zéro défaut

Limiter les défauts et les retours, réduire les rebuts et les opérations de retouches, bref essayer le plus possible de faire bien la première fois. L'effort peut être particulièrement conséquent. Mais les gains liés à un meilleur emploi des ressources, une moindre consommation et un accroissement de la satisfaction client sont en général rapidement significatifs.

1.9.3.2. Zéro papier

Limiter la production et la circulation de documents papiers, dématérialiser et informatiser la documentation, exploiter les fonctions sont autant de conseils bien plus faciles à distribuer qu'à mettre en œuvre. Accéder au zéro papier ou en tout cas en réduire

drastiquement la quantité en circulation est un véritable projet en soi.

I.9.3.3. Zéro pannes

Mettre en œuvre des principes de maintenance préventive. Pour inciter à l'utilisation de ces principes, il est intéressant de comparer le coût d'une période de maintenance préventive avec le coût complet des pannes susceptibles d'être évitées. Le bilan est sans discussion.

I.9.3.4. Zéro stock

Il s'agit de maintenir les stocks au minimum. Un stock non seulement coûte et pénalise la flexibilité (changement de série, de lots) mais de surcroît masque les problèmes de fluidité du processus.

I.9.3.5. Zéro délai

Le zéro délai, c'est l'optimisation optimale des processus pour servir le client dans les délais les plus courts possible, quelle que soit la commande. C'est là la recette pour résister à la Concurrence et gagner et conserver les clients .C'est aussi la pression maximale sur les Opérateurs qui n'ont alors guère le temps pour relaxer et décompresser.

I.9.4. Aspects de la maintenance

La maintenance d'un bien commence avec la prise en compte, dès sa conception, des notions de fiabilité de maintenabilité, qui sont de plus en plus inscrites dans le cahier de change.

Un équipement commence par une évolution peu de temps après sa mise en service. Ces caractéristiques, sa capacité à produire, la qualité de travail fournit, alors ses coûts d'exploitation peuvent diminuer. Il est donc indispensable d'agir pour le rendre conforme à un état initial. [4]

Plusieurs solutions s'offrent alors :

- Intervention du constructeur.
- Intervention d'une entreprise extérieure spécialisée.
- Intervention de service de maintenance interne à l'entreprise.

I.10. Les formes organisationnelles de la maintenance

La fonction maintenance est représentée de différentes manières dans les entreprises. Les principaux modèles organisationnels sont :

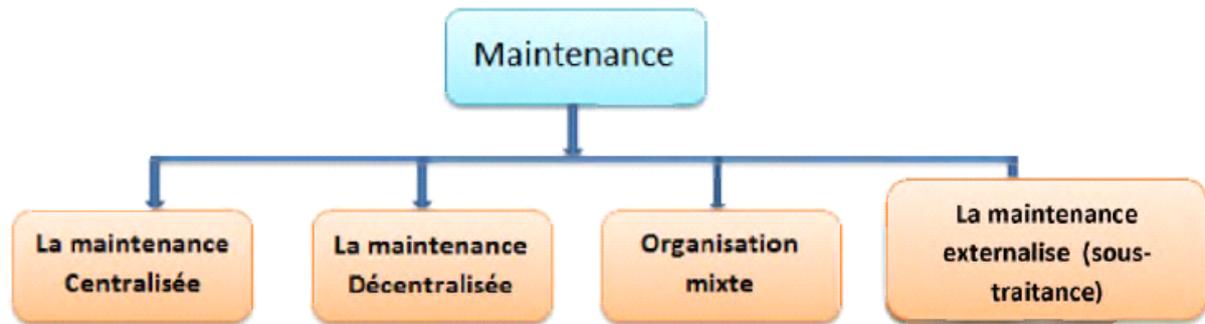


Figure I.6 : *Les formes organisationnelles de la maintenance. [3]*

I.10.1. La maintenance centralisée

C'est l'organisation traditionnelle de la maintenance distincte de la production et Regroupe tous les services techniques. Elle comprend un service méthode chargé de L'ordonnancement des travaux, de leur préparation, des équipes techniques d'intervention, des magasiniers...

Les avantages de ce type d'organisation sont : une maîtrise de la fonction et une maîtrise technique, une optimisation des efforts. Cependant on lui reproche parfois d'être cloisonnée et éloignée des préoccupations de la production.

I.10.2. La maintenance répartie ou décentralisée

Dans ce model organisationnel une partie de la maintenance dite maintenance rapprochée est intégrée aux équipes de production. Il y a donc une participation des agents de la production (technicien de plate-forme) sur la maintenance de l'outil de production.

On peut noter dans ce cas une meilleure prévention et une bonne maîtrise du processus de dégradation du matériel, une bonne collaboration des services maintenance et production.

I.10.3. Organisation mixte

C'est une organisation ou l'on retrouve une cohabitation des deux modèles susdits. Les processus de dégradation sont de mieux en mieux maîtrisés. Cependant elle requiert une Bonne définition des rôles des différentes équipes.

I.10.4. La maintenance externalisée (sous-traitance)

Faire de la sous-traitance une véritable externalisation de la maintenance, cela demande une parfaite maîtrise de toutes les étapes du processus : de l'identification des prestations au contrôle de l'exécution en passant par l'établissement de la relation contractuelle. Il existe quatre formes de sous-traitance :

- Sous-traitance partielle : le donneur d'ordre confie aux prestataires des travaux de maintenance bien définis car il n'a pas les moyens de les réaliser, révision d'un groupe électrogène...
- Sous-traitance totale : le donneur d'ordre confie aux prestataires l'ensemble des activités de maintenance d'un matériel bien défini : maintenance d'une machine spéciale...
- Sous-traitance ponctuelle : il s'agit d'une sous-traitance limitée dans le temps et par son Contenu à une ou plusieurs interventions : réparation d'un matériel.
- Sous-traitance continue : il s'agit d'une sous-traitance reconductible dans le temps et définie en général par un contrat de maintenance.

Notons que la sous-traitance d'une partie de ses activités de maintenance pour une Entreprise présente des avantages mais aussi des inconvénients.

- **Avantage** : des gains importants peuvent être obtenus par une diminution du nombre de corps de métier dans l'entreprise.
- **Inconvénient** : risque de perte de la maîtrise technique, de mobilité du personnel.

I.11. CONCLUSION

Nous pouvons conclure pour cette partie en disant que la maintenance subit des Évènements et se trouve parfois débordée par l'ampleur des actions à entreprendre.

C'est une solution où le long terme est sacrifié et semble plus rentable à court terme et tend à limiter le coût indirect sans vraiment y parvenir. En fait, les machines se dégradent rapidement, Les dépannages ne maintiennent pas les outils en état, le nombre de défaillances augmente et généralement, un bris grave vient conclure le processus.

Aussi l'accumulation des pertes de temps, le nombre important des rechanges à remplacer et l'immobilisation fréquente de la production qu'elle entraîne contribue à l'accroissement des charges à travers les coûts importants de ces bris et des coûts indirects dus à la perte de production. Par contre la maintenance préventive anticipe et intervient avant l'apparition de la panne ou la défaillance. Elle réduit ainsi considérablement les coûts et met les intervenants dans des conditions de travail moins stressantes ce qui est de nature à améliorer leurs efficacité et performance.

Chapitre II :
La gestion
maintenance assisté
par ordinateur

II.1. Introduction

Disposer d'un outil de Gestion de la Maintenance est aujourd'hui incontournable tant d'un point de vue technique, budgétaire qu'organisationnel pour optimiser la productivité des investissements des entreprises mais aussi pour garantir la disponibilité au moindre coût de la chaîne des moyens de production et logistiques.

Le marché des progiciels de gestion de maintenance est un marché mature qui propose des solutions capitalisant un fort savoir-faire. [07]

La GMAO fait partie du système d'information, de gestion et de pilotage de la fonction maintenance qui a pour mission de garder les installations dans un état tel qu'ils puissent constamment répondre aux spécifications pour lesquelles ils ont été conçus et ceci d'une manière efficace et économique. L'outil informatique de gestion est alors une aide pour tracer, archiver, analyser et prendre des décisions. [08]

II.2. Historique de l'évolution de l'informatique industrielle

Le paysage informatique des entreprises industrielles a été marqué par l'évolution parallèle de l'informatique pour l'ingénierie et de l'informatique pour la gestion, qu'il s'agisse de gestion de production ou de gestion administrative (ventes, personnel, comptabilité etc.). Ces applications ont été développées de manière indépendante, pour un domaine fonctionnel précis, sans avoir une vision intégrée des besoins. Les premiers outils développés pour les départements d'ingénierie ont été les outils de calcul et d'aide à la conception dans les années 60. Puis, dans les années 80, les systèmes de gestion des données techniques (SGDT) pour des départements d'ingénierie sont apparus. Ils étaient d'abord destinés à gérer les plans des pièces puis ils ont évolué pour gérer les données provenant des différents systèmes utilisés par les départements d'ingénierie, à savoir les calculs, les programmes de fabrication, les instructions de contrôle, les publications techniques, les notices etc.

L'informatique de gestion s'est développée depuis le début des années 60, évoluant depuis les systèmes de gestion de production MRP permettant la planification des besoins en composants aux systèmes de gestion de production MRPII permettant la planification de toutes les ressources associées à la production (Manufacturing Resource Planing). Aujourd'hui les ERP « Enterprise Resource Planing » ne se contentent plus de Gérer la production, ils intègrent des modules permettant de gérer l'ensemble de l'entreprise de façon intégrée. Les EPR

regroupent en effet des modules de gestion des achats et des ventes, de gestion du personnel, de comptabilité, de gestion de la qualité etc. [09]

II.3. Un progiciel de GMAO

En 1985 M.Gabriel et Y.Pimor définissaient la gestion de la maintenance assistée par ordinateur en ces termes : « Un système informatique de management de la maintenance est un progiciel organisé autour d'une base de données permettant de programmer et de suivre sous les trois aspects techniques, budgétaire et organisationnel. Toutes les activités d'un service de maintenance et les objets de cette activité (services, lignes, ateliers, machines, équipements, sous-ensembles, pièces, etc.) à partir de terminaux disséminés dans les bureaux techniques, ateliers, magasins et bureaux d'approvisionnement. » [10]

- GM (gestion de la maintenance) : c'est avant tout la compétence de l'acquéreur-utilisateur.
- AO (assistance informatique) : c'est la compétence du vendeur (qui n'ignore pas la maintenance, mais ne connaît pas votre entreprise).

Une GMAO investie est une «valise pleine d'informatique et vide de maintenance» il s'agit de la remplir, puis de la faire vivre à l'intérieur d'une organisation préalablement éprouvée. [2]

II.4. La GMAO un outil incontournable

Il existe des prérequis à l'acquisition d'une GMAO. Mais dès lors qu'un service maintenance est structuré et a fait la preuve de l'efficacité de son organisation, l'outil GMAO se révèle indispensable par sa capacité de mise en mémoire, par ses possibilités de traitement d'informations, par ses interfaces et par sa réactivité. Même dans une petite entreprise, la base de données atteint une taille impressionnante en «équivalent papier» ! Il suffit de penser au fichier des articles en stock, au fichier des fournisseurs, au parc matériel décomposé et au nombre d'interventions et de transactions effectuées chaque jour pour s'en convaincre.

D'autres facteurs rendent à terme l'exploitation de la GMAO incontournable :

- La «convivialité» des acteurs dans la cohérence du système : tout le service parle le même langage, l'information est partagée.
- La potentialité d'améliorations et d'optimisation : la GMAO est un vecteur de changement.
- La traçabilité des procédures et des actions requises en qualité (certifications ISO 9000).
- Le besoin de suivi des coûts et de contrôle économique.

II.5. La réussite d'une GMAO

Pour aboutir à la réussite du projet, l'implantation d'une GMAO nécessite obligatoirement au préalable une analyse fine et fouillée des besoins, une définition précise des objectifs, une préparation soignée des acteurs et l'adhésion de tous. En effet, une telle démarche doit se faire sous l'impulsion de la direction et la participation du personnel de l'entreprise. Ces prérequis permettront d'une part d'effectuer un choix pertinent parmi les propositions du marché et d'autre part de réfléchir aux organisations les plus intelligentes et les plus efficaces.

La décision d'investir dans un outil de GMAO remet en question les habitudes acquises par les différents services de l'entreprise qui utilisent pour des raisons historiques des systèmes de codages différents pour un même équipement en fonction de leur domaine de spécificité propre (achats, méthodes, travaux neufs, maintenance, conduite). L'évolution de l'offre GMAO et du marché va vers l'utilisation des technologies nomades, ce qui va entraîner une nouvelle remise en question des habitudes qui ont été prises depuis la première installation. [08]

II.6. Les conditions de réussite d'un GMAO

II.6.1. La convivialité de l'outil informatique

Le système va être donc utilisé par tout le personnel du service maintenance, et non réservé à certains. Il faut donc bannir l'ésotérisme, exigé absolument que la totalité de l'information soit disponible en langage connu, s'efforcer de trouver un bon compromis entre langage normalisé et terme de métier habituel de l'entreprise.

II.6.2. Volonté de réussite de consensus

C'est une question de motivation de la part de tous les intéressés, qu'il convient donc de faire participer à la définition générale du projet GMAO.

II.6.3. Facteur temps

Le facteur temps doit être pris en considération. La mise en place d'un système de GMAO peut durer jusqu'à 2 ans. [11]

II.7. Structure de la base de données maintenance

L'architecture type d'un progiciel de GMAO s'appuie sur un SGBD (Système de Gestion de Base de Données), qui assure les échanges et les traitements des données entre les différents modules d'une GMAO. Actuellement, les SGBD de type relationnel que nous trouvons sur le marché de la GMAO sont principalement :

- ORACLE.
- SYBASE.
- INGRES.
- INFORMIX.

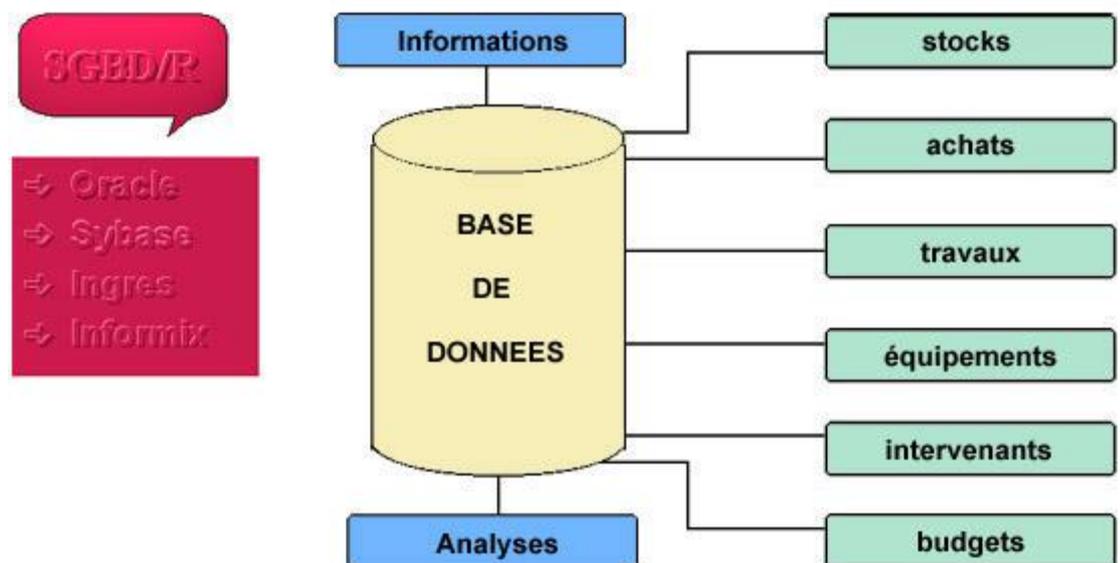


Figure II.1 : *Structure de SGBD [12]*

II.8. Les fonctionnalités du système GMAO

L'outil de GMAO se caractérise par quatre fonctionnalités standard :

- Gestion de la maintenance c'est-à-dire des interventions préventives et curatives sur les machines.
- Gestion du personnel de maintenance : planning, affectations aux personnes, gestion des formations (peu usité).
- Gestion des stocks de pièces détachées : contrôle des stocks en magasin, alertes sur seuil, réception de pièces.

- Gestion des achats : Edition des commandes, gestion des fournisseurs et de leur prix, facturation. [10]

Un recensement de l'offre commerciale en matière de logiciels de GMAO et d'aide à la maintenance fait ressortir environ 800 logiciels et progiciels de GMAO et d'aide diverses à la maintenance pour tous les secteurs industriels. [08]

II.9. Les progiciels de GMAO (analyse des différents modules fonctionnels)

Tous les progiciels de GMAO ont en commun la même structure modulaire proposant les mêmes fonctions. Mais, selon les logiciels, les fonctions remplies sont diversement dénommées, diversement réparties et diversement organisées. Prenons comme exemple Sirlog, la première GMAO développée en France et dont la composition reste représentative. [1]

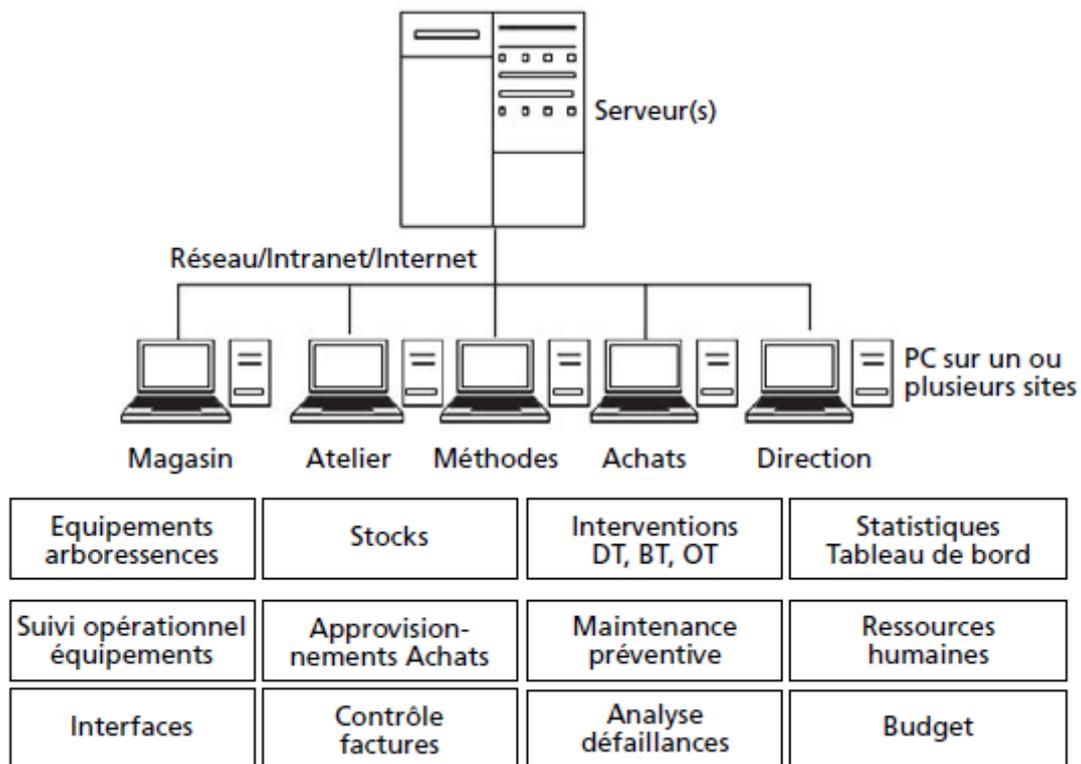


Figure II.2 : Exemple de structure modulaire d'une GMAO [1]

C'est dans les bureaux techniques (méthodes, ordonnancement, logistique et travaux neufs) que s'effectuera majoritairement la gestion par exploitation des 10 modules analysés.

