

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Boudiaf - M'sila

Faculté De Technologie
Département : Génie Électrique
N° :



Filière : Electromécanique
Option : Maintenance Industrielle

Mémoire présenté pour l'obtention
Du diplôme de Master Académique
Par : Abderrahmane JEDAA

Intitulé :

Développement d'un programme de gestion de la maintenance industrielle

Soutenu le 06/06/2024 devant le jury composé de :

Dr. BELKHIRI Salah	Université de M'sila	Président
Dr. DEFDAF Mabrouk	Université de M'sila	Encadreur
Pr. GHEMARI Zine	Université de M'sila	Examineur
Pr. DJERIOUI Ali	Université de M'sila	Examineur
Mr. YAHIAOUI Slimane	SONELGAZ (SPE) M'sila	Invité

Année universitaire : 2023/2024

Dédicaces

Je dédie ce mémoire :

À ma très chère mère, pour son amour inconditionnel, sa patience infinie, son soutien indéfectible et ses encouragements qui m'ont porté tout au long de ma vie. Tu es ma source de force et d'inspiration, et je te remercie du fond du cœur pour tout ce que tu as fait pour moi.

À mon frère et mes sœurs et mon âme-sœur, pour leur présence constante et leur amitié sincère.

À l'âme de mon père, ton absence me pèse chaque jour, mais ton souvenir me guide et me motive à toujours donner le meilleur de moi-même.

À mes oncles et ma famille, pour votre amour, votre soutien et vos encouragements. Vous avez toujours cru en moi et en mes capacités.

À tous les professeurs, que ce soit du primaire, du collège, du lycée ou de l'enseignement supérieur, qui ont marqué notre vie par leur passion, leur savoir et leur pédagogie.

Je dédie également ce mémoire à toutes les personnes qui m'ont soutenu et encouragé tout au long de ma vie.

Avec tout mon amour et ma gratitude

Remerciements

Je présente également mes profondes gratitudee au Docteur **Mabrouk DEFDAF** qui a toujours eu le temps de répondre à mes questions, de me guider et de me faire profiter de son extrême gentillesse.

Ce travail n'aurait pas pu être aussi riche et n'aurait pas pu aboutir sans son aide. Je le remercie pour la qualité de son encadrement exceptionnel, pour sa patience, sa rigueur et sa disponibilité durant la préparation de ce mémoire.

Enfin, un immense merci à mes plus chers proches, famille et ami(e)s, pour le soutien indéfectible qu'ils m'ont apporté tout au long de ce parcours.

Table des matières

Liste des figures	i
Liste des tableaux.....	iii
Liste des abréviations.....	iv
Introduction générale	1
Chapitre I : généralités sur la maintenance	3
Introduction :	3
I.1 Historique de la maintenance :	3
I.2 Définition	3
I.3 Les formes de la maintenance :	4
I.3.1 Maintenance corrective :	4
I.3.2 Maintenance préventive :	4
I.4 Classification des tâches de maintenance	5
I.4.1 Tâches de maintenance du premier niveau :	5
I.4.2 Tâches de maintenance de deuxième niveau.....	6
I.4.3 Tâches de maintenance de troisième niveau	6
I.4.4 Tâches de maintenance du quatrième niveau.....	6
I.4.5 Tâches de maintenance du cinquième niveau	6
I.5 Les opérations de maintenance	6
I.5.1 Les opérations de maintenance corrective.....	6
I.5.2 Les opérations de maintenance préventive.....	7
I.6 Les stratégies et les décisions associées.....	7
I.7 Autre formes et méthodes de maintenance	9
I.7.1 La maintenance améliorative	9
I.7.2 La total productive maintenance (T.P.M)	9
Conclusion :	10

Chapitre II : L'analyse FMD par les lois de probabilité	11
Introduction :	11
II.1 Définitions	11
II.1.1 La fiabilité :	11
II.1.2 La maintenabilité :	11
II.1.3 La disponibilité :	12
II.2 Indicateurs opérationnels d'FMD	12
II.2.1 Indicateurs de fiabilité.....	13
II.2.2 Indicateurs de maintenabilité	13
II.2.3 Indicateurs de disponibilité	13
II.3 Expressions mathématiques de la fiabilité.....	14
II.3.1 Fonctions de distribution et de répartition	14
II.3.2 Taux de défaillance	15
II.4 Les principales lois	15
II.4.1 Loi exponentielle	15
II.4.2 La loi de Weibull	16
II.4.3 Loi log-normale :	17
II.5 Analyse de la fiabilité et la maintenabilité à partir du modèle exponentiel.....	17
II.5.1 Caractéristiques.....	18
II.5.2 Représentations graphiques de la loi exponentielle	19
II.6 Analyse de la fiabilité et la maintenabilité à partir du modèle de Weibull	19
II.6.1 Expressions mathématiques caractéristiques	20
II.6.2 Structure du papier de Weibull	21
II.6.3 Méthodologie de l'analyse de fiabilité.....	23
II.7 Analyse de la fiabilité à partir du modèle de log-normal	26
II.7.1 Expressions mathématiques caractéristiques	26
II.7.2 Représentation graphique.....	27

II.8 Approche mathématique de la disponibilité	27
II.9 Test Kolmogorov Smirnov	28
Conclusion :	29
Chapitre III : la réalisation du programme.....	30
Introduction :	30
III.1 Choix du Langage de Programmation : Python	30
III.1.1 Polyvalence et puissance	30
III.1.2 Bibliothèques spécialisées	30
III.1.3 Lisibilité et Simplicité	30
III.1.4 Large Communauté et Ressources	30
III.1.5 Flexibilité et évolutivité.....	31
III.2 Les Bibliothèques Utilisées dans le Programme.....	31
III.2.1 Tkinter	31
III.2.2 NumPy.....	31
III.2.3 SciPy et SciPy.stats.....	32
III.2.4 Pandas	33
III.2.5 Matplotlib	33
III.3 Les Objectifs et Spécifications du Programme FMD.....	34
III.3.1 Objectifs.....	34
III.3.2 Spécifications :	35
III.4 Le Code Source pour le Programme FMD :	36
III.4.1 Développer le Frontend avec Tkinter :.....	36
III.4.2 Développer le Backend :.....	39
III.5 Tests de validation :	50
III.5.1 Préparation des données	51
III.5.2 Importer les données.....	52
III.5.3 Le choix d'analyse.....	53

Conclusion :.....	60
Conclusion générale.....	61
Perspective	62
Référence :	63

Liste des figures :

Fig I. 1 : digramme de la différents formes de la maintenance.....	4
Fig II. 1 : Les durées caractéristique de FMD	12
Fig II. 2 : Fonction de distribution.....	15
Fig II. 3 : Fonction de répartition.....	15
Fig II. 4 : Représentation graphique de la loi exponentielle	19
Fig II. 5 : variabilité de la loi de weibull sous l'influence de β	20
Fig II. 6 : Description du papier Weibull.....	22
Fig II. 7 : Exemple du traçage du nuage de points et du droit de weibull	24
Fig II. 8 : La Détermination des valeurs des trois paramètres β, η, γ	24
Fig II. 9 : les techniques de redressement.....	25
Fig II. 10 : variabilité de la loi de Log-normal sous l'influence de σ	27
Fig III. 1 : l'objectif de la programme	34
Fig III. 2 : L'interface d'utilisateur de la programme « 317_FMD »	39
Fig III. 3 : L'importation de donnes au programme « 317_FMD ».....	52
Fig III. 4 : L'affichage de donnée.....	52
Fig III. 5 : Les choix de l'analyse.....	53
Fig III. 6 : Procédure de l'analyse	53
Fig III. 7 : Tracé de la droite de Weibull pour les TBFs.....	54
Fig III. 8 : Les résultats de calcul de la fiabilité à l'aide de loi de weibull afficher par le programme.....	54
Fig III. 9 : Courbe de la fiabilité à l'aide de loi de weibull	55
Fig III. 10 : Courbe de la fonction de répartition à l'aide de loi de weibull	55
Fig III. 11 : Courbe de la densité de probabilité pour la fiabilité à l'aide de loi de weibull.....	55
Fig III. 12 : courbe de la Taux de défaillance à l'aide de loi de weibull	55
Fig III. 13 : Les résultats de calcul de la fiabilité à l'aide de loi log-normal afficher par le programme	55
Fig III. 14 : Courbe de la fiabilité à l'aide de loi Log-normal.....	56
Fig III. 15 : Courbe de la Taux de défaillance à l'aide de loi Log-normal.....	56
Fig III. 16 : Courbe de la densité de probabilité pour la fiabilité à l'aide de loi Log-normal	56
Fig III. 17 : Courbe de la Taux de défaillance à l'aide de loi de Log-normal.....	56

Fig III. 18 : Tracé de la droite de Weibull pour les TTRs	56
Fig III. 19 : Les résultats de calcul de la maintenabilité à l'aide de loi de Weibull afficher par le programme	57
Fig III. 20 : Courbe de la maintenabilité à l'aide loi de weibull.....	57
Fig III. 21 : Courbe densité de probabilité de la maintenabilité à l'aide de loi de weibull.....	57
Fig III. 22 : Courbe de la taux de réparation à l'aide loi de weibull.....	57
Fig III. 23 : Les résultats de calcul de la maintenabilité afficher par la programme	58
Fig III. 24 : Courbe de la maintenabilité à l'aide loi Log-normal.....	58
Fig III. 25 : Courbe densité de probabilité à l'aide de loi log-normal	58
Fig III. 26 : Courbe de la taux de réparation à l'aide loi log-normal.....	58
Fig III. 27 : Les résultats de calcul de la Disponibilité afficher par la programme	59
Fig III. 28 : Courbe de la disponibilité.....	59

Liste des tableaux

Tableau I. 1 : Les différents contextes de la décision à travers les réponses aux questions pourquoi, quand, où et quoi	8
Tableau II. 1 : l'hypothèse de test Kolmogorov-Smirnov	28
Tableau III. 1 : l'historique des pannes de four à billette de la presse 1600T	51

Liste des abréviations :

FMD : La fiabilité, la maintenabilité la disponibilité.

TPM : Total productive maintenance.

TBF : Temps de bon fonctionnement.

TTR : Temps de réparation.

MTTF : (mean time to first failure), temps moyen avant-première défaillance.

MTBF : (mean time between failure), temps moyen entre deux défaillances successives.

MDT : (mean down time), temps moyen d'indisponibilité ou temps moyen d'arrêt propre.

MUT : (mean up time), temps moyen de disponibilité.

MTTR : (mean time to repair), temps moyen de réparation.

GUI : Interface utilisateur graphique.

CDF : Fonction de répartition cumulative.

Introduction générale

La gestion de la maintenance moderne s'inscrit désormais dans une démarche globale de gestion des actifs physiques, visant à maximiser la valeur économique des équipements tout au long de leur cycle de vie [1]. La gestion de la maintenance joue un rôle essentiel dans les opérations industrielles pour garantir la fiabilité, la maintenabilité et la disponibilité des équipements critiques. Dans un contexte où une production ininterrompue est primordiale, la maintenance englobe diverses approches, allant des mesures réactives aux stratégies proactives. Ces efforts sont soigneusement orchestrés pour préserver le bon fonctionnement de systèmes complexes [2].