Le « cahier des charges » proposé pour chaque module n'a pas l'ambition d'être exhaustif (chaque service maintenance a ses propres critères), mais d'attirer l'attention sur certains points souvent négligés. Les modules analysés sont les suivants :

1. gestion des équipements.
2. gestion du suivi opérationnel des équipements.
3. gestion des interventions en interne et en externe.
4. gestion du préventif.
5. gestion des stocks.
6. gestion des approvisionnements et des achats
7. analyses des défaillances.
8. gestion du budget et suivi des dépenses.
9. gestion des ressources humaines.
10. tableaux de bord et statistiques.
11. autres modules et interfaçages possibles.

II.9. 1 MODULE « GESTION DES EQUIPEMENTS »

Il s'agit de décrire et de coder l'arborescence du découpage allant de l'ensemble du parc à Maintenir aux équipements identifiés et caractérisés par leur DTE (dossier technique Équipement) et leur historique, puis à leur propre découpage fonctionnel. A partir du code Propre à l'équipement, le module doit permettre de :

- pouvoir localiser et identifier un sous-ensemble dans l'arborescence.
- connaître l'indice de criticité fonctionnelle de l'équipement, sa durée d'usage relevé par Compteur.
- accéder rapidement au « plan de maintenance » de l'équipement ;
- pouvoir trouver ses caractéristiques techniques, historiques et commerciales à partir du DTE.
- pouvoir localiser un ensemble mobile, trouver son DTE et son historique (gestion multi-site).
- connaître ses consommations en énergie, en lubrifiants, etc.
- connaître la liste des rechanges consommés ;
- connaître le code des responsables exploitation et maintenance de l'équipement ;
- accéder aux dessins et schémas relatifs à l'équipement contenus dans un logiciel de Gestion documentaire (hors DTE).

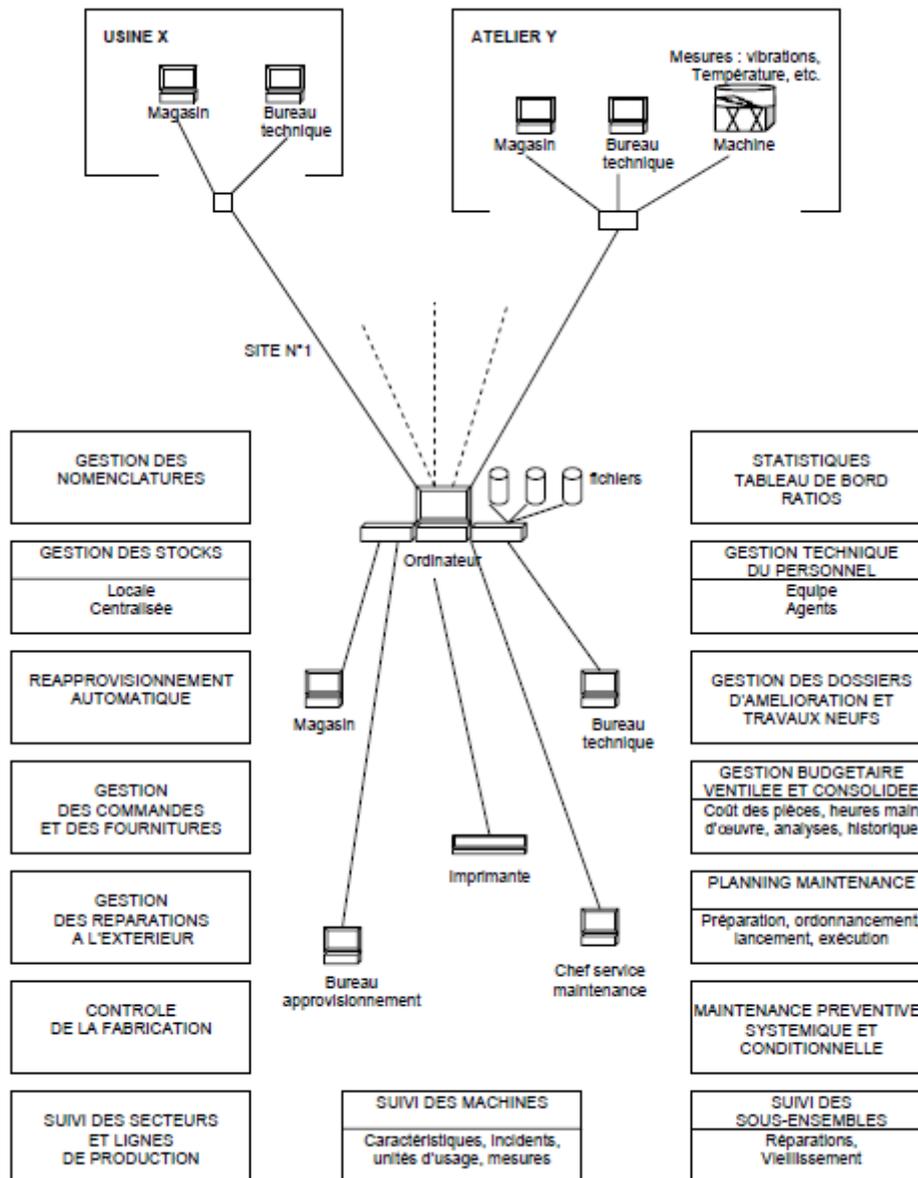


Figure II.3 : Exemple de structure modulaire d'une GMAO [1]

II.9.2 MODULE « GESTION DU SUIVI OPÉRATIONNEL DES ÉQUIPEMENTS »

A travers le module de suivi des performances d'un équipement, il s'agit de retrouver les Indicateurs de fiabilité, de maintenabilité, de disponibilité et le taux de rendement synthétique TRS si la TPM est envisagée ou effective. Le choix des indicateurs prédétermine la nature des Saisies nécessaires. Celles-ci doivent pouvoir se faire « au pied de la machine » et en temps Réel, aussi bien en ce qui concerne les demandes que les comptes rendus.

II.9.2.1 Dans le cadre d'un suivi technique par l'indicateur Disponibilité

Le module doit être capable d'assurer la gestion en affichant :

- les graphes d'évolution des Di par périodes de suivi ;
- les graphes de Pareto se rattachant aux équipements par nature des arrêts ;
- le rappel des valeurs des indicateurs MTA (moyenne des temps d'arrêt) ou MTTR (mean time to repair , en français TTR : temps technique de réparation) pour les dernières périodes.

II.9.2.2 Dans le cadre d'un suivi par le TRS

Le module doit être capable, à partir des données opérationnelles liées aux pertes ; de Performances, aux pertes de qualité et aux pertes de disponibilité, de calculer les trois taux et Leur produit (le TRS) par période, de montrer leur évolution, de présenter l'affichage Analytique des valeurs après sélection, pour diagnostic. De façon plus générale, l'agent des Méthodes doit être capable de trouver à travers ce module tous les éléments quantitatifs lui Permettant d'approfondir une analyse de logistique, de fiabilité, de maintenabilité ou de Disponibilité.

II.9.3 MODULE « GESTION DES INTERVENTIONS »

Nous avons vu en ordonnancement l'existence de plusieurs procédures adaptées à la nature Des travaux. Pour les nombreux petits travaux, pas de demande DT (demande de travail) ni D'attribution de numéro, mais un enregistrement rapide a posteriori de leur durée, de leur Localisation et de leur nature. Il est nécessaire de créer une bibliothèque des différents codes Utiles afférents aux clients, aux intervenants, aux différents statuts de l'intervention. D'autre Part, à chaque équipement doit correspondre une bibliothèque de codes standards, relatifs au Découpage de l'équipement, à l'effet déclenchant (souvent appelé par erreur « cause » d'arrêt) Et à la cause identifiée.

II.9.3.1 Pour les DT, demandes de travaux

Le module doit permettre :

- la création d'un numéro DT, OT, qui servira de référence pour toutes les opérations liées, procédures de sécurité spécifique, préparation et DA (demande d'approvisionnement) ou BSM (bon de sortie magasin) par exemple ;

- l'horodatage de la demande, avec identification du demandeur et du secteur (code client) et l'urgence ou le délai attribué ;
- le suivi possible du statut de la demande par le demandeur (code des différents statuts).

II.9.3.2 Au niveau de la préparation de l'OT

Le module doit permettre :

- l'insertion de gammes de maintenance préétablies ;
- les réservations d'outillages, de moyens spéciaux, de pièces de rechanges, etc. ;
- l'affectation des ressources ;
- le regroupement de la gamme de maintenance avec des plans, des pictogrammes et des schémas extraits d'un logiciel de gestion documentaire ;
- l'insertion automatique de procédures de sécurité liées à certains secteurs ou à certains équipements ;
- l'intégration d'un groupe de travaux à un gestionnaire de projet, avec graphismes Gantt et PERT

II.9.3.3 Pour les comptes rendus d'intervention

Le module doit permettre :

- la saisie « facile et rapide » (critère très important) des paramètres et de la caractérisation de l'intervention, même et surtout s'il s'agit d'une correction de micro-défaillance ;
- l'utilisation par les dépanneurs d'une borne en libre-service, située à proximité immédiate du site d'intervention, réduisant ainsi les distances et les temps de saisie d'intervention ;
- de caractériser l'intervention par les codes de la bibliothèque de l'équipement (localisation, cause, etc.) ;
- l'imputation des travaux à des comptes analytiques ;
- de distinguer les durées d'intervention des durées d'indisponibilité ;
- d'enrichir chronologiquement l'historique de l'équipement dès la clôture de l'OT ;
- de connaître les consommations de pièces utilisées, éventuellement leurs valeurs ;
- la rédaction d'un texte libre contenant les remarques et les suggestions de l'intervenant. Par contre, il ne doit pas donner l'impression d'une « inquisition », mais

d'un besoin de savoir pour mieux comprendre et améliorer avec l'aide du technicien d'intervention.

II.9.3.4 Pour la gestion des travaux externalisés

Le module doit permettre une gestion semblable aux procédures de préparation et D'ordonnancements internes :

- émission de DTE (demande de travaux externalisés) pour les prestations ponctuelles
- création de contrats-type (clauses techniques, économiques et techniques, plan de sécurité) qu'il suffit d'adapter à chaque commande.

II.9.4 MODULE « GESTION DU PRÉVENTIF »

Le module permettra de gérer la maintenance systématique à travers un planning Calendaire par équipement, les dates étant prédéterminées ou déterminées à partir d'un relevé De compteur (ou d'une mesure dans le cas de la maintenance conditionnelle). Le Déclenchement sera automatique, par listing hebdomadaire des opérations prévues dans la Semaine. Chaque opération sera définie par sa gamme préventive. Le module devra aussi Permettre un déclenchement « manuel d'opportunité », par exemple par anticipation d'une Opération préventive à la suite d'un arrêt fortuit.

II.9.5 MODULE « GESTION DES STOCKS »

Le système repose sur le « fichier des articles » en magasin comprenant les « lots de Maintenance » par équipement et sur les mouvements entrées/sorties du magasin. Une fiche Article doit comprendre :

- le code article défini par l'organisation interne, son libellé et sa désignation technique ;
- le code article du ou des fournisseurs et le code fournisseur (et fabricant éventuellement)
- le code du gisement en magasin ;
- les codes des articles de substitution, en cas de rupture ;
- le rattachement aux équipements possédant cet article ;
- le prix unitaire et le prix moyen pondéré automatiquement calculé ;
- les quantités en stock, commandées en attente ;
- la méthode de réapprovisionnement et ses paramètres (stock de sécurité, stock maxi. etc.) ;
- les dates des derniers mouvements ;

- l'historique des consommations.

Les outils d'analyse du stock en nature et en valeurs sont :

- le classement des articles en magasin par valeurs et par taux de rotation ;
- la valeur des stocks par nature et par périodes (mois par mois) ;
- la liste des articles « dormants » ;
- la liste des cas de ruptures de stock (demandes non satisfaites).
- Il importe de vérifier certaines potentialités du module :
- la possibilité ou non d'actualisation automatique des paramètres en fonction des Consommations ;
- la possibilité d'avoir le profil des consommations et le tracé de la courbe ABC en valeurs
- les possibilités relatives aux transactions du magasin : réceptions provisoires ou Définitives, retours au fournisseur en cas de non-conformité, etc. ;
- l'édition de pièces réservées sur une préparation (numéro d'OT pour l'imputation) ;
- la présence d'un écran d'inventaire comprenant les différents critères d'article ;
- la possibilité d'effectuer des recherches et des analyses multicritères.

II.9.6 MODULE « GESTION DES APPROVISIONNEMENTS ET DES ACHATS »

Caractéristiques de la fonction en maintenance : beaucoup de références et de fournisseurs Pour des quantités faibles et des délais courts. Ce module doit permettre, en interface avec le Logiciel du service « achat » de maîtriser et de gérer avec aisance :

- le fichier des fournisseurs et des fabricants avec leurs tarifs liés aux quantités.
- le lancement d'appels d'offre aux fournisseurs.
- l'édition de bon de commandes standard ou personnalisés, et le suivi des autorisations de dépenses.
- le contrôle des factures.
- l'édition automatique des codifications internes et fournisseurs (transcodage).
- le suivi des états de la commande.
- le suivi des réceptions totales, partielles et des refus.
- l'estimation de la qualité des fournisseurs par les contrôles de réception et le suivi des délais.
- l'édition automatique de lettres de relance pour les retards.

II.9.7 MODULE « ANALYSES DES DÉFAILLANCES »

La base de ce module est constituée des historiques automatiquement alimentés par Chaque saisie de BPT (bon de petits travaux) et d'OT (ordre de travail) mis en famille par ses codes d'imputation. A partir d'un équipement donné, il doit permettre :

- l'établissement des analyses quantitatives par graphes de Pareto, avec plusieurs critères (MTTR, T_A ; durée d'arrêt) et plusieurs mises en familles (par cause, par localisation, par Nature de défaillance, etc.) Et sur plusieurs périodes d'analyse (hier, la semaine écoulée, les Trois derniers mois, l'année, etc.) ;
- puis l'analyse qualitative des défaillances sélectionnées comme prioritaires, Éventuellement mise sous forme AMDEC.

La productivité de l'analyse de défaillance comme outil de progrès rend cette fonction de GMAO stratégique : il est indispensable de savoir par qui, quand, comment vont être Organisées ces analyses pour tester l'adéquation du logiciel au cahier des charges du module. Ce module est la base de la MBF (maintenance basée sur la fiabilité).

II.9.8 MODULE « BUDGET ET LE SUIVI DES DÉPENSES »

La gestion analytique ne permet que des « micro-analyses » des comptes. Un découpage Plus fin de la fonction maintenance doit donc pouvoir permettre des analyses détaillées grâce À la GMAO, l'objectif étant le suivi de l'évolution des dépenses par activité dans un budget Donné. Quelques éléments du cahier des charges à préciser, c'est-à-dire le module permet-il :

- la création d'un nouveau budget en modifiant des chapitres de l'ancien ?
- la comparaison entre plusieurs exercices ?
- la prise en compte des frais généraux du service ?
- l'éclatement en coûts directs et indirects (pertes de qualité, de production, etc.) ?
- la ventilation des coûts par équipement, par «client», par type d'activité de maintenance, Par origine de défaillance, par sous-ensemble «fragile» communs à plusieurs équipements, Etc. ?
- la comparaison entre la prévision et la réalisation ?
- la gestion en plusieurs devises : francs, euros, dollars, etc. ?
- la possibilité d'exporter les résultats comptables sur un logiciel de comptabilité ?
- la décomposition structurelle du budget en sous-budgets consolidables ?
- le suivi des coûts pour établir le LCC (life cycle cost ; coût du cycle de vie) d'un Équipement ?

II.9.9 MODULE « GESTION DES RESSOURCES HUMAINES »

Spécifiquement adapté au service maintenance, ce module sera principalement une aide à l'ordonnancement. Il sera construit autour d'un « fichier-technicien » pouvant comprendre, Pour chacun :

- la qualification, les habilitations, les diplômes, l'ancienneté dans son échelon actuel, les différentes affectations, l'affectation actuelle, etc. ;
- les formations suivies, demandées et le bilan de compétence ;
- les congés pris, demandés et les récupérations (données nécessaires à la programmation des travaux) ;
- les temps de présence et d'absence (historique des arrêts de travail) ;
- les coûts horaires pour chaque qualification (pour imputation des coûts d'intervention).

Remarquons l'intérêt, pour chaque technicien, de pouvoir accéder par la GMAO, à partir Du terminal atelier, à ses propres informations relatives aux reliquats des congés à prendre ou À des informations générales de l'entreprise. C'est un facteur d'acceptation du système Informatique.

II.9.10 MODULE « TABLEAUX DE BORD ET STATISTIQUES »

Les tableaux de bord concernent la mise en forme de tous les indicateurs techniques, Économiques et sociaux sélectionnés pour assurer la gestion et le management du service Maintenance. Certains sont livrés en « standard » avec le logiciel. Il faut vérifier s'ils peuvent Être personnalisés rapidement (courbes, graphiques et autres visuels), ou développés avec un Générateur d'état extérieur au logiciel. Vérifier également que l'extraction de données se fait Simplement.

En cas de projet TPM, il faut vérifier la possibilité de former l'indicateur TRS et de visualiser ses variations par périodes.

II.9.11 MODULE COMPLÉMENTAIRES OU INTERFAÇAGES UTILES

La revue des besoins internes et externes du service peut amener à rechercher des Extensions par interfaçage, par acquisition de modules complémentaires ou par Développement de logiciels applicatifs spécifiques. Interfaçage requis ou non avec :

- le logiciel de comptabilité et de paie,
- le logiciel de gestion des ressources humaines,
- le logiciel de gestion des achats et approvisionnements,

- les outils multimédias,
- la supervision : saisie automatique de données « machines » par collecteur portable, par Code-barres, par automates ou par capteurs.