L'analyse de la Fiabilité, de la Maintenabilité et de la Disponibilité (FMD) s'impose comme un outil indispensable pour les entreprises qui souhaitent optimiser la gestion de leurs actifs et maximiser leur rentabilité. En s'appuyant sur des modèles probabilistes et des données opérationnelles, cette approche rigoureuse permet de prédire la durée de vie des équipements, d'identifier les composants les plus susceptibles de tomber en panne et de planifier les interventions de maintenance en conséquence [3]. Les métriques opérationnelles telles que la fiabilité, la maintenabilité et la disponibilité sont exprimées mathématiquement avec précision et analysées à l'aide de modèles probabilistes sophistiqués, y compris les distributions exponentielles, de Weibull et log-normale. Ces analyses perspicaces permettent une prise de décision éclairée, favorisant l'optimisation des stratégies de maintenance et l'amélioration de la productivité globale.

Pour naviguer efficacement dans les méandres de la gestion de la maintenance et de l'analyse FMD, nous nous immergeons dans le monde de la programmation Python, en exploitant la puissance de bibliothèques spécialisées pour construire un programme pour les analyses FMD.

Ce programme fournit des informations précieuses, présentées sous forme de tableaux et de graphiques clairs et concis, éclairant la voie vers des stratégies de maintenance optimisées. En fin de compte, l'objectif ultime est de favoriser une prise de décision éclairée, d'accroître l'efficacité opérationnelle et de maximiser l'utilisation des actifs.

Le programme facilite également l'analyse FMD en rationalisant le processus de suivi et de résolution des retours clients. L'analyse des données FMD permet d'identifier les problèmes récurrents et les défauts de conception potentiels, menant à des actions correctives qui améliorent.

Le programme agit comme un guide précieux dans la complexité de la gestion de la maintenance et de l'analyse FMD. En transformant vos données en informations exploitables, il favorise une prise de décision éclairée, stimule l'efficacité opérationnelle et maximise l'utilisation des actifs, pour une activité commerciale plus résiliente et rentable.

Ce mémoire se divise en trois chapitres, Le premier chapitre présente une introduction à la maintenance, en abordant son historique, sa définition, ses différentes formes, la classification des tâches de maintenance, les stratégies et décisions associées, ainsi que d'autres formes et méthodes de maintenance telles que la maintenance préventive et la maintenance totale productive (TPM).

Le deuxième chapitre se concentre sur l'analyse de FMD en utilisant les lois de probabilité. Il définit les concepts clés de l'analyse FMD et présente ses indicateurs opérationnels, explore les expressions mathématiques de la fiabilité, détaille les principales lois de probabilité utilisées dans l'analyse FMD, et applique ces lois à des modèles spécifiques tels que les modèles exponentiels, de Weibull et log-normal. De plus, ce chapitre aborde aussi l'approche mathématique de la disponibilité et le test de Kolmogorov-Smirnov pour la validation des modèles.

Le troisième chapitre décrit le processus de développement du programme. Il commence par justifier le choix de Python comme langage de programmation, puis présente les bibliothèques spécialisées utilisées dans le programme. Ensuite, il détaille les objectifs et spécifications du programme FMD, avant de présenter le code source complet du programme avec des explications détaillées. Enfin, ce chapitre présente les tests de validation effectués pour garantir la fiabilité et la précision du programme.

Chapitre I :

Généralités sur la
Maintenance

Introduction :

La maintenance est une discipline cruciale qui joue un rôle essentiel dans la pérennité et l'efficacité des équipements industriels et des infrastructures. Elle englobe un ensemble de techniques et de méthodologies destinées à assurer le bon fonctionnement et la sécurité des systèmes tout au long de leur cycle de vie.

I.1 Historique de la maintenance :

Terme "Maintenance" : Le mot "maintenance" vient des racines latines "manus" (main) et "tenere" (tenir), apparu au XIIe siècle en français. Il évoque l'idée de soutien ou d'entretien.

Usage Littéraire : Le terme est utilisé dans la littérature française depuis le XVIe siècle et a pris une connotation moderne dans le vocabulaire militaire anglo-saxon pour désigner le maintien des effectifs et du matériel à un niveau constant¹.

Évolution de la Maintenance : La maintenance a grandement évolué au cours du dernier demi-siècle, passant d'une réaction aux défaillances à une approche proactive et améliorative, intégrant des stratégies préventives et correctives.

Qualité et Sécurité : Dans un contexte de concurrence accrue, la maintenance est cruciale pour assurer la qualité et la sécurité des produits et services, en évitant les défaillances et en garantissant une exploitation sûre des moyens de production [4]

I.2 Définition

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.

I.3 Les formes de la maintenance : [5]

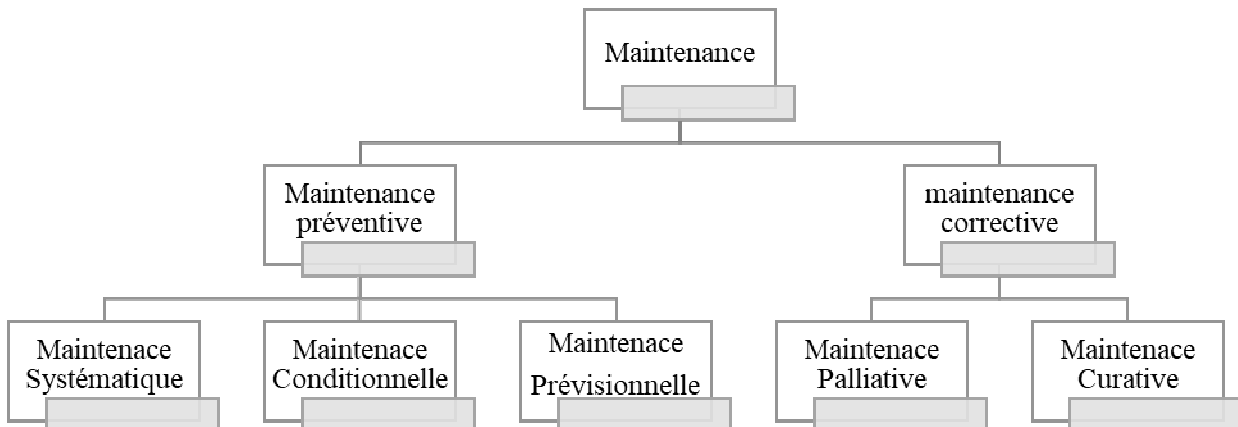


Fig I. 1 : digramme de la différents formes de la maintenance

I.3.1 Maintenance corrective :

C'est une « Maintenance exécutée après la détection d'une panne et destinée à remettre un bien dans un état dans lequel il peut accomplir une fonction requise ». Deux types de maintenance corrective peuvent être mis en œuvre :

I.3.1.1 La maintenance curative

Qui constitue une réparation définitive du système, et vise à supprimer tout dysfonctionnement survenu.

I.3.1.2 La maintenance palliative

Qui constitue une réparation à caractère provisoire, celle-ci intervient lorsque la défaillance empêche de poursuivre l'exploitation du système de production, et a pour but de le remettre en état afin d'assurer une ou plusieurs de ses fonctions.

I.3.2 Maintenance préventive :

C'est une « Maintenance exécutée à des intervalles prédéterminés ou selon des critères prescrits et destinée à réduire la probabilité de défaillance ou la dégradation du fonctionnement d'un bien ». La maintenance préventive sert à entretenir le système, afin de le maintenir en bon état de fonctionnement. Trois stratégies de maintenances préventives peuvent être mises en œuvre :

I.3.2.1 La maintenance systématique

« Maintenance préventive exécutée à des intervalles de temps préétablis ou selon un nombre défini d'unités d'usage mais sans contrôle préalable de l'état du bien ». Cette politique de maintenance s'applique à des équipements :

- Soumis à une réglementation sécuritaire (ponts roulants, matériels d'incendie, etc.),
- Présentant des coûts de défaillance très élevés (système avec processus de production continu, lignes de fabrication automatisés, etc.)
- Pour lesquels une défaillance peut entraîner des accidents graves (matériels de transport en commun des personnes, appareils médicaux, etc.).

I.3.2.2 La maintenance prévisionnelle :

« Maintenance préventive exécutée selon un calendrier préétabli ou selon un nombre défini d'unités d'usage ». Ce type de maintenance se base sur l'estimation du temps de fonctionnement correct du système, qui peut être établie par l'analyse de différentes mesures sur le système. Cette politique peut s'appliquer à tous les équipements, mais elle est généralement coûteuse.

I.3.2.3 La maintenance conditionnelle

« Maintenance préventive basée sur une surveillance du fonctionnement du bien et/ou des paramètres significatifs de ce fonctionnement exercé au cours des visites préventives ». Cette politique de maintenance se base sur l'évolution de paramètres qui décrivent l'état du système par exemple un jeu mécanique, une température ou tout autre paramètre qui puisse mettre en évidence l'imminence d'une défaillance. Tous les équipements sont concernés. Elle peut être appliquée à des équipements.

I.4 Classification des tâches de maintenance

Les tâches de maintenance sont classées en cinq niveaux. Ceux-ci font référence à la complexité des tâches à effectuer et, entre autres aux ressources matérielles nécessaires à la réalisation de chacune des tâches [6, 7] :

I.4.1 Tâches de maintenance du premier niveau :

Comporte des réglages simples prévus par le constructeur au moyen d'organes accessibles sans aucun démontage ; ou échange d'éléments accessibles en toute sécurité, sans avoir la nécessité à un outillage spécifique.

I.4.2 Tâches de maintenance de deuxième niveau

Comporte des opérations de dépannage par échange standard d'éléments prévus à cet effet ou opérations mineures de maintenance préventive qui peuvent nécessiter un outillage standard.

I.4.3 Tâches de maintenance de troisième niveau

Nécessite une identification et un diagnostic des pannes. La réparation s'effectue toujours par un échange de composants fonctionnels et les réparations mécaniques à réaliser sont mineures. L'outillage nécessaire est courant et prévu pour ce type d'intervention. De plus des appareils de mesure (banc d'essai, contrôle, etc.&) sont nécessaires pour la remise en route correcte de l'équipement qui a nécessité l'intervention.

I.4.4 Tâches de maintenance du quatrième niveau

Est celui des travaux importants de maintenance corrective ou préventive. Un outillage plus spécialisé est généralement requis tel que du matériel d'essai ou de test, un banc de contrôle, etc.

I.4.5 Tâches de maintenance du cinquième niveau

Comporte les travaux de rénovation, de reconstruction ou réparations importantes confiés à un atelier central. Les moyens nécessaires pour effectuer ce type d'intervention sont proches de ceux qui ont été utilisés lors de la fabrication de l'équipement par le constructeur. Après avoir défini l'activité du service de maintenance dans le milieu industriel, nous allons maintenant en détailler la composition avec notamment ses ressources.