Autres fonctionnalités possibles :

- liaison avec le logiciel de gestion de projet,
- lecteur de badges,
- saisie des images : scanner, hypertexte, etc.,
- analyses de pannes, génération d'AMDEC, etc.

Toutes ces potentialités étant très évolutives, il importe de ne pas prendre de retard au Départ d'un projet GMAO, qui doit déboucher sur une durée d'exploitation significative pour Se justifier économiquement, sans pour cela aller au superflu.

II.10. LE CHOIX D'UN OUTIL GMAO BIEN ADAPTÉ

Il appartient à chaque service maintenance de déterminer ses besoins internes en matière D'informatisation, mais également ses besoins de communication externes, présents et avenir. Cette réflexion doit se faire dans la cohérence du programme d'informatisation de l'entreprise, À l'horizon 5 à 8 ans, en pensant que si 35 % seulement des potentialités d'une GMAO sont exploitées (surestimation des besoins), l'exploitation de certaines GMAO doit être abandonnée, par sous-estimation des besoins, souvent faute d'être compatibles avec les Nouvelles organisations de l'entreprise. Le choix d'un outil GMAO passe par son adéquation :

- à la stratégie globale du système informatique de l'entreprise : problème de l'intégration
- aux besoins exprimés du service maintenance : problème du cahier des charges et Problème du paramétrage (personnalisation).

II.10.1 Intégration de la GMAO dans le système d'information de l'entreprise

La réduction de l'hétérogénéité des matériels, des langages et des systèmes d'exploitation. La suppression des redondances et les doubles saisies passent par l'intégration de la GMAO à la cohérence d'un système informatique global. Deux types d'intégration sont possibles :

- à partir d'une base de données « entreprise », autour de laquelle les fonctions comptabilité, personnel, commercial, production et maintenance échangent et communiquent ;
- à partir d'un système global d'exploitation, architecture du site (suivi de production assuré à partir de toutes les données opérationnelles du terrain) jusqu'à un tableau de

bord de pilotage de la production. Ce type d'intégration se prête bien à la stratégie TPM par suivi de la maintenance de premier niveau et analyses des défauts, pertes de cadences et micro défaillances mesurées par le TRS. Aux autres niveaux, les GPAO et GMAO retrouvent leurs spécificités.

II.10.2 Importance du paramétrage : la « flexibilité » d'une GMAO

L'outil GMAO doit proposer des propriétés de modules et un paramétrage pour pouvoir S'adapter à l'entreprise, à son organisation, à son évolution prévisible et à son vocabulaire. Et Non l'inverse. L'aspect paramétrage des produits GMAO s'impose, permettant à l'utilisateur de Modeler ses interfaces au logiciel suivant ses besoins propres. Une gestion de configuration Doit permettre des ajouts ou des suppressions de champs, des calculs sur les champs et des Modifications de libellés. La GMAO doit permettre une extraction rapide de données Permettant de personnaliser des indicateurs.

II.10.3 Importance du cahier des charges

Les fournisseurs de GMAO, en 1998, estimaient qu'une moyenne de 35 % seulement des Potentialités des GMAO vendues sont exploitées : ce qui peut poser le problème de L'adéquation de l'offre et de la demande, mais plus sûrement le problème de la définition Précise des besoins au moyen d'un bon cahier des charges. L'exposé précédent peut permettre « Une revue de critères » à l'appui de la démarche interne de rédaction d'un cahier des charges, Préalable indispensable à l'acquisition d'une GMAO et à son acceptation par les acteurs de son Exploitation. Il apparaît que l'acquisition courante de « petits logiciels » est en fait une Solution d'attente, « de crainte de se tromper », faute de lisibilité suffisante de l'avenir de L'entreprise et du service, de l'évolution du marché et des produits, faute de cahier des charges Suffisant et faute de préparation des acteurs.

II.11. LA CONDUITE D'UN PROJET GMAO

II.11.1 IMPORTANCE DE L'ASPECT HUMAIN DANS LA RÉUSSITE DU PROJET

Le projet GMAO est pour le service maintenance un projet « structurant » remettant en Cause des habitudes de travail, donc susceptible de modifier en profondeur l'état d'esprit et la Motivation des acteurs. C'est une opportunité pour réorganiser un secteur, pour optimiser des Procédures, pour élever le niveau de sensibilité à la gestion de tous les acteurs, pour en

Promouvoir certains. Il ne faut pas négliger ni sous-estimer le poids de la formation dans le Coût du projet : l'acceptation de l'outil est la condition incontournable de réussite du projet, Son refus la cause majoritaire d'échec. Or il ne peut y avoir d'acceptation sans une stratégie de Formation adaptée au niveau de départ des techniciens.

II.11.2 ÉTAPES DU PROJET

Elles seront différentes suivant que le projet est « intégré » à un programme informatique Conduit au niveau de l'entreprise, ou qu'il est « autonome » car conduit au niveau du seul Service maintenance. Dans ce cas, la maintenance aura davantage de liberté, mais aussi le Poids de la maîtrise du projet. Nous nous placerons dans ce cas, qui implique la nomination D'un chef de projet interne, détaché à plein temps pendant une période voisine d'une année Suivant l'ambition du projet. Les étapes seront également différentes suivant qu'il s'agit de « Démarrer » une première GMAO ou de renouveler une ancienne GMAO, ce qui pose dans Ce cas le problème de récupération des données et du « basculement » de l'ancienne sur la Nouvelle.

II.11.2.1 Préalables

Dans tous les cas, comme pour tout projet d'ailleurs, une forte implication de la direction Est indispensable. Elle se manifestera par un plan de communication interne, la rédaction de Directives encadrant le projet, dont le partage entre les ressources internes et externes allouées. La nomination du chef de projet entouré à temps partiel d'un groupe de pilotage et L'affectation de moyens (salle de travail équipée en informatique) est indispensable. D'autre Part, rappelons que le miracle assisté par ordinateur n'aura pas lieu : une GMAO n'est qu'un Outil, certes structurant, mais incapable d'organiser un service. L'implantation ne peut se Réussir qu'à partir d'une organisation ayant fait auparavant la preuve de son efficacité. C'est à Partir de cette organisation existante que seront étudiés les éléments du cahier des charges.

II.11.2.2. Rédaction du cahier des charges de consultation

Il ne suffit pas d'établir un cahier des charges technique et gestionnaire de la fonction Mais de prendre en compte des critères :

- d'intégration immédiate et à moyen terme dans l'informatique de l'entreprise ;
- de qualité du conseil, du service client et de l'assistance ;
- de pérennité du fournisseur et du produit ;

- de transferts de compétence : du vendeur au client de l'installateur à l'utilisateur pour une autonomie rapide ;
- de formation : plan de formation quantitatif et qualitatif ;
- d'évolution, d'interfaçage et de paramétrage ;
- de convivialité et d'ergonomie ;
- d'implantation (exemple d'une borne tactile en libre-service à proximité des dépanneurs).

L'idée du «juste nécessaire» doit éviter d'investir dans des fonctions inutiles, inadaptées, Superflues et coûteuses. L'expérience montre qu'il vaut mieux chercher à dégonfler L'enveloppe budgétaire par cette recherche du juste nécessaire que sur la formation et L'assistance.

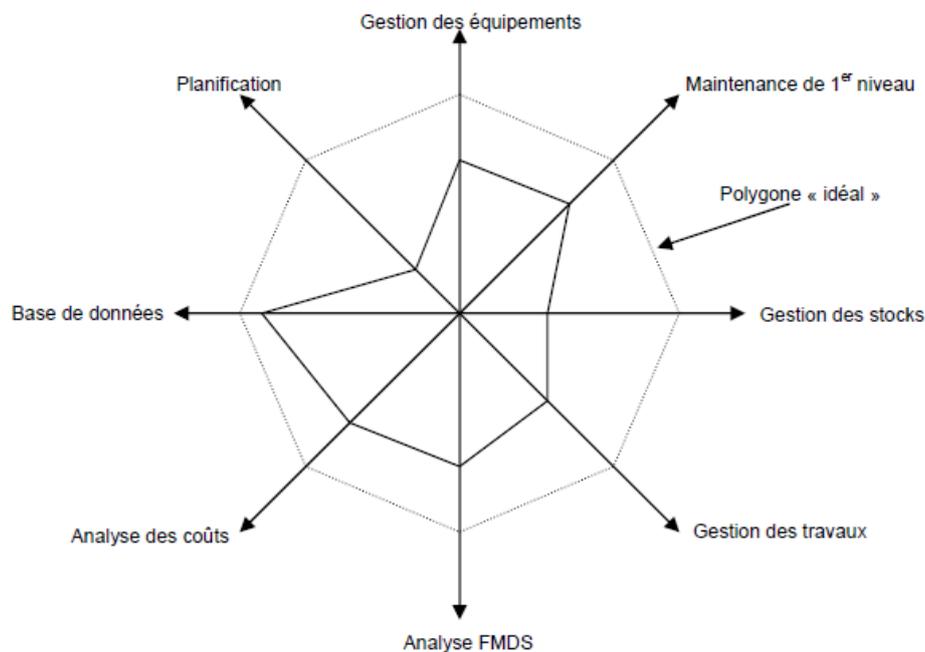


Figure II.4 : fonction de la maintenance [1]

II.11.2.3. Choix de l'outil GMAO et des modules nécessaires

A partir d'un problème bien posé, la réponse est supposée aisée. Il reste donc à passer un Appel d'offre détaillé (dimensionnement, technologies, nombre de terminaux, cahier des Charges fonctionnel, etc.), puis à effectuer les essais sur les deux ou trois produits Présélectionnés. Après démonstrations, tests et jeux d'essais de chaque module, il sera Possible d'évaluer chaque logiciel à partir de critères de choix pertinents et rigoureux. Le Choix définitif étant réalisé, une négociation est toujours souhaitable avant de passer la Commande.

II.11.2.4. Implantation, plan de formation et démarrage

Le moment de l'implantation doit être bien choisi, en dehors d'une période de forte Activité, et doit être précédé d'une forte information, la crainte « a priori » de l'informatique Étant toujours forte. Cette information doit porter à la fois sur les objectifs généraux de L'informatisation, sur les caractéristiques de l'outil sélectionné et sur le rôle de chaque acteur.

La formation peut alors débiter, avec deux personnes par poste au maximum, sous forme De travaux dirigés sur maquette. Des groupes professionnels homogènes seront constitués par Sites (agents des méthodes, maîtrise, techniciens et dépanneurs, magasiniers et responsables Des approvisionnements) de façon à personnaliser les paramétrages des différents modules en Fonction de l'organisation interne voulue. A chaque groupe correspondra un module Spécifique de formation. Pour chaque groupe, il faut créer les codes, les accès et les sécurités (Mots de passe), définir les options autorisées, les accès à la base de données, etc.

Le découpage topologique du site et le découpage fonctionnel des équipements seront Réalisés par les agents des méthodes, assistés par le conseil-fournisseur pour ce qui concerne La codification.

Le conseil est aussi souhaitable pour l'ordre de réalisation des nomenclatures et des Saisies, variable selon les logiciels. Selon la qualité de la préparation des acteurs et suivant L'ambition du projet, les premiers résultats positifs se manifesteront en quelques semaines, L'ensemble du projet durant de six mois à deux ans pour des logiciels à haut degré D'intégration nécessitant un paramétrage lourd.

II.11.3. QUELQUES CAUSES D'ÉCHEC

Toutes les données récentes émanant des fournisseurs comme des utilisateurs de GMAO Convergent : le taux d'échec total ou partiel de l'implantation d'une GMAO est élevé, puisque 30 % des projets avortent.

Certains ne remplissent pas les fonctions prévues, d'autres entraînent des dépassements Considérables de budget. Ce taux d'échec doit donner à réfléchir, concernant un outil Stratégique et incontournable à terme. Listons quelques causes qu'il vaut mieux identifier pour Mieux les contourner.

II.11.3.1. Insuffisance de la prise en compte des facteurs humains

N'ayons aucune illusion : l'implantation d'une GMAO aura ses détracteurs a priori. Et ils

Seront d'autant plus nombreux que le projet sera imposé. D'où la nécessité d'un plan de Communication précédant un plan de formation pour éviter une dynamique de rejet.

II.11.3.2. Insuffisance de l'organisation initiale

Nous avons déjà vu que l'objectif de l'implantation d'une GMAO n'est pas de mettre de L'ordre : il faut savoir que là où la technicité est insuffisante, là où l'organisation est inefficace Et là où le climat social est dégradé, l'implantation est vouée à l'échec et que le remède GMAO sera pire que le mal initial. Un regard extérieur et un audit de la fonction doivent Pouvoir dissuader de se lancer dans l'aventure.

II.11.3.3. Le projet est mal piloté, il y a confusion et absence d'objectifs clairs

Les objectifs du projet doivent être clairement identifiés par tous. Si l'opération est mal Préparée, si, au nom du consensus, tout le monde veut développer son idée, alors la cohérence Sera perdue. Rappelons que ce projet n'est jamais une fin en soi, mais seulement un outil au Service d'un projet global d'amélioration de l'efficacité de la maintenance.

II.11.3.4. Le projet est vu sous son seul aspect technique

Lorsque les acteurs, et spécialement l'encadrement, ne sont pas suffisamment sensibilisés À la gestion économique, l'utilisation de l'outil risque de dériver vers la seule maîtrise Technique des événements, qui ne permettront pas un bon retour sur investissement, la réduction des coûts de maintenance étant un des éléments clés du projet.

II.11.3.5. les difficultés de démarrage et de formation sont sous-évaluées

Les vendeurs de GMAO, pour des raisons commerciales, ont parfois tendance à sous-estimer les difficultés, les temps et les coûts de démarrage et de formation.

II.11.3.6. Le « juste nécessaire » est surévalué

Lorsque les conditions d'un sympathique dynamique collectif sont créées, l'expression Des besoins de chacun amène naturellement à une surabondance de demandes qu'il faudra Tempérer par un arbitrage dans le respect de l'enveloppe allouée.

II.11.3.7. L'exploitation de la GMAO est insuffisante

Lorsque le système de gestion est opérationnel, il est mis à la disposition des « hommes de L'art ». Encore faut-il que ces derniers sachent mettre l'outil à disposition de l'optimisation de

La fonction maintenance. Car n'oublions pas, pour conclure, que la seule justification de L'investissement GMAO est l'analyse pertinente des données aux fins de propositions D'amélioration permanente de la maintenance.

II.12. Conclusion

Dans ce chapitre, il était question des généralités sur la GMAO. En effet nous y avons ressort une ébauche de définition général d'un progiciel de GMAO pour en déduire les conditions de réussite. Nous avons ensuite présentée la structure de la base de données maintenance. C'est pourquoi nous allons discuter.

Dans le chapitre suivant la relation entre l'informatique et la maintenance.

Chapitre III :
Machine learning,
deep learning et
réseaux de neurones

III.1. Introduction

l'intelligence artificielle (IA) est un ensemble de techniques permettant à des machines d'accomplir des tâches et de résoudre des problèmes normalement réservés aux humains et à certains animaux.

Les tâches relevant de l'IA sont parfois très simples pour les humains, comme par exemple reconnaître et localiser les objets dans une image, planifier les mouvements d'un robot pour attraper un objet, ou conduire une voiture et des fois beaucoup plus complexe. C'est grâce à l'apprentissage qu'il pourra apprendre à exécuter de nouvelles tâches et acquérir de nouvelles compétences. Le domaine de l'IA n'a pas toujours considéré l'apprentissage comme essentiel à l'intelligence. Par le passé, construire un système intelligent consistait à écrire un programme « à la main » pour jouer aux échecs (par recherche arborescente), reconnaître des caractères imprimés (par comparaison avec des images prototypes), ou faire un diagnostic médical à partir des symptômes (par déduction logique à partir de règles écrites par des experts). Mais cette approche « manuelle » à ses limites.

Les méthodes manuelles dans l'apprentissage machine se sont avérées très difficiles à appliquer pour des tâches en apparence très simples comme la reconnaissance d'objets dans les images ou la reconnaissance vocale. Les données venant du monde réel – les échantillons d'un son ou les pixels d'une image – sont complexes, variables et entachées de bruit. Pour une machine, une image est un tableau de nombres indiquant la luminosité (ou la couleur) de chaque pixel, et un signal sonore une suite de nombres indiquant la pression de l'air à chaque instant. Comment une machine peut-elle transcrire la suite de nombres d'un signal sonore en série de mots tout en ignorant le bruit ambiant, l'accent du locuteur et les particularités de sa voix ? Comment une machine peut-elle identifier un chien ou une chaise dans le tableau de nombre d'une image quand l'apparence d'un chien ou d'une chaise et des objets qui les entourent peuvent varier infiniment ?

On va essayer dans ce chapitre de répondre à ces questions et de connaître les types d'apprentissage machines et leur différence.

III.2. Machine Learning (ML)

III.2.1. Définition

Si le Machine Learning ne date pas d'hier, sa définition précise demeure encore confuse pour de nombreuses personnes. Concrètement, il s'agit d'une science moderne permettant de découvrir des patterns et d'effectuer des prédictions à partir de données en se basant sur des

statistiques, sur du forage de données, sur la reconnaissance de patterns et sur les analyses prédictives. Les premiers algorithmes sont créés à la fin des années 1950. Le plus connu d'entre eux n'est autre que le Perceptron.

Le Machine Learning est très efficace dans les situations où les insights doivent être découvertes à partir de larges ensembles de données diverses et changeantes, c'est à dire : le Big Data. Pour l'analyse de telles données, il se révèle nettement plus efficace que les méthodes traditionnelles en termes de précision et de vitesse. Par exemple, pour en se basant sur les informations associées à une transaction comme le montant et la localisation, et sur les données historiques et sociales, le Machine Learning permet de détecter une fraude potentielle en une milliseconde. Ainsi, cette méthode est nettement plus efficace que les méthodes traditionnelles pour l'analyse de données transactionnelles, de données issues des réseaux sociaux. [22]

III.2.2. Les principales méthodes de machine learning

Les deux méthodes de Machine Learning les plus couramment utilisées sont l'apprentissage supervisé et l'apprentissage non supervisé. Les algorithmes d'apprentissage supervisé sont entraînés à l'aide d'exemples étiquetés. Par exemple, un appareil peut avoir des points de données étiquetés F (failed) ou R (runs). L'algorithme reçoit un ensemble d'inputs ainsi que les outputs corrects correspondants, et apprend en comparant les outputs avec les résultats corrects attendus pour détecter les erreurs. Il modifie ensuite son modèle en fonction. Les méthodes comme la classification, la régression, et prédiction permettent à l'apprentissage supervisé d'utiliser des patterns pour prédire la valeur d'une étiquette ou d'une donnée additionnelle sans étiquette. Cette méthode d'apprentissage est couramment utilisée dans les applications où les données historiques permettent de prédire les événements futurs. Par exemple, elle permet d'anticiper les transactions frauduleuses ou les risques qu'un client d'une assurance ait un accident.