I.5 Les opérations de maintenance [7]

I.5.1 Les opérations de maintenance corrective

I.5.1.1 Le dépannage

Action sur un bien en panne, en vue de le remettre en état de fonctionnement. Compte tenu de l'objectif, une action de dépannage peut s'accommoder de résultats provisoires (maintenance palliative) avec des conditions de réalisation hors règles de procédures, de coûts et de qualité, et dans ce cas sera suivie de la réparation. Le dépannage n'a pas de conditions d'applications particulières. La connaissance du comportement du matériel et des modes de dégradation est à la base d'un bon diagnostic et permettent souvent de gagner du temps.

I.5.1.2 La réparation

Intervention définitive et limitée de maintenance corrective après panne ou défaillance. L'application de la réparation peut être décidée soit immédiatement à la suite d'un incident ou d'une défaillance, soit après un dépannage, soit après une visite de maintenance préventive conditionnelle ou systématique. Remarque : la réparation correspond à une action définitive. L'équipement réparé doit assurer les performances pour lesquelles il a été conçu. Tous les équipements sont concernés.

I.5.2 Les opérations de maintenance préventive**I.5.2.1 Les inspections**

Activités de surveillance consistant à relever périodiquement des anomalies et exécuter des réglages simples ne nécessitant pas d'outillage spécifique, ni d'arrêt de l'outil de production ou des équipements.

I.5.2.2 Les visites

Opérations de surveillance qui, dans le cadre de la maintenance préventive systématique, s'opèrent selon une périodicité déterminée. Ces interventions correspondent à une liste d'opérations définies préalablement qui peuvent entraîner des démontages d'organes et une immobilisation du matériel. Une visite peut entraîner une action de maintenance corrective.

I.5.2.3 Les contrôles

Cette opération a pour objectif de vérifier des critères ou des données définis. Elle a pour base des références de vérification parfaitement établies.

I.5.2.4 Révision

Ensemble des actions d'exams, de contrôles et des interventions effectuées en vue d'assurer le bien contre toute défaillance majeure ou critique, pendant un temps ou pour un nombre d'unités d'usage donné. Il faut distinguer suivant l'étendue des opérations à effectuer les révisions partielles et les révisions générales. Dans les 2 cas, cette opération nécessite la dépose de différents sous-ensembles. Le terme révision ne doit en aucun cas être confondu avec les termes visites, contrôles, inspections. Les 2 types d'opérations définis (révision générale ou partielle) relèvent du 4ème niveau de maintenance.

I.6 Les stratégies et les décisions associées [8]

Les stratégies de maintenance permettent de définir des règles de décision et de déterminer le contexte informationnel qui détermine un espace de décision de maintenance.

Les questions Pourquoi, Quand, Quoi et Où sont les questions essentielles auxquelles une décision de maintenance doit répondre. En effet les questions Comment et Combien de temps, d'ordre opérationnel, sont caractéristiques de l'action envisagée et peuvent être données par le plan de maintenance. Toutefois les réponses peuvent être considérées comme des contraintes sur les actions envisagées. Le tableau I.1 synthétise les différents contextes de la décision à travers les réponses aux questions POURQUOI, QUAND, OÙ et QUOI.

	Pourquoi	Quand	Où	Quoi
Maintenance corrective	Corriger une défaillance	Corriger une défaillance	Composant défaillant	Dépannage ou Réparation
Maintenance préventive	Eviter une défaillance	Avant la défaillance qui est déterminée par le franchissement (présent ou futur) d'un seuil représentatif de la défaillance par un indicateur	Composant associé à un indicateur ou résultat d'un diagnostic	Défini par le plan de maintenance ou par l'expert
Maintenance systématique	Eviter une défaillance	Un indicateur d'usage	Composant associé	Défini par le plan de
Maintenance Conditionnelle	Eviter une défaillance en optimisant le risque d'action superflue	Un indicateur surveillé atteint un seuil représentatif d'un niveau de dégradation	Composant associé à l'indicateur surveillé ou résultat d'un	Défini par le plan de maintenance
Maintenance prévisionnelle	Eviter une défaillance en agissant au meilleur moment vis-à-vis de la dégradation et des performances du composant ou système	Un indicateur prédit de dégradation atteint un seuil	Composant associé à l'indicateur prédit ou résultat d'un diagnostic	Issue d'un processus d'aide à la décision rationalisant les performances du composant ou système

Tableau I. 1 : Les différents contextes de la décision à travers les réponses aux questions pourquoi, quand, où et quoi

I.7 Autre formes et méthodes de maintenance

I.7.1 La maintenance améliorative [8] :

L'amélioration des biens d'équipements qui consiste à procéder à des modifications, des changements, des transformations sur un matériel correspond à la maintenance d'amélioration. Dans ce domaine beaucoup de choses restent à faire. C'est un état d'esprit qui nécessite une attitude créative. Cette créativité impose la critique. Cependant, pour toute maintenance d'amélioration une étude économique sérieuse s'impose pour s'assurer de la rentabilité du projet. Les améliorations à apporter peuvent avoir comme objectif l'augmentation des performances de production du matériel ; l'augmentation de la fiabilité, c'est-à-dire diminuer les fréquences d'interventions ; l'amélioration de la maintenabilité (amélioration de l'accessibilité des sous-systèmes et des éléments à haut risque de défaillance) ; la standardisation de certains éléments pour avoir une politique plus cohérente et améliorer les actions de maintenance, l'augmentation de la sécurité du personnel.

Tous les matériels sont concernés à condition que la rentabilité soit vérifiée. Cependant une petite restriction pour les matériels à renouveler dont l'état est proche de la réforme, pour usure généralisée ou par obsolescence technique.

I.7.2 La total productive maintenance (T.P.M) : [9]

La stratégie TPM est un élargissement de la mission de la maintenance à la totalité du personnel, c'est-à-dire impliquant la participation de tous les services notamment ingénierie, implication de toutes les productions, maintenance et tous les niveaux hiérarchiques, des dirigeants aux services opérateurs. Elle s'inscrit dans les stratégies fondamentales du juste à temps (zéro panne, zéro stock, zéro délai, etc.).

Les fonctions et responsabilités envers les équipements sont partagées, en procédant comme suit :

- Les opérateurs sont chargés de tâches de maintenance du 1er niveau (nettoyage, lubrification, examen externe, etc.). Ils ont la responsabilité de leur équipement.
- Le service maintenance intervient comme spécialiste pour des tâches plus complexes.
- La TPM fait participer des petits groupes analogiques aux cercles de qualités ayant pour objectif l'amélioration de la maintenance dans l'intérêt de l'entreprise.

Les objectifs de TPM sont :

- Réduction du délai de mise au point des équipements.
- Augmentation de la disponibilité, et du taux de rendement synthétique (T.R.S.).
- Augmentation de la durée de vie des équipements.
- Participation des utilisateurs à la maintenance appuyée par des spécialistes de maintenance.
- Pratique de la maintenance préventive systématique et conditionnelle.
- Meilleure maintenabilité des équipements (envisagée à la conception, aide au diagnostic, systèmes experts).

Conclusion :

Il est essentiel de porter une attention particulière à l'élaboration d'une stratégie de maintenance adaptée aux activités spécifiques de l'entreprise et à ses perspectives de développement. Cela permet de satisfaire les exigences des consommateurs tout en maximisant la rentabilité.

Le chapitre que nous venons de conclure a mis en lumière divers aspects de la maintenance industrielle, notamment les différentes approches et formes de la maintenance et leurs impacts sur la performance des entreprises. La maintenance, étant un secteur en constante évolution, ne peut se conformer à une pratique unique et universelle. L'objectif principal d'une politique de maintenance est d'optimiser la disponibilité des équipements de production tout en minimisant les coûts associés

Chapitre II :

L'analyse FMD Par les lois de Probabilité

Introduction :

La Fiabilité, Maintenabilité et Disponibilité (FMD) sont des éléments clés dans l'évaluation de la performance des systèmes industriels. Ce chapitre se concentre sur l'application des lois de probabilités pour analyser et prédire le comportement de ces systèmes en termes de fiabilité, de maintenabilité et de disponibilité. Ce chapitre se concentre sur l'analyse de FMD en utilisant les lois de probabilité. Il définit les concepts clés de l'analyse FMD, présente les indicateurs opérationnels de FMD, explore les expressions mathématiques de la fiabilité, détaille les principales lois de probabilité utilisées dans l'analyse FMD, et applique ces lois à des modèles spécifiques tels que les modèles exponentiels, de Weibull et log-normal. De plus, ce chapitre aborde aussi l'approche mathématique de la disponibilité et le test de Kolmogorov-Smirnov pour la validation des modèles.

II.1 Définitions [10] :

II.1.1 La fiabilité :

La fiabilité (Reliability en anglais) est l'aptitude d'une entité à accomplir les fonctions requises dans des conditions données pendant une durée donnée. Elle est caractérisée par la probabilité $R(t)$ que l'entité E accomplisse ces fonctions, dans les conditions données pendant l'intervalle de temps $[0, t]$, sachant que l'entité n'est pas en panne à l'instant 0.

$$R(t) = P [E \text{ non défectueuse sur } [0, t]] \quad (II.1)$$

II.1.2 La maintenabilité :

La maintenabilité (Maintainability en anglais) est l'aptitude d'une entité à être maintenue ou rétablie dans un état dans lequel elle peut accomplir une fonction requise, lorsque la maintenance est réalisée dans des conditions données avec des procédures et des moyens prescrits. Elle est caractérisée par la probabilité $M(t)$ que l'entité E soit en état, à l'instant t , d'accomplir ses fonctions, sachant que l'entité était en panne à l'instant 0.

$$M(t) = P [E \text{ est réparée sur } [0, t]] \quad (II.2)$$

II.1.3 La disponibilité :

La disponibilité (Availability en anglais) est l'aptitude d'une entité à être en état d'accomplir les fonctions requises dans les conditions données et à un instant donné. Elle est caractérisée par la probabilité $D(t)$ que l'entité E soit en état, à l'instant t , d'accomplir les fonctions requises dans des conditions données.

$$D(t) = P [E \text{ non défaillante à l'instant } t] \quad (II.3)$$

II.2 Indicateurs opérationnels d'FMD [11] :

La figure I.1 schématise les états successifs que peut prendre un système réparable.

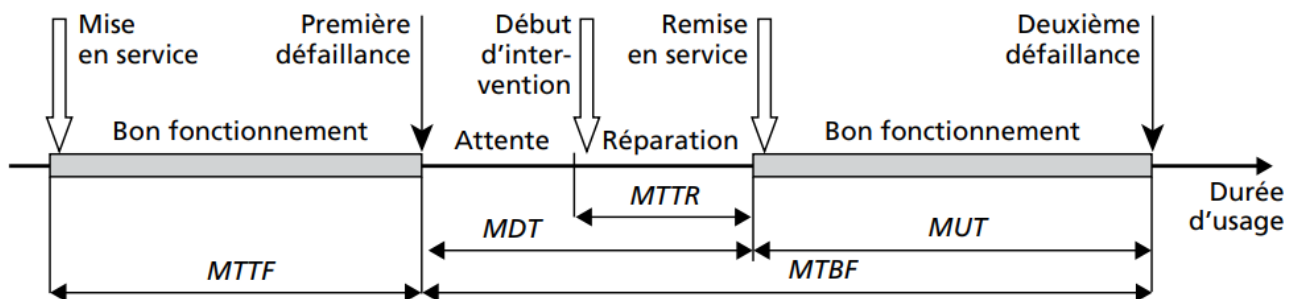


Fig II. 1 : Les durées caractéristique de FMD. [6]