L'apprentissage non supervisé est utilisé pour les données qui n'ont pas d'étiquettes historiques. Le système ne connaît pas la réponse correcte, et l'algorithme doit comprendre par lui-même ce qui lui est présenté. L'objectif est d'explorer les données et de trouver une structure en leur sein. Cette méthode fonctionne bien pour les données de transaction. Par exemple, elle permet d'identifier des segments de consommateurs dotés d'attributs similaires pouvant être traités de façon similaire dans le cadre d'une campagne marketing. Elle permet également de trouver les principaux attributs qui séparent différents segments de consommateurs. Ces

algorithmes sont utilisés pour segmenter les textes, pour recommander des produits et pour identifier des sources de données.

L'apprentissage semi-supervisé est utilisé pour les mêmes applications que l'apprentissage supervisé. Tous deux utilisent à la fois des données étiquetées et non-étiquetées pour s'entraîner. En règle générale, une petite quantité de données étiquetées est utilisée avec une grande quantité de données non étiquetées. Pour cause, les données non étiquetées sont moins chères et plus faciles à obtenir. Ce type d'apprentissage peut être utilisé avec des méthodes comme la classification, la régression et la prédiction. L'apprentissage semi-supervisé est utile quand les coûts associés à l'étiquetage sont trop élevés pour permettre un processus d'apprentissage entièrement étiqueté. Il est par exemple utilisé pour identifier le visage d'une personne sur une webcam.

L'apprentissage de renforcement est souvent utilisé pour la robotique, le jeu vidéo et la navigation. Grâce à l'apprentissage de renforcement, l'algorithme multiplie les tentatives pour tenter de découvrir quelles actions apportent les plus grandes récompenses. Ce type d'apprentissage regroupe trois principaux composants : l'agent (qui apprend ou prend les décisions), l'environnement (tout ce avec quoi l'agent interagit), et les actions (ce que peut faire l'agent). L'objectif est que l'agent choisisse les actions qui maximisent les récompenses attendues sur une période donnée. L'agent atteindra ce but plus rapidement en suivant des règles bien établies. L'objectif de l'apprentissage de renforcement est donc d'apprendre les meilleures règles.

III.2.3. Champs d'application du Machine Learning

III.2.3.1. Marketing en ligne

Quelles mesures marketing apportent des résultats ? Les humains sont généralement mauvais pour passer en revue de grandes quantités de données et fournir des estimations fiables. Dans ce cas, il vaut mieux utiliser des outils d'analyse marketing qui s'appuie sur le Machine Learning. Ils évaluent des données définies et peuvent fournir des diagnostics fiables à propos du type de contenu capable d'aboutir à une conversion, des contenus que les clients veulent lire et des canaux marketing les plus efficaces pour conclure une vente.

III.2.3.2. Support client

Les chatbots peuvent s'appuyer sur le Machine Learning. Ils s'orientent en fonction des mots-clés trouvés dans la question de l'utilisateur et, par des questions pour obtenir plus

d'informations ou prendre des décisions, dialoguent avec l'utilisateur jusqu'à lui apporter la réponse désirée.

III.2.3.3. Vente

Ce qui fonctionne pour Netflix et Amazon est aussi idéal pour la vente. Grâce au Machine Learning, les systèmes peuvent anticiper avec précision les produits et services qui pourraient intéresser les clients sur leur site. Ils peuvent ainsi faire des recommandations détaillées, ce qui facilite la vente avec des gammes de produits très large ou des produits hautement personnalisables.

III.2.3.4. Informatique décisionnelle

Le Machine Learning peut aussi servir à visualiser les données importantes de l'entreprise et à rendre différentes prévisions compréhensibles pour les décideurs humains.

III.2.4. L'avantage du Machine Learning

Le Machine Learning nous a permis de réaliser des algorithmes et de résoudre des problèmes trop complexes pour les capacités humaines. Il a pour principal rôle de décharger la complexité des problèmes auprès des Hommes. Cette technologie donne aux ordinateurs la capacité d'apprendre. [23]

III.2.5. Les limites du Machine Learning

Le Machine Learning ne donne pas toujours des résultats fiables, car il est très difficile pour une machine de reconnaître parfaitement un élément. C'est le cas notamment pour la reconnaissance vocale. En effet, les voix, les intonations, les accents ou les expressions sont trop changeantes. C'est ici qu'entre en scène le Deep Learning. [24]

III.3. Machine Deep Learning (MDL)

III.3.1 Définition

Le concept de Machine Learning date du milieu du 20ème siècle. Dans les années 1950, le mathématicien britannique Alan Turing imagine une machine capable d'apprendre, une « Learning Machine ». Au cours des décennies suivantes, différentes techniques de Machine Learning ont été développées pour créer des algorithmes capables d'apprendre et de s'améliorer de manière autonome. [22]

III.3.2. Fonctionnement

Au sein du cerveau humain, chaque neurone reçoit environ 100 000 signaux électriques des autres neurones. Chaque neurone en activité peut produire un effet excitant ou inhibiteur sur ceux auxquels il est connecté. Au sein d'un réseau artificiel, le principe est similaire. Les signaux voyagent entre les neurones. Toutefois, au lieu d'un signal électrique, le réseau de neurones assigne un certain poids à différents neurones. Un neurone qui reçoit plus de charge exercera plus d'effet sur les neurones adjacents. La couche finale de neurones émet une réponse à ces signaux.

Pour comprendre comment fonctionne le Deep Learning, prenons un exemple concret de reconnaissance d'images. Imaginons que le réseau de neurones soit utilisé pour reconnaître les photos qui comportent au moins un chat. Pour pouvoir identifier les chats sur les photos, l'algorithme doit être en mesure de distinguer les différents types de chats, et de reconnaître un chat de manière précise quel que soit l'angle sous lequel il est photographié.

III.3.3. L'évolution des réseaux de neurones

Parmi les autres techniques populaires de Machine Learning, on compte l'« adaptative boosting » ou AdaBoost. Cette technique introduite en 2001 par Paul Viola et Michael Jones de Mitsubishi Electric Research Laboratories permet de détecter les visages en temps réel sur une image. Plutôt que de reposer sur un réseau de neurones interconnectés, AdaBoost filtre une image à partir d'un ensemble de décisions simples pour repérer les visages.

Cette technique et d'autres ont bien failli faire oublier les réseaux de neurones. Toutefois, grâce à l'explosion du nombre de données étiquetées, les réseaux de neurones sont revenus sur le devant de la scène. En 2007, une base de données regroupant des millions d'images étiquetées en provenance d'internet, ImageNet, a été lancée. Grâce à des services comme Amazon Mechanical Turk, proposant aux utilisateurs deux centimes pour chaque image étiquetée, la base de données a très rapidement été alimentée. Aujourd'hui, ImageNet regroupe 10 millions d'images étiquetées.

Les réseaux de neurones d'apprentissage profond ont également évolué et contiennent désormais bien plus de couches différentes. Le deep learning de Google Photos comporte par exemple 30 couches. Une autre évolution massive est celle des réseaux de neurones convolutifs. Ces réseaux ne s'inspirent pas seulement du fonctionnement du cerveau humain, mais aussi du système visuel.

Au sein d'un tel réseau, chaque épaisseur applique un filtre sur les images pour identifier des patterns ou des éléments spécifiques. Les premières épaisseurs détectent les principaux

attributs, tandis que les dernières épaisseurs repèrent les détails les plus subtils et les organisent en éléments concrets. Ainsi, ces réseaux convolutifs sont en mesure d'identifier des attributs hautement spécifiques, comme la forme des pupilles ou la distance entre le nez et les yeux, afin de reconnaître un chat avec une précision inouïe.

III.3.4. Champs d'application de Machine Deep Learning [23]

III.3.4.1. Sécurité informatique

Contrairement aux solutions basées sur le Machine Learning, les systèmes IT et de cyber-sécurité qui s'appuient sur le Deep Learning peuvent identifier aussi bien les dangers documentés que les risques jusqu'alors inconnus grâce à leur capacité à détecter les anomalies dans les patterns connus du réseau neuronal. Le Deep Learning démultiplie l'efficacité des mesures de sécurité.

III.3.4.2. Support client

Les chatbots basés sur le Deep Learning comprennent l'expression naturelle des personnes et ne sont pas limités à l'utilisation de mots-clés précis. Le dialogue est clairement plus efficace et les solutions proposées plus susceptibles de répondre à la demande.

III.3.4.3. Création de contenu

Le Deep Learning peut servir à automatiser la création de contenu. À partir d'une base de données de contenus suffisamment fournie, le système peut créer un nouveau contenu ou effectuer des traductions en autonomie.

III.3.4.4. Assistant vocal

Les assistants numériques, comme Siri, Alexa ou Google, reposent sur le Deep Learning. Les premiers assistants numériques commencent à faire leur apparition en entreprise. Les utilisateurs peuvent s'exprimer naturellement pour demander, par exemple, d'abandonner une commande, d'envoyer un email, de créer un rapport ou de lancer une recherche.

Outre les champs d'application cités, les deux technologies s'utilisent dans de nombreux domaines du quotidien, comme la médecine, les sciences ou la mobilité.

III.3.5. Limites de machine deep Learning

Les premières limitations tiennent à la grande quantité d'exemples nécessaires pour obtenir de très bonnes performances et au besoin d'étiqueter ces exemples. Or, l'étiquetage

requiert une intervention humaine très coûteuse, d'autant plus que les exemples doivent être massifs (plusieurs centaines de milliers, voire plusieurs millions d'instances).

Il existe aussi une limitation intrinsèque à l'apprentissage supervisé qui tient au langage de description des exemples : celui-ci demeure figé et ne peut s'accroître automatiquement. Or, ce que le philosophe américain et historien des sciences Thomas Kuhn (1922-1996) appelle des « révolutions scientifiques », ou ce que le Français Gaston Bachelard (1884-1962) décrit comme des ruptures épistémologiques, passe par l'introduction de nouvelles notions qui viennent bousculer les conceptions anciennes et changer les systèmes de représentation – comme l'ont fait par exemple, en leur temps, le passage du système géocentrique (Terre au centre de l'Univers et immobile) de Ptolémée au système héliocentrique de Copernic (Soleil au centre de l'Univers et immobile) ou celui de la représentation newtonienne du temps à celle d'Einstein et de la relativité. Pour reprendre les termes de Thomas Kuhn, si les techniques d'apprentissage profond permettent d'automatiser en partie la « science normale », elles sont incapables de procéder à des changements de paradigmes. [24]

III.4. Les différences entre les Machine Learning et les Machine Deep Learning

Le Machine Learning (apprentissage automatique) est la technologie la plus ancienne et la plus simple. Elle s'appuie sur un algorithme qui adapte lui-même le système à partir des retours faits par l'humain. La mise en place de cette technologie implique l'existence de données organisées. Le système est ensuite alimenté par des données structurées et catégorisées lui permettant de comprendre comment classer de nouvelles données similaires. En fonction de ce classement, le système exécute ensuite les actions programmées. Il sait par exemple identifier si une photo montre un chien ou un chat et classer le document dans le dossier correspondant.

Après une première phase d'utilisation, l'algorithme est optimisé à partir des feedbacks du développeur, qui informent le système des classifications erronées et lui indiquent les bonnes catégories.

Le Deep Learning (apprentissage profond) n'a pas besoin de données structurées. Le système fonctionne à partir de plusieurs couches de réseaux neuronaux, qui combinent différents algorithmes en s'inspirant du cerveau humain. Ainsi, le système est capable de travailler à partir de données non structurées.

Cette approche est particulièrement adaptée pour les tâches complexes, lorsque tous les aspects des objets à traiter ne peuvent pas être catégorisés en amont. Le système du Deep Learning identifie lui-même les caractéristiques discriminantes. Dans chaque couche, il recherche un nouveau critère spécifique de l'objet, qui sert de base pour décider de la classification retenue pour l'objet à la fin du processus.

Par ailleurs, la technologie nécessaire pour le Deep Learning est plus sophistiquée. Elle exige plus de ressources IT et s'avère nettement plus coûteuse que le Machine Learning : elle n'est donc pas intéressante, du moins à l'heure actuelle, pour une utilisation de masse par les entreprises. [23]

III.5. Synthèse des différences entre Machine Learning et Deep Learning

	Machine Learning	Machine Deep Learning
Organisation des données	Données structurées	Données non structurées
Base de données	Contrôlable	> 1 million de données
Entraînement	Entraînement par l'humain nécessaire	Système d'apprentissage autonome
Algorithme	Algorithme modifiable	Réseau neuronal d'algorithmes
Champ d'application	Actions simples de routine	Tâches complexes

Tableau III.1 : Différences entre Machine Learning et Deep Learning

III.6. Les réseaux de neurones

III.6.1. Introduction

Depuis une dizaine d'années, l'utilisation des réseaux de neurones artificiels (RNA) s'est développée dans de nombreuses disciplines (sciences économiques, écologie et environnement, biologie et médecine...). Ils sont notamment appliqués pour résoudre des problèmes de classification, de prédiction, de catégorisation, d'optimisation, de reconnaissance des formes et de mémoire associative (Drew et Monson, 2000). Dans le cadre du traitement des

données, les RNA constituent une méthode d'approximation de systèmes complexes, particulièrement utile lorsque ces systèmes sont difficiles à modéliser à l'aide des méthodes statistiques classiques. Les RNA sont également applicables dans toutes les situations où il existe une relation non linéaire entre une variable prédictive et une variable prédite. Par leur nature et leur fonctionnement, les RNA peuvent détecter les interactions multiples non linéaires parmi une série de variables d'entrée, ils peuvent donc gérer des relations complexes entre les variables indépendantes et les variables dépendantes

III.6.2. Historique

L'Intelligence Artificielle, branche de l'Informatique fondamentale s'est développée avec pour objectif la simulation des comportements du cerveau humain. Les premières tentatives de modélisation du cerveau sont anciennes et précèdent même l'ère informatique. C'est en 1943 que Mc Culloch (neurophysiologiste) et Pitts (logicien) ont proposé les premières notions de neurone formel. Ce concept fut ensuite mis en réseau avec une couche d'entrée et une sortie par Rosenblatt en 1959 pour simuler le fonctionnement rétinien et tacher de reconnaître des formes. C'est l'origine du perceptron. Cette approche dite connexionniste a atteint ses limites technologiques, compte tenu de la puissance de calcul de l'époque, mais aussi théoriques au début des années 70. L'approche connexionniste à connaissance répartie a alors été supplantée par une approche symbolique qui promouvait les systèmes experts à connaissance localisée dont L'objectif était d'automatiser le principe de l'expertise humaine en associant trois concepts : • une base de connaissance dans laquelle sont regroupées les connaissances d'experts humains sous forme de propositions logiques élémentaires ou plus élaborées en utilisant des quantificateurs (logique du premier ordre). une base de faits contenant les observations du cas à traiter comme, par exemple, des résultats d'examens, d'analyses de sang, de salive pour des applications biomédicales de choix d'un antibiotique, un moteur d'inférence chargé d'appliquer les règles expertes sur la base de faits afin d'en déduire de nouveaux faits jusqu'à la réalisation d'un objectif comme le choix du traitement d'une infection bactérienne. Face aux difficultés rencontrées lors de la modélisation des connaissances d'un expert humain, au volume considérable des bases qui en découlaient et au caractère exponentiel de la complexité des algorithmes d'inférence mis en jeu, cette approche s'est éteinte avec les années 80. Il a été montré que les systèmes basés sur le calcul des prédicats du premier ordre conduisaient à des problèmes NP complets. L'essor technologique et quelques avancées théoriques : estimation du gradient par rétro-propagation de l'erreur (Hopkins, 1982) analogie de la phase d'apprentissage avec les modèles markoviens de systèmes de particules de la

mécanique statistique (verres de spin) par (Hopfield, 1982), au début des années 80 ont permis de relancer l'approche connexionniste. Celle-ci a connu au début des années 90 un développement considérable si l'on considère le nombre de publications et de congrès qui lui ont été consacrés mais aussi les domaines d'applications très divers où elle apparaît. La motivation initiale de simulation du cortex cérébral a été rapidement abandonnée alors que les méthodes qui en découlaient ont trouvé leur propre intérêt de développement méthodologique et leurs champs d'applications. Remis en vogue depuis le milieu des années 90 au profit d'autres algorithmes d'apprentissage machine ou plutôt statistique : boosting, support vector machine..., les réseaux de neurones connaissent un regain d'intérêt et même un énorme battage médiatique sous l'appellation d'apprentissage profond (deep Learning). La taille des bases de données, notamment celles d'images issues d'internet, associée à la puissance de calcul disponible, permettent d'estimer les millions de paramètres du perceptron accumulant des dizaines voire centaines de couches de neurones aux propriétés très spécifiques. Ce succès médiatique est la conséquence des résultats spectaculaires obtenus par ces réseaux en reconnaissance d'image, jeux de go, traitement du langage naturel... [13]

III.6.3. Le neurone biologique

Les cellules nerveuses, appelées neurones, sont les éléments de base du système nerveux central ; celui-ci en posséderait environ cent milliards. Par ailleurs, il existe de nombreuses variétés de neurones, les différences portant tant sur les aspects anatomiques que sur les aspects fonctionnels. [14]

En général, chaque neurone biologique possède un axone qui peut atteindre plusieurs dizaines de centimètres, un corps cellulaire où se situe le mécanisme du déclenchement du neurone, des dendrites qui reçoivent des informations des autres neurones et de terminaisons axiales qui émettent des informations vers les autres neurones (figure III.1). Les neurones sont interconnectés entre eux par des synapses dont le nombre est environ 10000 milliards. Ceci signifie qu'un neurone reçoit en moyenne des excitations en provenance de 1000 neurones, et qui émet également des informations à 1000 neurones différents.

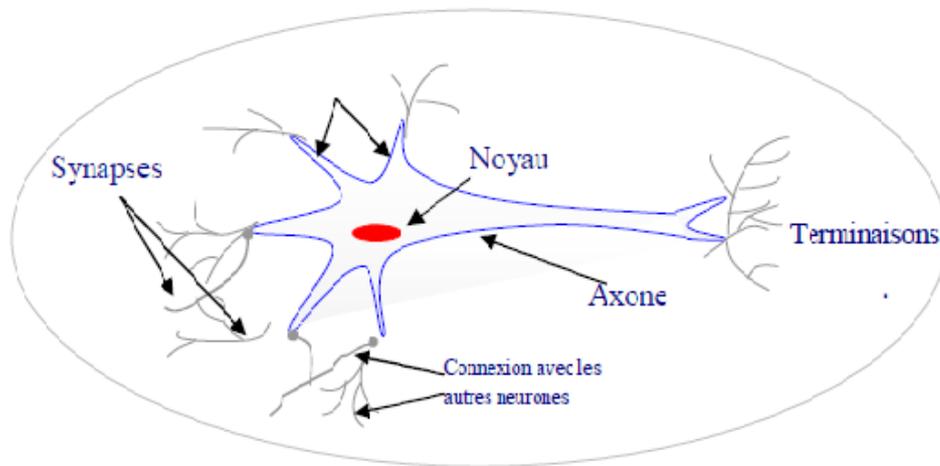


Figure III.1 : schéma simplifié d'un neurone du cerveau humain [15]

Il existe quatre modes de communications entre neurones :

- liaison terminaison axiale-dendritique.
- liaison terminaison axiale-corps cellulaire.
- liaison axone-axone.
- liaison dendrite-dendrite.

Le processus de communication entre les neurones se fait par voie électrochimique aux niveaux des synapses par des neurotransmetteurs ; ces fonctions spécifiques dépendent essentiellement des propriétés de la membrane externe du neurone, cette dernière remplit cinq fonctions principales :

- elle sert à propager des impulsions électriques tout au long de l'axone et des dendrites.
- elle libère des médiateurs à l'extérieur de l'axone.
- elle réagit au niveau du corps cellulaire aux impulsions électriques que lui transmettent les dendrites pour générer ou non une nouvelle impulsion.
- enfin, elle permet au neurone de reconnaître les autres neurones afin qu'il puisse se situer dans l'architecture anatomique du cerveau et de trouver les cellules auxquelles il doit être connecté.

III.6.4. Le neurone artificiel

L'élément de base d'un réseau de neurones est, bien entendu, le neurone artificiel [16] présenté par la figure. Un neurone artificiel contient deux éléments principaux :

- un ensemble de poids associés aux connexions du neurone.

- une fonction d'activation (Tableau 1).

Les valeurs d'entrée sont multipliées par leur poids correspondant et est additionnées pour obtenir la somme S .

Cette somme devient l'argument de la fonction d'activation, qui est le plus souvent d'une des formes présentées ci-dessous. Une fonction d'activation importante est la simple multiplication avec la valeur 1, c'est à dire que la sortie est simplement une somme pondérée.

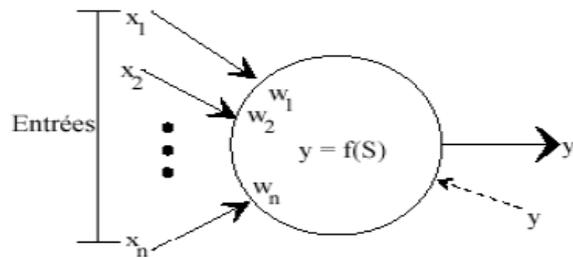


Figure III.2 : Le neurone artificiel générique

Généralement, les réseaux de neurones artificiels sont basés sur une théorie électrique selon laquelle la jonction synaptique, est l'endroit où le neurone perçoit un stimulus par voie électrochimique. La membrane synaptique, lorsqu'elle ne reçoit pas d'excitation, est polarisée à une tension supérieure à un seuil donné, le neurone étant alors activé et cela déclenche un processus de dépolarisation ; le neurone émet sur l'axone une tension positive d'environ (+60 à +70mv) en se comportant comme un système non linéaire. C'est ce type de modèle qui est majoritairement retenu pour simuler artificiellement les neurones du cerveau humain.

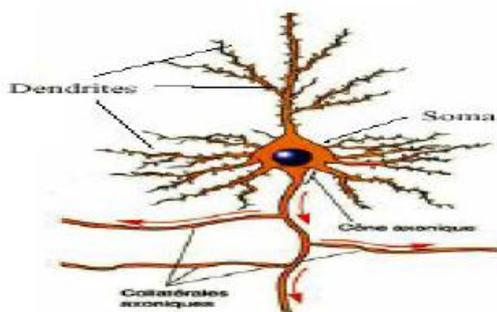


Figure III.3 : Un neurone biologique [17]

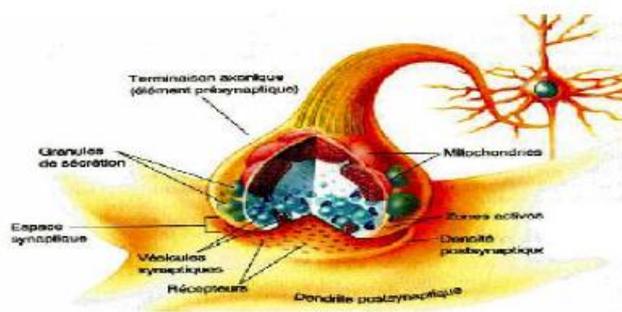


Figure III.4 : Une synapse [17]

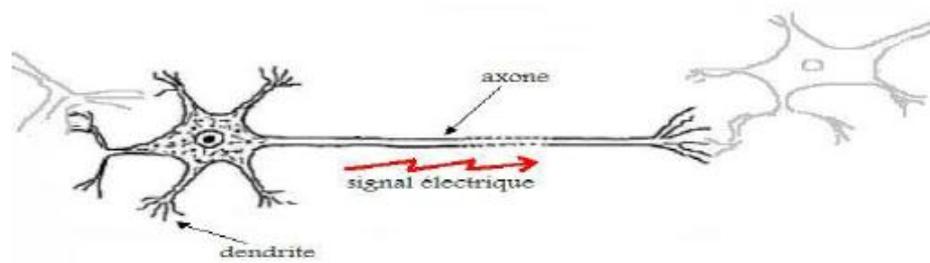


Figure III.5 : impulsions électrique entre les neurones [17]

III.6.5. Fonction d'activation

La fonction d'activation est en générale une fonction non linéaire monotone croissante par ailleurs les fonctions de transfert sont de qualités diverses : elles peuvent être déterministes, continues, discontinues ou aléatoires [18] [19].

Ces fonctions sont présentées dans la figure III.6

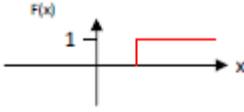
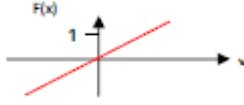
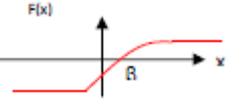
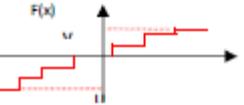
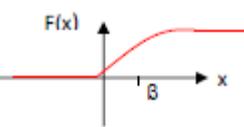
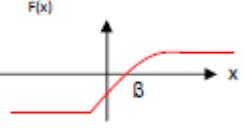
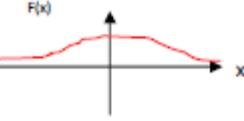
(a) Fonction de Heaviside	
(b) Fonction linéaire sans saturation	
(c) Fonction linéaire avec seuil	
(d) fonction à seuils multiples	
(e) fonction sigmoïde $F(x) = \frac{1}{1 + e^{-x}}$	
(f) fonction sigmoïde bipolaire $F(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$	
(g) fonction stochastique	

Figure III.6 : fonction d'activation

III.7. Les réseaux de neurones artificiels

Les recherches actuelles faites sur les réseaux de neurones artificiels n'ont pas permis de donner une définition exacte ou universelle à ce concept.

En effet, la définition admise, utilise la notion de réseaux d'automates ou de système connexionniste [15]. Un réseau de neurones artificiels est composé d'automates connectés en réseau et fonctionnant en parallèle et dans lequel les connexions contiennent la connaissance d'un domaine particulier.

Par ailleurs, il existe une grande variété des réseaux de neurones artificiels, ceux-ci pouvant être définis selon plusieurs critères, parmi lesquels, on peut citer :

- différentes architectures de réseaux de neurones ;
- mécanisme d'apprentissage ;
- apprentissage des réseaux multicouches.

III.7.1. Différentes architectures de réseaux de neurones

Selon la topologie de connexion des neurones, on peut classifier plusieurs modèles de réseaux de neurones.

III.7.1.2. Réseau multicouche (*feed-forward*)

Dans les réseaux multicouche (*feed-forward*) les neurones sont arrangés par couche, il n'y a pas de connexion entre neurone d'une même couche et la connexion ne se fait qu'avec les neurones des couches en aval. Habituellement, chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et celle-ci seulement, nous permet d'introduire la notion de sens de parcours de l'information (de l'activation) au sein d'un réseau et de définir les concepts de neurone d'entrée et de neurone de sortie. Par extension on appelle couche d'entrée l'ensemble des neurones d'entrées et la couche de sortie l'ensemble des neurones de sorties. Les couches intermédiaires n'ayant aucun contact avec l'extérieur sont appelées couches cachées, tel que représenté sur la figure III.7

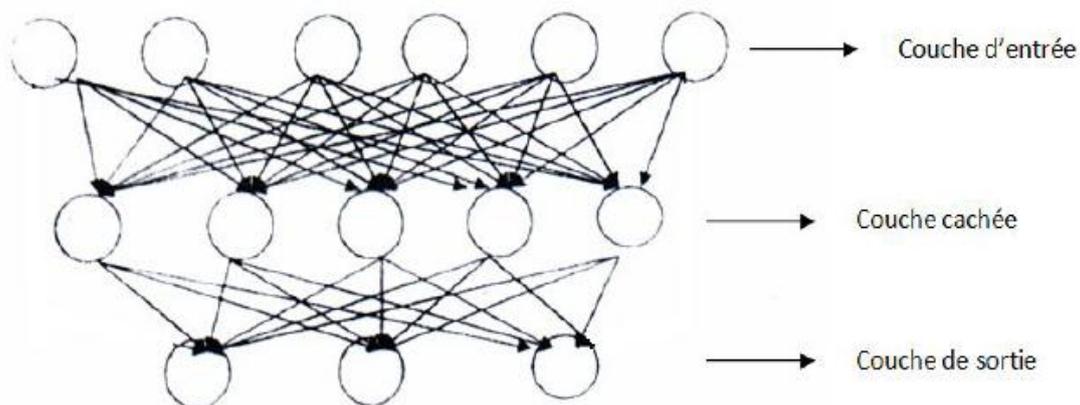


Figure III.7 : Réseau multicouche

III.7.1.2. Réseau à connexion locale

Il s'agit d'une structure multicouche, mais qui a l'image de la rétine, conserve une certaine topologie. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurone de couche avale. Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique. Tel que le montre la figure III.8

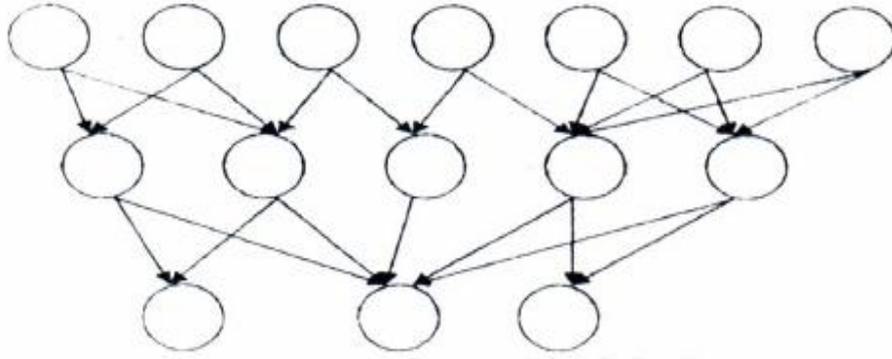


Figure III.8 : Structure multicouche à réseau de connexion locale.

III.7.1.3. Réseau à connexion récurrente

Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales. Ceci est représenté par la figure 3.8

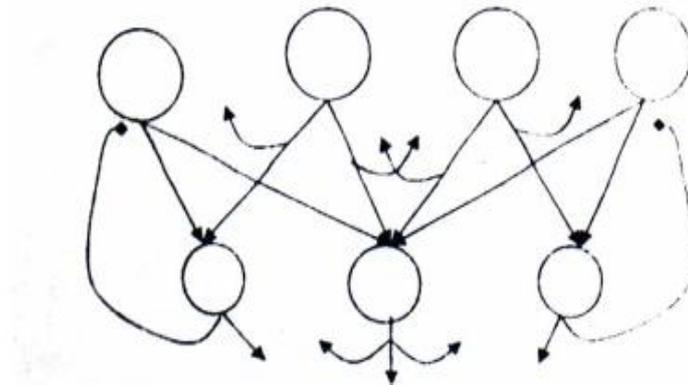


Figure III.9 : Réseau à connexion récurrente.

III.7.1.4. Réseau à connexion complète

C'est la structure d'interconnexion la plus générale. Chaque neurone est connecté à tous les neurones du réseau et à lui-même. Tel que le montre la figure 3.9

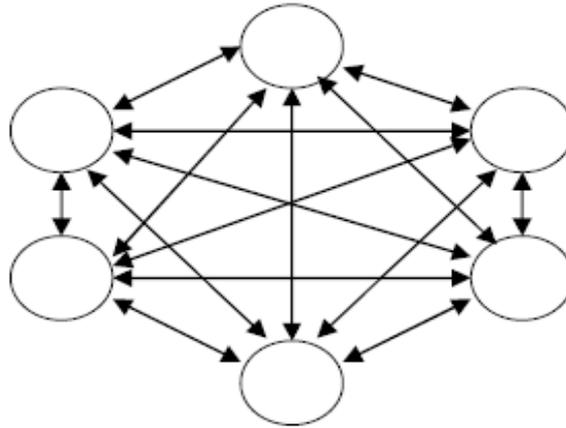


Figure III.10 : Réseau de connexion complète.

III.7.1.5. Les couches de neurones

- **LSTM** : Un réseau de long court terme (en anglais long short-term memory, LSTM) est un type de modèle RNN (recurrent neural network) qui empêche le phénomène de vanishing gradient en ajoutant des portes d'oubli
- **Dropout** : le dropout est une technique qui est destinée à empêcher le surajustement sur les données de training en abandonnant des unités dans un réseau de neurones. En pratique, les neurones sont soit abandonnés avec une probabilité p ou gradés avec une probabilité $1-p$. Cette méthode consiste à « désactivé » des sorties de neurones aléatoirement (avec une probabilité prédéfinie, par exemple 0.5 pour les couches cachées et 0.8 pour la couche d'entrée) pendant la phase d'apprentissage. Le dropout peut permettre une accélération de l'apprentissage
- **Dense** : c'est la plus simple des couches possibles : chaque neurone est simplement connecté à tous les neurones de la couche précédente.

III.7.2. Apprentissage

L'apprentissage [17] est vraisemblablement la caractéristique la plus intéressante des réseaux de neurones. Il ne concerne cependant pas tous les modèles, mais les plus utilisés.

III.7.2.1. définition

L'apprentissage est une phase du développement d'un réseau de neurone durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré, l'apprentissage neuronal fait appel à des exemples de comportement. Dans le cas des réseaux

de neurones artificiels on ajoute souvent à la description un modèle de l'algorithme d'apprentissage. Le modèle sans apprentissage présente en effet peu d'intérêt dans la majorité des algorithmes actuels. Les variables modifiées pendant l'apprentissage sont les poids de connexion.

III.7.2.2. Apprentissage supervisé

Dans ce type d'apprentissage, l'algorithme détermine les poids synaptiques à partir d'exemples étiquetés de formes auquel un professeur (teacher) a associé des réponses ou des cibles également étiquetées. Il existe plusieurs algorithmes, parmi lesquels on distingue l'algorithme de rétro-propagation qui est destiné aux réseaux multicouches.

III.7.2.3. Apprentissage non supervisé

Dans ce type d'algorithmes, on présente à l'entrée du réseau des exemples connus et le réseau s'organise lui-même autour d'attracteurs qui correspondent à des configurations stables du modèle dynamique non linéaire associé au réseau. L'apprentissage est accompli à l'aide de règles qui changent ou adaptent le poids des coefficients synaptiques en fonction des exemples présentés à l'entrée et dans certains cas en fonction des sorties désirées.

L'apprentissage a lieu souvent en temps réel avec des réseaux qui peuvent être éventuellement bouclés, parmi lesquels, on peut citer : le réseau de HOPFIELD, et les cartes topologiques de KOHONEN.

III.7.2.4. Choix du taux d'apprentissage

Plusieurs solutions ont été proposées pour le choix de taux d'apprentissage ou (le pas d'apprentissage), d'après Freeman on choisit un pas d'apprentissage entre (-0.5 ; 0.25) mais le problème qui se pose à chaque fois c'est le rapport vitesse / complexité qui n'est pas toujours convaincant. L'une des solutions est celle de Brunel qui consiste à adapter le pas afin d'avoir une décroissance régulière de l'erreur quadratique moyenne.

III.7.2.5. Valeurs initiales des poids

Les valeurs des poids doivent être différentes de zéro. Si c'était le cas les inconvénients d'adaptation des poids sont nuls. En pratique on choisit aléatoirement les valeurs initiales des poids.

III.7.2.6. Test d'arrêt

En pratique il faut calculer un indice de convergence car on ne peut pas assurer la convergence de l'algorithme pour un nombre d'itérations alors qu'un autre cherche à arrêter l'algorithme si l'erreur est minimale.

III.7.2.7. Choix de réseau

De nombreux travaux théoriques abordent l'approximation des fonctions continues par les réseaux de neurones ce qui a amené à l'application du choix de ne pas pouvoir savoir la taille exacte du réseau pour un problème donné.

III.7.2.8. Apprentissage des réseaux multicouches

Les pondérations des différents réseaux utilisées sont ajustées par l'apprentissage supervisé en exploitant l'idée de minimisation itérative d'un critère de l'erreur en sortie [20] qui a pour principe de minimiser une « fonction de coût » de l'erreur calculée entre la sortie du réseau et la sortie souhaitée pour une entrée donnée. Donc le rôle de ce principe est de trouver le plus rapidement possible le minimum d'une fonction d'énergie dont la forme dépend uniquement de l'architecture du réseau.