Les grandeurs portées par le graphe sont des durées (TBF) auxquelles on fait correspondre des moyennes (MTBF) obtenues par exploitation statistique $m(t)$ ou probabiliste $E(t)$ des n durées constatées et enregistrées. Les sigles utilisés sont d'origine anglo-saxonne et correspondent aux notions suivantes :

- MTTF (mean time to [first] failure) : temps moyen avant-première défaillance.
- MTBF (mean time between failure) : temps moyen entre deux défaillances successives ;
- MDT ou MTI (mean down time) : temps moyen d'indisponibilité ou temps moyen d'arrêt propre.
- MUT (mean up time) : temps moyen de disponibilité.
- MTTR (mean time to repair) : temps moyen de réparation.

II.2.1 Indicateurs de fiabilité

Ils sont classés ici du plus approximatif vers le plus rigoureux.

- **N** : nombre de pannes.
- **MTTF** : moyenne des temps de bon fonctionnement jusqu'à la première défaillance, dont la date d'arrivée est peu significative de la suite.
- **MTBF** : calculée à partir d'une moyenne statistique d'un échantillon de n durées TBF.
- **R(t)** : fonction fiabilité (R vient de l'anglais reliability). C'est la fiabilité « stricte » définie comme la probabilité de bon fonctionnement d'un système à l'instant t, déterminée comme précédemment à partir d'un modèle probabiliste ajusté à un échantillon.
- **MTBF** calculée à partir de l'espérance mathématique de la variable aléatoire TBF en utilisant une loi de probabilité R(t) ajustée à un échantillon de n valeurs de TBF relevées.
- **$\lambda(t)$** : fonction taux de défaillance. Déduite de la fonction R(t), c'est un bon indicateur du comportement d'un système.

II.2.2 Indicateurs de maintenabilité

MTTR est l'indicateur de maintenabilité. Elle peut être obtenue par la moyenne statistique d'un échantillon de n valeurs TTR ou par l'espérance mathématique de la variable TTR obtenue à partir d'un modèle probabiliste.

Notons que la durée d'intervention TTR n'est pas une donnée facile à acquérir.

M(t) est la maintenabilité stricte. C'est la probabilité associée à un instant T, d'une remise en état de fonctionnement. Elle est déterminée à partir d'un modèle probabiliste.

II.2.3 Indicateurs de disponibilité

L'indicateur de base de la disponibilité opérationnelle est :

$$D_{op} = \frac{MTBF}{MRBT + MTTR} < 1 \quad (II.4)$$

Mais bien d'autres indicateurs peuvent être imaginés, plus ou moins rigoureux et choisis en fonction des données existantes. Quelques exemples :

- MTBF-MTTR est un indicateur de disponibilité propre, MUT est un indicateur de disponibilité effective ;
- MTI est un indicateur d'indisponibilité propre.

II.3 Expressions mathématiques de la fiabilité [11]

II.3.1 Fonctions de distribution et de répartition

II.3.1.1 Notion de variable aléatoire

On appelle variable aléatoire X une variable telle qu'à chaque valeur x de X on puisse associer une probabilité $F(x)$. Une variable aléatoire peut être :

- Continue : intervalle de temps entre deux défaillances consécutives ;
- Discrète : nombre de défaillances d'un composant sur un intervalle de temps.

II.3.1.2 Cas général d'une variable aléatoire continue

Soit une loi de probabilité relative à une variable aléatoire continue t . Elle est caractérisée par sa fonction de distribution, ou densité de probabilité $f(t)$ et par sa fonction de répartition $F(t)$ telles que :

$$f(t) = \frac{dF(t)}{dt} = \lim_{dt \rightarrow \infty} \frac{\text{Prob}(t < T < t + dt)}{dt} \quad (II.5)$$

La fonction de répartition $F(t)$ représente la probabilité qu'un événement (une défaillance) survienne à l'instant T dans l'intervalle $(0, t)$. $F(t) = P(T < t)$.

II.3.1.3 Représentations graphiques de $f(t)$ et $F(t)$

$$f(t)dt = \text{Prob}[t < T < t + dt] \quad (II.6)$$

$f(t)$ est une densité de probabilité.

$$F(t) = \int_{-\infty}^{t_i} f(t)dt = \text{Prob}(T < t_i) \quad (II.7)$$

Si variable aléatoire discrète :

$$F(t_i) = \sum_0^n f(t_i) = \text{Prob}(T < t_i) \quad (II.8)$$

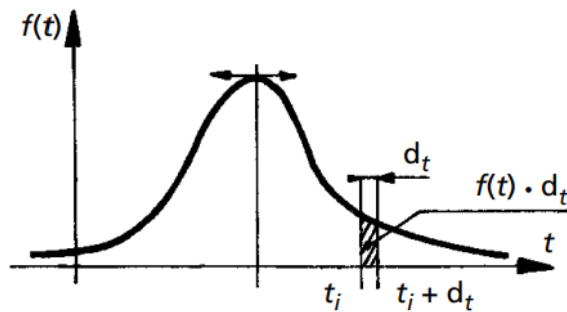


Fig II. 2 : Fonction de distribution. [11]