Toutefois, les principes sous-jacents aux règles d'apprentissage les plus connus ne sont rien d'autre que des principes d'optimisation. Le plus utilisé est le principe de descente de gradient, il a été conçu pour dévier les problèmes de non séparabilité linéaire. Tel que le montre la figure 3.10

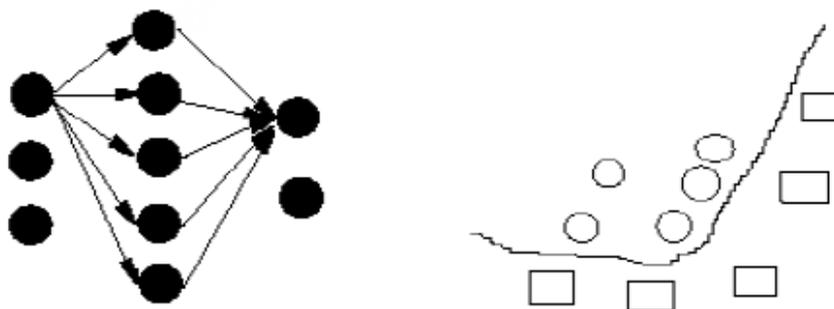


Figure III.11 : La séparation non linéaire (rétro-propagation : cas général).

III.8. Conclusion

L'objectif pédagogique visé dans ce survol du monde biologique est la mise en exergue d'une organisation structurelle des neurones. Chaque structure est dotée d'une fonction particulière et ces structures adaptent leur comportement par des mécanismes d'apprentissage. L'apprentissage implique des modifications physiques des connexions entre neurones. L'association entre plusieurs structures neuronales, dotées chacune d'une fonction précise, permet l'émergence d'une fonction d'ordre supérieure pour l'ensemble.

Chapitre IV : conception et implémentation

IV.1. Introduction

Dans ce chapitre nous allons présenter une problématique de maintenance assistée par ordinateur : maintenance prédictive des pannes d'un turboréacteur et on va expliquer comment la propagation des dommages peut être modélisée dans le turboréacteur des avions.

Ensuite, nous allons simuler une détection de défauts après une période étudiée en utilisant des techniques d'IA sous un environnement de développement basée sur python.

Nous allons discuter les outils, plateformes et interfaces nécessaires au développement de notre approche. Nous discuterons ensuite, quelques résultats obtenus et nous conclurons le chapitre.

IV.2. Problématique (Maintenance assistée par ordinateur : maintenance prédictive des pannes d'un turboréacteur)

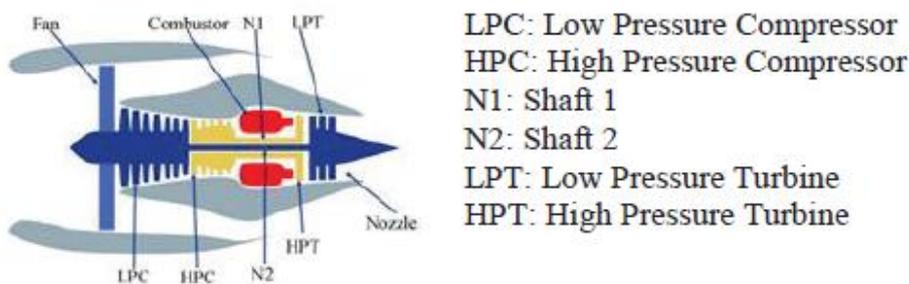


Figure IV.1 Schéma simplifié d'un moteur à turboréacteur [28]

Le turboréacteur est un système de propulsion qui transforme le potentiel d'énergie chimique contenu dans un carburant, associé à un comburant qui est l'air ambiant, en énergie cinétique permettant de générer une force de réaction en milieu élastique dans le sens opposé à l'éjection.

Ce type de moteur est essentiellement utilisé sur les avions de type commercial ou militaire. La poussée générée résulte de l'accélération d'une certaine quantité d'air entre l'entrée (buse d'entrée d'air) et la sortie (tuyère d'éjection). Afin d'injecter une quantité d'air suffisante en masse, un accroissement de la pression à vitesse à peu près constante est assuré par le compresseur d'entrée. Un important dégagement d'énergie est ensuite provoqué par la combustion d'un carburant, généralement du kérosène, dans l'oxygène de l'air qui traverse la machine. Une partie de l'énergie produite est récupérée par une turbine à la sortie de la chambre de combustion pour entraîner certains accessoires, dont le compresseur situé juste en aval de

l'entrée d'air. L'autre partie du flux chaud (additionnée ou non au flux froid suivant le type de réacteur) produit la poussée par détente dans la tuyère d'éjection.

IV.3. Conception

Dans cette section nous allons présenter :

IV.3. 1.Environment & outils de développement

IV.3.1.1.Environment matériel

Avant de commencer l'implémentation de l'application, il y a lieu d'abord de spécifier les outils utilisés, qu'on a jugés bon de parte les avantages qu'ils offrent

IV.3.1.1.1 Ressources matérielles

Nous avons développé notre application sur une machine Intel ® CORE ™ i5-7200U CPU @M380, 2.53 2,70 GHz, doté d'une capacité mémoire de 8.00 GO de RAM.

IV.3.1.1.2. Ressources logicielles

Le système d'exploitation qu'on a utilisé est Micro Microsoft Windows 10 Pro N, 64-bit operating system, x64-based processor dans lequel est installé l'interpréteur python 3.7.6

IV.3. 1.2.Outil de développement

Nous avons choisi l'application de langage " python " et Jupyter pour afficher le résultat après chaque exécution.

IV.3.1.2.1. Définition python :

Coelho et Richert [29] présentent Machine Learning et Python comme une équipe de rêve. Ils exposent l'approche du ML comme un processus itératif, et que c'est exactement ce qui fait de Python. Si est un bon langage pour ML. Python est un langage open-source largement utilisé dans l'industrie ou pour des fins académiques .Python dispose de plusieurs bibliothèques utiles pour faciliter les opérations, telles que NumPy, Pandas,Sklearn et SciPy. Il dispose également de plusieurs cadres d'apprentissage approfondi qui fonctionnent en plus de Python comme Tensorflow, Keras, PaddlePaddle. [30]

IV.3.1.2.2. Définition Jupyter

Jupyter Notebook App est un logiciel qui permet d'éditer et d'exécuter des documents en python via un navigateur web [31]. Sur leur page web, un cours décrit Jupyter comme un outil où l'on peut travailler interactivement avec du code. . Grâce à la structure cellulaire de Jupyter, il permet de faire fonctionner une cellule à la fois, ce qui permet d'effectuer rapidement des tests et de développer des modèles.

IV.3. 2. Modèle d'apprentissage

IV.3. 2.1.Préparation des données

IV.3.2.1.1. Base de données sur maintenance prédictive des pannes d'un turboréacteur

La base de données de la NASA contient des données sur la dégradation des moteurs qui ont été simulées à l'aide du C-MAPSS (Commercial Modular Aero-Propulsion System Simulation). Quatre ensembles différents ont été simulés dans différentes combinaisons de conditions opérationnelles et de modes de défaillance. Chaque ensemble comprend des paramètres opérationnels et des mesures de capteurs (température, pression, vitesse de la soufflante, etc.) pour plusieurs moteurs et pour chaque cycle de leur vie. Pour plus d'informations sur la base de données.

Le turboréacteur est un moteur moderne à turbine à gaz utilisé par l'agence d'exploration spatiale de la NASA.

La NASA a créé la base de données suivante pour prévoir les défaillances des turboréacteurs à long terme. La base de données est disponible sur le site internet [28]

IV.3.2.1.2. Techniques d'apprentissage

Pour faire de la maintenance prédictive, nous ajoutons d'abord au système des capteurs qui vont surveiller et collecter des données sur ses opérations. Les données pour la maintenance prédictive sont des données de séries chronologiques. Les données comprennent un horodatage, un ensemble de relevés de capteurs collectés en même temps que les horodatages, et des identificateurs de dispositifs. L'objectif de la maintenance prédictive est de prévoir au moment "temps", en utilisant les données jusqu'à ce moment, si l'équipement tombera en panne dans un avenir proche.

La maintenance prédictive peut être formulée de l'une des deux façons suivantes :

- Approche par classification : Prédit s'il y a une possibilité de défaillance dans les n étapes suivantes.
- Approche de régression : Prédit le temps qui reste avant la prochaine défaillance. C'est ce que nous appelons la durée de vie utile restante

La première approche ne fournit qu'une réponse booléenne, mais peut fournir une plus grande précision avec moins de données. La seconde a besoin de plus de données, bien qu'elle fournisse plus d'informations sur le moment où l'échec se produira. Nous allons explorer ces deux approches en utilisant la base de données de la NASA sur les pannes de moteur.

IV.3.2.1.3. PAQUETS ET BIBLIOTHÈQUES

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix, accuracy_score

from keras.models import Sequential
from keras.layers import Dense, Dropout, LSTM, Activation
from keras.callbacks import EarlyStopping

import matplotlib.pyplot as plt
plt.style.use('ggplot')
%matplotlib inline
```

FigureIV.2 : Les bibliothèques requises

- **Les bibliothèques**

En Python, **NumPy** et **SciPy** permettent de décharger les tâches d'écrasement de nombres vers la couche inférieure du forme d'extension C. NumPy prend en charge des extensions hautement optimisées des tableaux multidimensionnels. SciPy utilise ces tableaux pour fournir un ensemble de recettes numériques rapides. Dans leur livre, Coelho et Richert [29], présentent **matplotlib** comme le package le plus pratique et le plus riche en fonctionnalités pour tracer des graphiques de haute qualité en Python. **Pandas** [32] fournit des structures de données et des outils d'analyse de données performants et faciles à utiliser pour le langage de programmation Python [33]. Pandas offrent un traitement de visualisation matricielle. **Sklearn** est le nom python de **Scikit-learn**. **Scikitlearn** est construit sur NumPy,

SciPy et matplotlib. Il est conçu comme un outil simple et efficace pour le traitement des données, notamment le prétraitement et l'analyse des données [34].

- **Framework d'apprentissage**

- **Keras** : Keras est une API de réseaux neuronaux de haut niveau, écrite en Python et capable de fonctionner sur TensorFlow. Elle a été développée dans le but de permettre une expérimentation rapide avec un prototypage facile et rapide. Pouvoir passer de l'idée au résultat dans les plus brefs délais. [35]
- **TensorFlow** : TensorFlow est une bibliothèque de logiciels libres pour le calcul numérique de haute performance. Son architecture flexible permet le déploiement de calculs sur une variété de plateformes (CPU, GPU, TPU), et des ordinateurs de bureau aux grappes de serveurs en passant par les appareils mobiles et périphériques. [35]

IV.3.2.1.4. Importer la base de données dans l'application de langage python

```
dataset_train=pd.read_csv('data/train_FD001.txt', sep=' ', header=None).drop([26,27], axis=1)
col_names = ['id', 'cycle', 'setting1', 'setting2', 'setting3', 's1', 's2', 's3', 's4', 's5', 's6', 's7', 's8', 's9', 's10', 's11', 's12', 's13', 's14', 's15', 's16', 's17', 's18', 's19', 's20', 's21']
dataset_train.columns=col_names
```

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s12	s13	s14	s15	s16
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	521.66	2388.02	8138.62	8.4195	0.03
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	522.28	2388.07	8131.49	8.4318	0.03
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	522.42	2388.03	8133.23	8.4178	0.03
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	522.86	2388.08	8133.83	8.3682	0.03
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	522.19	2388.04	8133.80	8.4294	0.03

5 rows × 26 columns

Figure IV.3 : importer le fichier 'train_FD001'

Nous commençons par importer la base de données. Nous importons le fichier train_FD001 et ajouterons dans chaque en-tête de colonne une étiquette afin de classer chaque colonne par nom et par ordre.

```
dataset_test=pd.read_csv('data/test_FD001.txt', sep=' ', header=None).drop([26,27], axis=1)
dataset_test.columns=col_names
```

Figure IV.4 : importer le fichier 'test_FD001'

In [4]:

```
pm_truth=pd.read_csv('data/RUL_FD001.txt',sep=' ',header=None).drop([1],axis=1)
pm_truth.columns=['more']
pm_truth['id']=pm_truth.index+1
pm_truth.head()
```

Out[4]:

	more	id
0	112	1
1	98	2
2	69	3
3	82	4
4	91	5

Figure IV.5 : importer le fichier 'RUL_FD001'

Nous importons le fichier 'RUL_FD001' et Nous ajoutons le nom 'more' à l'en-tête de la colonne et Ajoutez une nouvelle colonne et nommez-la 'id' dans l'en-tête de colonne et la calculez avec 'index + 1'

In [6]:

```
# run to failure
pm_truth['rtf']=pm_truth['more']+rul['max']
pm_truth.head()
```

Out[6]:

	more	id	rtf
0	112	1	143
1	98	2	147
2	69	3	195
3	82	4	188
4	91	5	189

Figure IV.6 : ajouter une nouvelle colonne 'rtf'(run to failure)

Nous ajoutons une nouvelle colonne ' rtf ' (run to failure) et calculez la avec colonne ' more ' + colonne ' max'

In [7]:

```
pm_truth.drop('more', axis=1, inplace=True)
dataset_test=dataset_test.merge(pm_truth,on=['id'],how='left')
dataset_test['ttf']=dataset_test['rtf'] - dataset_test['cycle']
# dataset_test.drop('rtf', axis=1, inplace=True)
dataset_test.head()
```

```
[7]:
```

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s14	s15	s16	s17	s18	s19	s20	s21	rtf	ttf
0	1	1	0.0023	0.0003	100.0	518.67	643.02	1585.29	1398.21	14.62	...	8125.55	8.4052	0.03	392	2388	100.0	38.86	23.3735	143	142
1	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916	143	141
2	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166	143	140
3	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737	143	139
4	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130	143	138

5 rows × 28 columns

Figure IV.7 : ajouter ttf et rtf au tableau des données de test

Nous ajoutons ttf et rtf au tableau des données de test et calculez la ('Rtf' - 'cycle')

In [9]:

```
dataset_train['ttf'] = dataset_train.groupby(['id'])['cycle'].transform(max)-dataset_train['cycle']
dataset_train.head()
```

```
[9]:
```

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s13	s14	s15	s16	s17	s18	s19	s20	s21	ttf
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	2388.02	8138.62	8.4195	0.03	392	2388	100.0	39.06	23.4190	191
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236	190
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442	189
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739	188
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044	187

5 rows × 27 columns

Figure IV.8 : ajouter une nouvelle colonne ' ttf' dans le tableau des données de train

Nous ajoutons une nouvelle colonne ' ttf' dans le tableau des données de train et calculez-la Avec colonne ' max ' - colonne 'cycle '

In [10]:

```
df_train=dataset_train.copy()
df_test=dataset_test.copy()
period=30
df_train['label_bc'] = df_train['ttf'].apply(lambda x: 1 if x <= period else 0)
df_test['label_bc'] = df_test['ttf'].apply(lambda x: 1 if x <= period else 0)
df_train.head()
```

```
[10]:
```

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s14	s15	s16	s17	s18	s19	s20	s21	ttf	label_bc
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	8138.62	8.4195	0.03	392	2388	100.0	39.06	23.4190	191	0
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236	190	0
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442	189	0
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739	188	0
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044	187	0

5 rows × 28 columns

Figure IV.9 : Créez deux nouvelles tableaux et nommez-les df_train et df_test

Créez deux nouvelles tableaux et nommez-les `df_train` et `df_test` et copiez-y les données de tableau des données de train et tableau des données de test par ordre

IV.3. 2.2.Création du modèle

Un modèle séquentiel est approprié pour une simple pile de couches où chaque couche a exactement un tenseur d'entrée et un tenseur de sortie. Il est constitué de cinq de couches

les couches utilisé dans ce modele est :

- LSTM (*Long Short-Term Memory*).
- Dropout.
- Dense.

```
In [12]: sc=MinMaxScaler()
df_train[features_col_name]=sc.fit_transform(df_train[features_col_name])
df_test[features_col_name]=sc.transform(df_test[features_col_name])

In [13]: # function to generate sequences
def gen_sequence(id_df, seq_length, seq_cols):
    df_zeros=pd.DataFrame(np.zeros((seq_length-1,id_df.shape[1])),columns=id_df.columns)
    id_df=df_zeros.append(id_df,ignore_index=True)
    data_array = id_df[seq_cols].values
    num_elements = data_array.shape[0]
    lstm_array=[]
    for start, stop in zip(range(0, num_elements-seq_length), range(seq_length, num_elements
)):
        lstm_array.append(data_array[start:stop, :])
    return np.array(lstm_array)

# function to generate labels
def gen_label(id_df, seq_length, seq_cols,label):
    df_zeros=pd.DataFrame(np.zeros((seq_length-1,id_df.shape[1])),columns=id_df.columns)
    id_df=df_zeros.append(id_df,ignore_index=True)
    data_array = id_df[seq_cols].values
    num_elements = data_array.shape[0]
    y_label=[]
    for start, stop in zip(range(0, num_elements-seq_length), range(seq_length, num_elements
)):

In [14]: # timestamp or window size
seq_length=50
seq_cols=features_col_name

In [15]: # generate X_train
X_train=np.concatenate(list(list(gen_sequence(df_train[df_train['id']==id], seq_length, seq
cols)) for id in df_train['id'].unique()))
print(X_train.shape)
# generate y_train
y_train=np.concatenate(list(list(gen_label(df_train[df_train['id']==id], 50, seq_cols,'label
_bc')) for id in df_train['id'].unique()))
print(y_train.shape)

(20531, 50, 24)
(20531,)
```

Figure IV.10 : générer x-train et y-train

```
In [16]:
# generate X_test
X_test=np.concatenate(list(list(gen_sequence(df_test[df_test['id']==id], seq_length, seq_cols) for id in df_test['id'].unique()))
print(X_test.shape)
# generate y_test

y_test=np.concatenate(list(list(gen_label(df_test[df_test['id']==id], 50, seq_cols, 'label_bc') for id in df_test['id'].unique()))
print(y_test.shape)

(12996, 50, 24)
(12996,)
```

Figure IV.11 : générer x-test et y-test

```
In [17]:
nb_features =X_train.shape[2]
timestamp=seq_length

model = Sequential()

model.add(LSTM(
    input_shape=(timestamp, nb_features),
    units=100,
    return_sequences=True))
model.add(Dropout(0.2))

model.add(LSTM(
    units=50,
    return_sequences=False))
model.add(Dropout(0.2))

model.add(Dense(units=1, activation='sigmoid'))
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

model.summary()

Model: "sequential"

```

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 50, 100)	50000
dropout (Dropout)	(None, 50, 100)	0
lstm_1 (LSTM)	(None, 50)	30200
dropout_1 (Dropout)	(None, 50)	0
dense (Dense)	(None, 1)	51

```

Total params: 80,251
Trainable params: 80,251
Non-trainable params: 0

```

Figure IV.12 : Création de model avec plusieurs couches

In [18]:

```
# fit the network
model.fit(X_train, y_train, epochs=10, batch_size=200, validation_split=0.05, verbose=1,
         callbacks = [EarlyStopping(monitor='val_loss', min_delta=0, patience=0, verbose=0, mode='auto')])

Epoch 1/10
98/98 [=====] - 18s 180ms/step - loss: 0.1983 - accuracy: 0.9242
- val_loss: 0.0907 - val_accuracy: 0.9562
Epoch 2/10
98/98 [=====] - 16s 163ms/step - loss: 0.0779 - accuracy: 0.9688
- val_loss: 0.0657 - val_accuracy: 0.9757
Epoch 3/10
98/98 [=====] - 16s 162ms/step - loss: 0.0720 - accuracy: 0.9712
- val_loss: 0.0601 - val_accuracy: 0.9766
Epoch 4/10
98/98 [=====] - 16s 161ms/step - loss: 0.0652 - accuracy: 0.9719
- val_loss: 0.0545 - val_accuracy: 0.9766
Epoch 5/10
98/98 [=====] - 16s 163ms/step - loss: 0.0618 - accuracy: 0.9737
- val_loss: 0.0795 - val_accuracy: 0.9630
```

Figure IV.13 : donner x-train et y-train et démarrage l'entraînement de modèle

IV.3.2.3. Test du modèle

In [19]:

```
# training metrics
scores = model.evaluate(X_train, y_train, verbose=1, batch_size=200)
print('Accuracy: {}'.format(scores[1]))

103/103 [=====] - 6s 62ms/step - loss: 0.0759 - accuracy: 0.9687
Accuracy: 0.9686815142631531
```

Figure IV.14 : Evaluation de la précision du modèle dans la phase d'entraînement

In [20]:

```
y_pred=model.predict_classes(X_test)
print('Accuracy of model on test data: ',accuracy score(y test,y pred))

WARNING:tensorflow:From <ipython-input-20-2b012bcfa289>:1: Sequential.predict_classes (from tensorflow.python.keras.engine.sequential) is deprecated and will be removed after 2021-01-01.
Instructions for updating:
Please use instead: * `np.argmax(model.predict(x), axis=-1)`, if your model does multi-class classification (e.g. if it uses a `softmax` last-layer activation). * `(model.predict(x) > 0.5).astype("int32")`, if your model does binary classification (e.g. if it uses a `sigmoid` last-layer activation).
Accuracy of model on test data: 0.9876885195444752
```

Figure IV.15 : comparaison entre y-test et y-pred

Nous avons besoin de y-test en entrée, nous obtenons y-pred, après comparaison nous obtenons la précision du modèle. Dans notre cas la précision était de 0.9876885195444752

In [21]:

```
def proba_failure(machine_id):
    machine_df=df_test[df_test.id==machine_id]
    machine_test=gen_sequence(machine_df,seq_length,seq_cols)
    m_pred=model.predict(machine_test)
    failure_prob=list(m_pred[-1]*100)[0]
    return failure_prob
```

Figure IV.16 : Créer une fonction 'proba failure' pour faire le test

Calculer la probabilité de panne d'une machine donnée. Nous saisissons la période que nous étudierons Pour obtenir la probabilité de défaillance de la machine au cours de cette période particulière et la période que nous voulons étudier pour 30 jours.

IV.4. Résultats obtenus

In [22]: # calculating each machine failure probability

```
probabilities = []
ids = []
for machine_id in range(1, 101):
    ids.append(machine_id)
    probabilities.append(proba_failure(machine_id))
print(ids)
print(probabilities)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 2
6, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 4
9, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 7
2, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 9
5, 96, 97, 98, 99, 100]
[0.01783669, 0.04352033, 0.21296442, 0.043073297, 0.07401407, 0.041642785, 0.04042089, 0.0660
33006, 0.04016161, 0.06937981, 0.043591857, 0.05134642, 0.07289946, 0.039345026, 0.032049417,
0.03708899, 0.6293565, 7.885459, 0.050774217, 97.05482, 0.0541538, 0.031915307, 0.024235249,
78.90358, 0.033780932, 0.030320883, 0.12434125, 0.04826188, 0.07537007, 0.08992255, 99.32828,
0.519076, 0.03798604, 99.11892, 99.32488, 91.20456, 27.127361, 0.2886206, 0.018194318, 75.272
41, 46.832436, 96.14785, 0.12053549, 0.03899932, 0.10716617, 2.1638513, 0.04903078, 0.0468641
52, 96.256325, 0.049373507, 0.084376335, 17.938274, 12.142721, 0.023773313, 0.036862493, 98.6
17325, 0.06260872, 3.067574, 0.021725893, 0.06857216, 41.009193, 1.5156806, 0.2821833, 37.305
717, 0.018930435, 93.7487, 0.035631657, 99.269714, 0.03221035, 0.061011314, 0.038331747, 0.12
721717, 0.035595894, 0.041744113, 0.053983927, 99.29487, 8.304897, 0.016066432, 0.076276064,
```

Figure IV.17 : calculer probabilité des pannes de cent machines

ID machine	Probabilité de panne	ID machine	Probabilité de panne	ID machine	Probabilité de panne
1	0,01783669	34	99,11892	67	0,035631657
2	0,04352033	35	99,32488	68	99,269714
3	0,21296442	36	91,20456	69	0,03221035
4	0,043073297	37	27,127361	70	0,061011314
5	0,07401407	38	0,2886206	71	0,038331747
6	0,041642785	39	0,018194318	72	0,12721717
7	0,04042089	40	75,27241	73	0,035595894
8	0,066033006	41	46,832436	74	0,041744113
9	0,04016161	42	96,14785	75	0,053983927
10	0,06937981	43	0,12053549	76	99,29487
11	0,043591857	44	0,03899932	77	8,304897
12	0,05134642	45	0,10716617	78	0,016066432
13	0,07289946	46	2,1638513	79	0,076276064
14	0,039345026	47	0,04903078	80	0,057452917
15	0,032049417	48	0,046864152	81	99,345825
16	0,03708899	49	96,256325	82	98,13733
17	0,6293565	50	0,049373507	83	0,039175153
18	7,885459	51	0,084376335	84	0,32821298
19	0,050774217	52	17,938274	85	0,03221631
20	97,05482	53	12,142721	86	0,04003644
21	0,0541538	54	0,023773313	87	0,024068356
22	0,031915307	55	0,036862493	88	0,045010448
23	0,024235249	56	98,617325	89	0,039038062
24	78,90358	57	0,06260872	90	46,653168
25	0,033780932	58	3,067574	91	92,09919
26	0,030320883	59	0,021725893	92	87,957565
27	0,12434125	60	0,06857216	93	11,714414
28	0,04826188	61	41,009193	94	0,24526417
29	0,07537007	62	1,5156806	95	0,045511127
30	0,08992255	63	0,2821833	96	0,024855137
31	99,32828	64	37,305717	97	0,041806698
32	0,519076	65	0,018930435	98	0,073724985
33	0,03798604	66	93,7487	99	0,019261241
				100	51,5073

TableauVI.1 : Probabilité de pannes de cent machines.

```
In [23]: # creating the plot
print(len(probabilities))
plt.figure(figsize=[20,10])
plt.plot(ids, probabilities, label="Probabilité des pannes")
plt.legend()
plt.xlabel('Machines ID',fontsize=15)
plt.ylabel('Probabilité',fontsize=15)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)
plt.show()
```

Figure IV.18 : Créer le graphe de probabilité des pannes pour cent machines

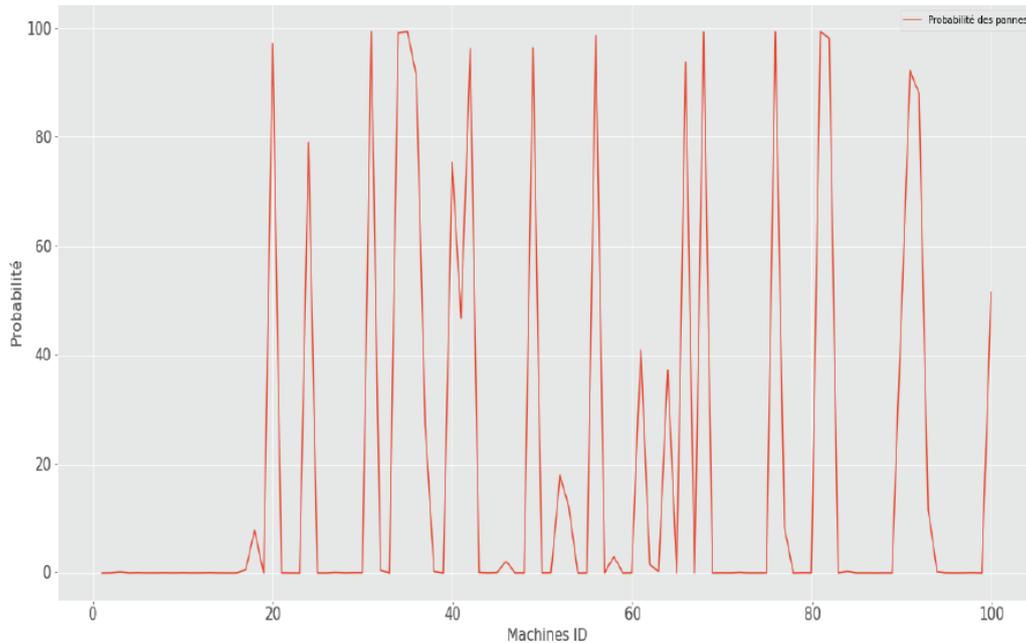


Figure IV.19 : ID machines vs Probabilité.

```
In [24]: # creating the histogram
plt.figure(figsize=[20,15])
plt.bar(ids, probabilities, width = 0.5, color='#0504aa',alpha=0.7)
plt.grid(axis='y')
plt.xlabel('Machine ID',fontsize=15)
plt.ylabel('Probabilité',fontsize=15)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)
plt.title('Histogramme de probabilité des pannes',fontsize=15)
plt.show()
```

Figure IV.20 : créer l'histogramme de probabilité des pannes de cent machines

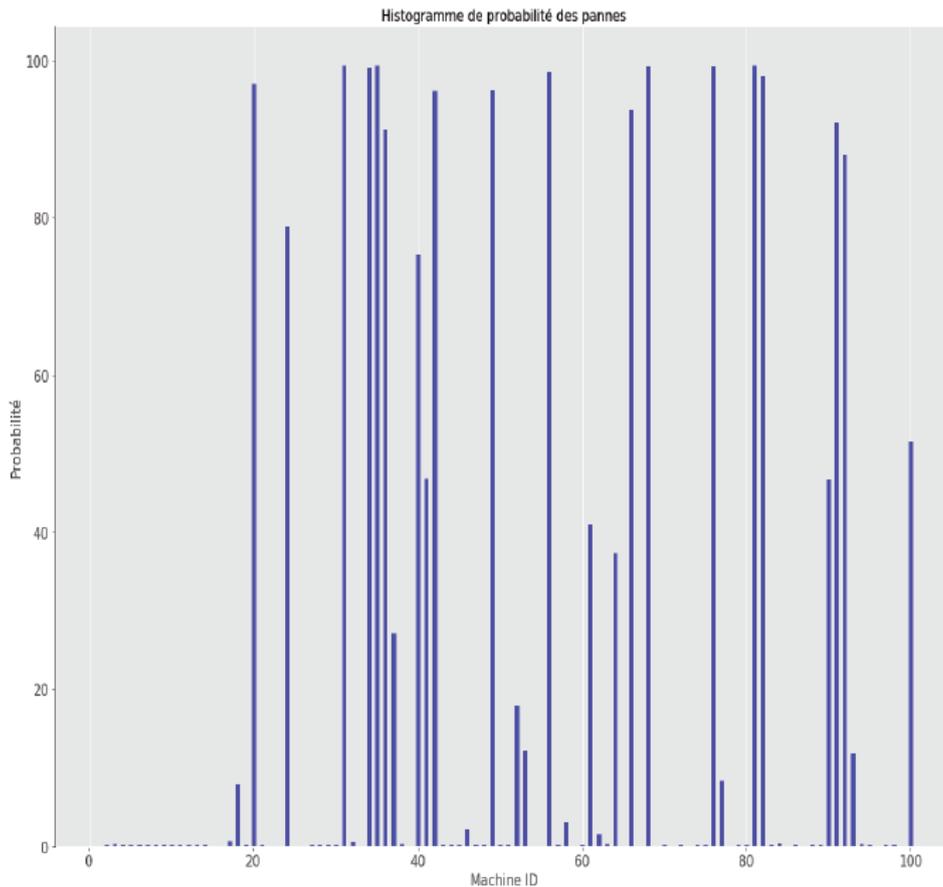


Figure IV.21 : Histogramme des probabilités de pannes

IV.4.1. Analyse des résultats

Nous allons appliquer la méthode Pareto (méthode ABC) pour catégoriser les cents machines en 03 familles, l'exploitation de cette méthode permet de déterminer les machines les plus susceptibles de présenter la probabilité des pannes afin d'établir un programme approprié selon l'ordre de priorité.

IV.4.2. Définition de Pareto

La méthode de Pareto affirme qu'une minorité de causes, d'entrées ou d'efforts généralement conduisent à la majorité des résultats, sorties ou récompenses. Pris littéralement, cela signifie que, par exemple, 80% de ce que vous accomplissez dans votre travail provient de 20% du temps passé au travail. Ainsi, La méthode de Pareto (méthode ABC) stipule qu'il y a un déséquilibre intrinsèque entre les causes et les résultats, les intrants et les extrants, et l'effort et la récompense [36]

IV.4.3. Principe de Pareto (METHODE ABC)

La méthode de Pareto est une méthode d'analyse qui permet de séparer les quelques problèmes essentiels des nombreux problèmes sans importance. Elle permet de choisir à qui ou à quoi s'intéresser en priorité en fonction d'un facteur déterminé. Cette méthode sera de nombreuses fois utilisée en gestion de la qualité pour faire ressortir :

- Les quelques secteurs, défauts, produits ou procédés qui sont les plus coûteux.
- Les quelques causes ou défauts qui sont les plus fréquents.
- Les quelques personnes, poste ou machines qui occasionnent le plus de problèmes, etc.

Le graphe IV.21 tableau IV.2 illustre la méthode Pareto :

Id machine	Probabilité de panne	%		Catégorie	Id machine	Probabilité de panne	%		Catégorie		
78	0,01607	0,00%	environ de 4%	C	10	0,06938	0,00%				
1	0,01784	0,00%			13	0,0729	0,00%				
39	0,01819	0,00%			98	0,07372	0,00%				
65	0,01893	0,00%			5	0,07401	0,00%				
99	0,01926	0,00%			29	0,07537	0,00%				
59	0,02173	0,00%			79	0,07628	0,00%				
54	0,02377	0,00%			51	0,08438	0,00%				
87	0,02407	0,00%			30	0,08992	0,00%				
23	0,02424	0,00%			45	0,10717	0,01%				
96	0,02486	0,00%			43	0,12054	0,01%				
26	0,03032	0,00%			27	0,12434	0,01%				
22	0,03192	0,00%			72	0,12722	0,01%				
15	0,03205	0,00%			3	0,21296	0,01%				
69	0,03221	0,00%			94	0,24526	0,01%				
85	0,03222	0,00%			63	0,28218	0,01%				
25	0,03378	0,00%			38	0,28862	0,02%				
73	0,0356	0,00%			84	0,32821	0,02%				
67	0,03563	0,00%			32	0,51908	0,03%				
55	0,03686	0,00%			17	0,62936	0,03%				
16	0,03709	0,00%			62	1,51568	0,08%				
33	0,03799	0,00%			46	2,16385	0,11%				
71	0,03833	0,00%			58	3,06757	0,16%				
44	0,039	0,00%			18	7,88546	0,41%				
89	0,03904	0,00%			77	8,3049	0,43%				
83	0,03918	0,00%			93	11,7144	0,61%				
14	0,03935	0,00%			53	12,1427	0,63%				
86	0,04004	0,00%			52	17,9383	0,93%				
9	0,04016	0,00%			37	27,1274	1,41%			environ de 16%	B
7	0,04042	0,00%			64	37,3057	1,94%				
6	0,04164	0,00%			90	41,0092	2,13%				
74	0,04174	0,00%			41	46,6532	2,43%				
97	0,04181	0,00%			61	46,8324	2,44%				
4	0,04307	0,00%			100	51,5073	2,68%				
2	0,04352	0,00%			40	75,2724	3,92%			environ de 80%	A
11	0,04359	0,00%			24	78,9036	4,11%				
88	0,04501	0,00%			92	87,9576	4,58%				
95	0,04551	0,00%			36	91,2046	4,75%				
48	0,04686	0,00%			91	92,0992	4,79%				
28	0,04826	0,00%			66	93,7487	4,88%				
47	0,04903	0,00%			42	96,1479	5,00%				
50	0,04937	0,00%	49	96,2563	5,01%						
19	0,05077	0,00%	20	97,0548	5,05%						
12	0,05135	0,00%	82	98,1373	5,11%						
75	0,05398	0,00%	56	98,6173	5,13%						
21	0,05415	0,00%	34	99,1189	5,16%						
80	0,05745	0,00%	68	99,2697	5,17%						
70	0,06101	0,00%	76	99,2949	5,17%						
57	0,06261	0,00%	35	99,3249	5,17%						
8	0,06603	0,00%	31	99,3283	5,17%						
60	0,06857	0,00%	81	99,3458	5,17%						

Tableau IV.2 : Probabilité des pannes et classification selon la méthode Pareto

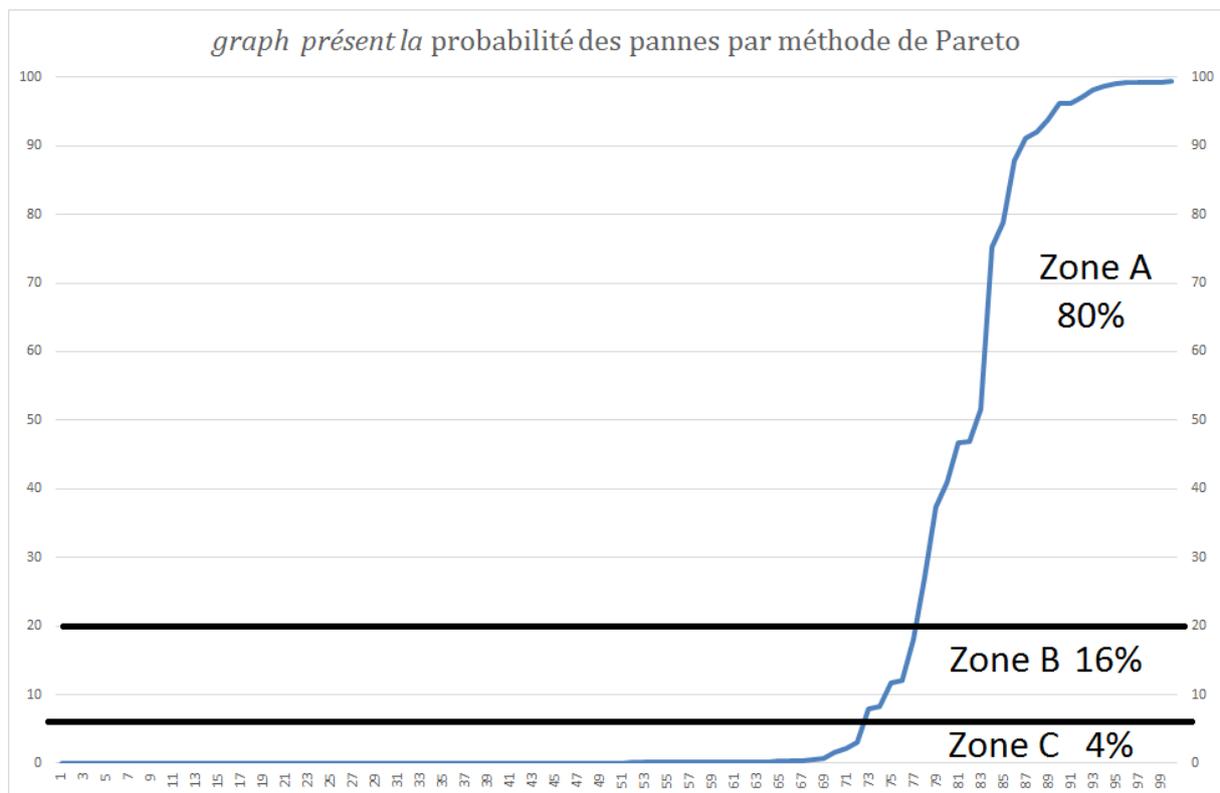


Figure IV.22 : Probabilité des pannes et zones Pareto

Interprétation de résultat :

- ZONE A : 80% des équipements doivent être examinés, Que ce soit pour une réparation ou un remplacement.
- ZONE B : 16% des équipements sont normaux, qu'ils soient supervisés.
- ZONE C : 4% des autres équipements sont assez bons pour être négligé.

IV.5. Conclusion

L'objectif principal de la maintenance prédictive est de prévoir quand des pannes d'équipement peuvent survenir. Il s'agit ensuite de prévenir cette défaillance en prenant des mesures appropriées. Le système de maintenance préventive prédictive surveille les futures défaillances et planifie la maintenance à l'avance.

Cela permet de réaliser plusieurs économies

- Réduit la fréquence de maintenance.
- Minimise le temps passé sur un équipement particulier en cours de maintenance, et permet à ce temps d'être productif.
- Minimise le coût de la maintenance.

Les techniques ci-dessus nécessitent une quantité importante de données d'entraînement comprenant suffisamment de scénarios de défaillance. Comme les défaillances sont rares, la collecte de données peut prendre beaucoup de temps. Cela reste un obstacle important à l'application de la maintenance prédictive.

Conclusion générale et perspectives

Le travail de recherche présenté dans ce mémoire porte sur l'étude de la gestion assistée par ordinateur en utilisant les réseaux de neurones artificiels, qui sont le cœur de plusieurs applications réelles et un outil indispensable qui a fait ses preuves dans la pratique par leurs caractéristiques de généralisation et de robustesses. Alors ces réseaux-là ne sont en fait qu'un :

- Ensemble de neurones formels inspiré du neurone biologique.
- Relié entre eux par des synapses contenant des poids.
- Utilisant un nombre d'algorithmes d'apprentissage afin de réaliser une tâche généralement : la classification, prédiction, approximation.

Nous nous sommes concentrés principalement dans ce projet sur la gestion maintenance assistée par ordinateur : maintenance préventive d'un processus industriel

Cette étude a été structurée autour de quatre chapitres essentiels. Le premier consacré à une introduction au domaine généralité sur la maintenance il a permis de présenter des généralités sur la maintenance

Dans le second, Nous avons parlé de la gestion maintenance assistée par ordinateur. Alors que dans le troisième en a parlé du Machine learning et machine deep learning Enfin le quatrième et dernier chapitre, a fait l'objet d'une étude concernant Conception et Implémentation

On peut conclure que l'utilisation des réseaux de neurones a beaucoup contribué au développement de la GMAO et à la génération d'algorithmes robuste dans la prédiction des pannes dans le domaine industriel. Ce champ de recherche reste ouvert pour apporter de nouveaux algorithmes dont le but de développer le domaine industriel par introduire l'intelligence artificielle dans la gestion maintenance.

Bibliographie

- [1] Vernier, François Monchy Jean-Pierre. "MAINTENANCE Méthodes et organisations
- [2] 3ème édition, l'USINE NOUVELLE DUNOD
- [3] Monchy, François. "Maintenance, Méthodes et Organisation, dunod." (2000)
- [4] MONCHY FRANCOIS / LA FONCTION de maintenance / MASSON/1997
- [5] <https://www.technologuepro.com/cours-maintenance-industrielle/La-maintenance-industrielle.htm>
- [6] <https://www.piloter.org/qualite/cinq-zero.htm>
- [7] <http://stmg.education/les-dicos/dico-management/zero.html>
- [8] Gestion de la maintenance industrielle D. Richet www.cxp.fr 2004
- [9] Le 19e panorama de la GMAO Francis Vasse Gilles Zwingelstein
- [10] www.afim.asso.fr 2005
- [11] Appels d'offres dans les entreprises industrielles Enjeux, Pratiques et Informatisation Véronique, Athané Mémoire du Magister www.hec.unil.ch 2000
- [12] La gestion de la maintenance assistée par ordinateur et la maintenance des logiciels Nicolas Buchy www.lrgl.uqam.ca 2002
- [13] Conception et réalisation d'un système de gestion de la maintenance assistée par ordinateur Laala Amor Mémoire du Magister, 2001
- [14] Gestion de la maintenance assistée par ordinateur M. Gabriel Y. Pimor
- [15] www.cyber.uhp-nancy.fr 2004
- [16] Claude Touzer, << les réseaux de neurones artificiels >>, Livre, 1992.
- [17] E. Davalo, P. Naim, << Des réseaux de neurones >>, éd. Eyrolles, Paris, 1993.
- [18] G.Z winngelsten, << Diagnostic des défaillances : théorie et pratique pour les systèmes industriel >>, Ed. Hermès paris, 1995.
- [19] L. personnaz, I.Rivals, << réseaux de neurones formels pour la modélisation, la commande et la classification >>, CNRS éditions, collection Science et Techniques de l'Ingénieur, 2003.
- [20] J.L.Amat, G.Yahhiaoui, << Technique avancées pour le traitement de l'information, réseaux de neurones, logique flou, algorithme génétique >>, Edition CEPAUDIES, Toulouse France, 1996.
- [21] M.Gondra, C.Miller, << Réseaux de neurones et apprentissage >>, Revue REE n° 3, pp. 6-9, 1997.
- [22] Y.Ziquel, J.Ermont, << Apprentissage par réseaux de neurones >>, Rapport de projet, 2004-2005
- [23] T.A. Freman, D. M.Skapura, << Neural networks : algorithm, applications and programming techniques >>, CNS, Computation and neural systems series, 1992

[24] D.Lefebvre, A.E.Moudni, <<Détection et localisation des défaillances dans les systèmes de production par réseaux sz nzueonzq, logique flou, algorithme génétique >>, Edition CEPAUDIES, Toulouse France, 1996

[25] <https://www.lebigdata.fr/machine-learning-et-big-data>

[26] <https://www.ionos.fr/digitalguide/web-marketing/search-engine-marketing/deep-learning-vs-machine-learning/>.

[27] <https://robankhood.com/deep-learning/deep-learning-explications/deep-leaning-vs-machine-learning/>.

[28] <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>

[29] Coelho, L.P. and W. Richert, *Building machine learning systems with Python*. 2015: Packt Publishing Ltd.

[30] Ng, K.K., Y. B. Mourri, *Neural Networks and Deep Learning* 2017, Coursera: www.coursera.org.

[31] http://jupyternotebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html.

[32] <http://pandas.pydata.org/pandas-docs/stable/index.html>.

[33] <https://pandas.pydata.org/about.html>.

[34] <http://scikitlearn.org/stable/index.html#>.

[35] <https://keras.io/>.

[36] Koch, R. (1998). *praise for the 80/20 principle*

Annexes

Annexe A :

Dégradation de turboréacteur base de données de simulation,

README – FROM ZIPFILE. Téléchargé depuis le site de la NASA

La base de données est :

Base d données	Les tricycles de train	Tester les trajectoires	Conditions	Modes de défaillance
FD001	100	100	1(niveau de la mer)	dégradation HPC
FD002	260	259	6	dégradation HPC
FD003	100	100	1	dégradation du HPC, dégradation du ventilateur
FD004	149	148	6	dégradation du HPC, dégradation du ventilateur

Les bases de données sont constituées de multiples séries. Chronologiques multi variées. Chaque base de données est ensuite divisée en sous-ensembles de formation et de test. Chaque série chronologique provient d'un moteur différent, c'est-à-dire que les données peuvent être considérées comme provenant d'une flotte de moteurs du même type. Chaque moteur démarre avec un degré d'usure initial différent et une variation de fabrication inconnue de l'utilisateur. Cette usure et cette variation sont considérées comme normales, c'est-à-dire qu'elles ne sont pas considérées comme un état de défaut.

Il existe trois réglages de fonctionnement qui ont un effet important sur les performances du moteur. Ces réglages sont également inclus dans les données. Les données sont contaminées par le bruit des capteurs.

Le moteur fonctionne normalement au début de chaque série chronologique, et développe un défaut à un moment donné au cours de la série. Dans la série d'entraînement, le défaut augmente en importance jusqu'à la défaillance du système. Dans le kit de test, la série chronologique se termine quelque temps avant la défaillance du système. L'objectif de la compétition est de prédire le nombre de cycles opérationnels restants avant la défaillance dans la série de test, c'est-à-dire le nombre de cycles opérationnels après le dernier cycle que le moteur continuera à fonctionner. Un vecteur des valeurs réelles de la durée de vie utile restante (RUL) a également été fourni pour les données d'essai.

Les données sont fournies sous la forme d'un fichier texte compressé au format zip comportant 26 colonnes de chiffres, séparées par des espaces. Chaque ligne est un instantané des données prises au cours d'un seul cycle opérationnel, chaque colonne est une variable différente. Les colonnes correspondent à :

- 1) numéro de l'unité
- 2) le temps, par cycles
- 3) cadre opérationnel 1
- 4) cadre opérationnel 2
- 5) cadre opérationnel 3
- 6) mesure par capteur 1
- 7) mesure par capteur 2
- ...
- ...
- 26) mesure par capteur 26

Annexe B : code

Code depuis “Split_data.ipynb” et “Split_data.py”

```
In [1]: # importing required libraries

import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix, accuracy_score
from keras.models import Sequential
from keras.layers import Dense, Dropout, LSTM, Activation
from keras.callbacks import EarlyStopping

import matplotlib.pyplot as plt
plt.style.use('ggplot')
%matplotlib inline

In [2]: # importing train dataset
dataset_train=pd.read_csv('data/train_FD001.txt', sep=' ', header=None).drop([26,27], axis=1)
col_names = ['id', 'cycle', 'setting1', 'setting2', 'setting3', 's1', 's2', 's3', 's4', 's5', 's6', 's7', 's8', 's9', 's10', 's11', 's12', 's13', 's14', 's15', 's16', 's17', 's18', 's19', 's20', 's21']
dataset_train.columns=col_names
print('Shape of Train dataset: ', dataset_train.shape)
dataset_train.head()

In [3]: # importing test dataset
dataset_test=pd.read_csv('data/test_FD001.txt', sep=' ', header=None).drop([26,27], axis=1)
dataset_test.columns=col_names
print('Shape of Test dataset: ', dataset_test.shape)
dataset_test.head()

In [4]: # importing RUL dataset
pm_truth=pd.read_csv('data/RUL_FD001.txt', sep=' ', header=None).drop([1], axis=1)
pm_truth.columns=['more']
pm_truth['id']=pm_truth.index+1
pm_truth.head()

In [5]: # generate column max for test data
rul = pd.DataFrame(dataset_test.groupby('id')['cycle'].max()).reset_index()
rul.columns = ['id', 'max']
rul.head()

In [6]: # run to failure
pm_truth['rtf']=pm_truth['more']+rul['max']
pm_truth.head()

In [7]: # calculating TTF in the test table
pm_truth.drop('more', axis=1, inplace=True)
dataset_test=dataset_test.merge(pm_truth, on=['id'], how='left')
dataset_test['ttf']=dataset_test['rtf'] - dataset_test['cycle']
dataset_test.head()

In [8]: # removing RTF from the test table
dataset_test.drop('rtf', axis=1, inplace=True)
dataset_test.head()

In [9]: # calculating TTF in the train table
dataset_train['ttf'] = dataset_train.groupby(['id'])['cycle'].transform(max)-dataset_train['cycle']
```

```

In [10]: # adding label_bc to test and train tables
df_train=dataset_train.copy()
df_test=dataset_test.copy()
period=30
df_train['label_bc'] = df_train['ttf'].apply(lambda x: 1 if x <= period else 0)
df_test['label_bc'] = df_test['ttf'].apply(lambda x: 1 if x <= period else 0)
df_train.head()

In [11]: features_col_name=['setting1', 'setting2', 'setting3', 's1', 's2', 's3', 's4', 's5', 's6',
's7', 's8', 's9', 's10', 's11',
's12', 's13', 's14', 's15', 's16', 's17', 's18', 's19', 's20', 's21']
target_col_name='label_bc'

In [12]: sc=MinMaxScaler()
df_train[features_col_name]=sc.fit_transform(df_train[features_col_name])
df_test[features_col_name]=sc.transform(df_test[features_col_name])

In [13]: # function to generate sequences
def gen_sequence(id_df, seq_length, seq_cols):
    df_zeros=pd.DataFrame(np.zeros((seq_length-1, id_df.shape[1])),columns=id_df.columns)
    id_df=df_zeros.append(id_df,ignore_index=True)
    data_array = id_df[seq_cols].values
    num_elements = data_array.shape[0]
    lstm_array=[]
    for start, stop in zip(range(0, num_elements-seq_length), range(seq_length, num_elements
)):
        lstm_array.append(data_array[start:stop, :])
    return np.array(lstm_array)

# function to generate labels
def gen_label(id_df, seq_length, seq_cols,label):
    df_zeros=pd.DataFrame(np.zeros((seq_length-1, id_df.shape[1])),columns=id_df.columns)
    id_df=df_zeros.append(id_df,ignore_index=True)
    data_array = id_df[seq_cols].values
    num_elements = data_array.shape[0]
    y_label=[]
    for start, stop in zip(range(0, num_elements-seq_length), range(seq_length, num_elements
)):

In [14]: # timestamp or window size
seq_length=50
seq_cols=features_col_name

In [15]: # generate X_train
X_train=np.concatenate(list(list(gen_sequence(df_train[df_train['id']==id], seq_length, seq_
cols)) for id in df_train['id'].unique()))
print(X_train.shape)
# generate y_train
y_train=np.concatenate(list(list(gen_label(df_train[df_train['id']==id], 50, seq_cols,'label_
_bc')) for id in df_train['id'].unique()))
print(y_train.shape)

(20531, 50, 24)
(20531,)

In [16]: # generate X_test
X_test=np.concatenate(list(list(gen_sequence(df_test[df_test['id']==id], seq_length, seq_col
s)) for id in df_test['id'].unique()))
print(X_test.shape)
# generate y_test
y_test=np.concatenate(list(list(gen_label(df_test[df_test['id']==id], 50, seq_cols,'label_b
c')) for id in df_test['id'].unique()))
print(y_test.shape)

```

```
In [17]: # creating the model
nb_features =X_train.shape[2]
timestamp=seq_length

model = Sequential()

model.add(LSTM(
    input_shape=(timestamp, nb_features),
    units=100,
    return_sequences=True))
model.add(Dropout(0.2))

model.add(LSTM(
    units=50,
    return_sequences=False))
model.add(Dropout(0.2))

model.add(Dense(units=1, activation='sigmoid'))
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 50, 100)	50000
dropout (Dropout)	(None, 50, 100)	0
lstm_1 (LSTM)	(None, 50)	30200
dropout_1 (Dropout)	(None, 50)	0
dense (Dense)	(None, 1)	51

Total params: 80,251
Trainable params: 80,251
Non-trainable params: 0

```
In [18]: # fit the network
model.fit(X_train, y_train, epochs=10, batch_size=200, validation_split=0.05, verbose=1,
    callbacks = [EarlyStopping(monitor='val_loss', min_delta=0, patience=0, verbose=0,
```

```
Epoch 1/10
98/98 [=====] - 16s 162ms/step - loss: 0.1773 - accuracy: 0.9328 - v
al_loss: 0.0716 - val_accuracy: 0.9649
Epoch 2/10
98/98 [=====] - 15s 152ms/step - loss: 0.0801 - accuracy: 0.9671 - v
al_loss: 0.0587 - val_accuracy: 0.9766
Epoch 3/10
98/98 [=====] - 15s 152ms/step - loss: 0.0659 - accuracy: 0.9725 - v
al_loss: 0.0815 - val_accuracy: 0.9611
```

```
In [19]: # training metrics
scores = model.evaluate(X_train, y_train, verbose=1, batch_size=200)
print('Accuracy: {}'.format(scores[1]))
```

```
103/103 [=====] - 6s 55ms/step - loss: 0.0692 - accuracy: 0.9703
Accuracy: 0.9702888131141663
```

```
In [21]: # creating the function to calculate the probability of failure
def prob_failure(machine_id):
    machine_df=df_test[df_test.id==machine_id]
    machine_test=gen_sequence(machine_df, seq_length, seq_cols)
    m_pred=model.predict(machine_test)
    failure_prob=list(m_pred[-1]*100)[0]
    return failure_prob
```

```
In [22]: # calculating each machine failure probability
probabilities = []
ids = []
for machine_id in range(1, 101):
    ids.append(machine_id)
    probabilities.append(prob_failure(machine_id))
print(ids)
print(probabilities)
```

```
In [23]: # creating the plot
print(len(probabilities))
plt.figure(figsize=[20,10])
plt.plot(ids, probabilities, label="Probabilité des pannes")
plt.legend()
plt.xlabel('Machines ID', fontsize=15)
plt.ylabel('Probabilité', fontsize=15)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)
plt.show()
```

```
In [24]: # creating the histogram
plt.figure(figsize=[20,15])
plt.bar(ids, probabilities, width = 0.5, color='#0504aa', alpha=0.7)
plt.grid(axis='y')
plt.xlabel('Machine ID', fontsize=15)
plt.ylabel('Probabilité', fontsize=15)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)
plt.title('Histogramme de probabilité des pannes', fontsize=15)
plt.show()
```