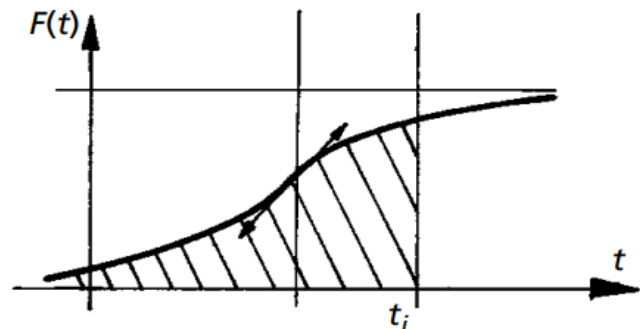


Fig II. 3 : Fonction de répartition. [11]

II.3.2 Taux de défaillance

$\lambda(t)dt$ est une probabilité conditionnelle de défaillance sur l'intervalle $(t, t + dt)$, car elle ne s'applique qu'aux survivants à l'instant t

Appliquons le théorème des probabilités conditionnelles :

$$P(\text{d'avoir une panne entre } t \text{ et } dt) = P(\text{de survivre à } t) \lambda(t) dt \quad (II.9)$$

$$f(t) \cdot dt = R(t) \cdot \lambda(t) \cdot dt \quad (II.10)$$

$$\lambda(t) = \frac{f(t)}{R(t)} \quad (II.11)$$

II.4 Les principales lois [10]

Nous présentons dans cette section les lois et les modèles de FMD susceptibles, selon l'expérience, de représenter des distributions de durée de vie qui interviennent le plus fréquemment dans l'analyse de la fiabilité mécatronique. Nous rappelons les principales propriétés de ces lois, les fonctions de fiabilité associées, les densités de probabilité ainsi que les taux de défaillance.

II.4.1 Loi exponentielle

Cette loi a de nombreuses applications dans plusieurs domaines. Elle décrit la vie des matériels qui subissent des défaillances brutales. La loi exponentielle est la plus couramment utilisée en fiabilité électronique pour décrire la période durant laquelle le taux de défaillance des équipements est

considéré comme constant (défaillance aléatoire). Elle décrit le temps écoulé jusqu'à une défaillance, ou l'intervalle de temps entre deux défaillances.

Elle est définie par un seul paramètre, le taux de défaillance, λ .

Elle est caractérisée par :

- La fonction de répartition

$$F(t) = 1 - e^{-\lambda t} \quad (II.12)$$

- La densité de probabilité

$$f(t) = \lambda e^{-\lambda t} \quad (II.13)$$

- Le taux de défaillance

$$\lambda(t) = \lambda \quad (II.14)$$

II.4.2 La loi de Weibull

La loi de Weibull, est souvent utilisée en mécanique ; elle caractérise bien le comportement du produit dans les trois phases de vie selon la valeur du paramètre de forme β : période de jeunesse ($\beta < 1$), période de vie utile ($\beta = 1$) et période d'usure ou vieillissement ($\beta > 1$). La loi de Weibull est définie par trois paramètres β et η , γ .

- La fonction de répartition

$$F(t) = 1 - e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \quad (II.15)$$

- La densité de probabilité

$$f(t) = \frac{\beta}{\eta} \left(\frac{t-\gamma}{\eta}\right)^{\beta-1} e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \quad (II.16)$$

- Le taux de défaillance

$$\lambda(t) = \frac{\beta}{\eta} \left(\frac{t-\gamma}{\eta}\right)^{\beta-1} \quad (II.17)$$

II.4.3 Loi log-normale :

Une variable aléatoire continue et positive t est distribuée selon une loi lognormale si son logarithme est distribué suivant une loi normale. Cette distribution est utilisée en fiabilité pour modéliser les défaillances par fatigue. La loi lognormale a deux paramètres μ et σ :

- La fonction de répartition

$$F(t) = \Phi\left(\frac{\log(t) - \mu}{\sigma}\right) \quad (II.18)$$

- La densité de probabilité

$$f(t) = \frac{1}{t\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2} \quad (II.19)$$

- Le taux de défaillance

$$\lambda(t) = \frac{e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2}}{t \int_0^{\infty} \sigma\sqrt{2\pi}f(t)dt} \quad (II.20)$$

I.5 Analyse de la fiabilité et la maintenabilité à partir du modèle exponentiel [11]

La courbe en baignoire, relative à la durée de vie d'un système, fait apparaître une période de maturité plus ou moins longue pendant laquelle le taux de défaillance d'un système ou d'un composant est sensiblement constant :

C'est le champ d'application de la loi exponentielle, qui repose sur l'hypothèse $\lambda = \text{constante}$.

Dans ce cas, les défaillances « aléatoires » émergent sous l'action de causes diverses et indépendantes par opposition à la période suivante, dite « de vieillesse », pendant laquelle un ou plusieurs modes de défaillance prédominants apparaissent (fatigue, corrosion, etc.).

L'électronique se prête bien à l'utilisation de la loi exponentielle, dès lors que les composants sont déterminés. La plupart des analyses prévisionnelles de fiabilité et de maintenabilité reposent sur les hypothèses exponentielles :

λ taux de défaillance constant en fiabilité, donc $MTBF = 1/\lambda$.

μ taux de réparation constant en maintenabilité, donc $MTTR = 1/\mu$.

II.5.1 Caractéristiques

II.5.1.1 Pour la fiabilité

Elle se retrouve à partir de la loi fondamentale de la fiabilité avec λ constant. La probabilité de survie entre 0 et t devient :

$$R(t) = e^{-\lambda t} \quad (II.21)$$

- Densité de probabilité :

$$f(t) = \lambda e^{-\lambda t} \quad (II.22)$$

- Fonction de répartition :

$$F(t) = 1 - R(t) = 1 - e^{-\lambda t} \text{ probabilité de panne entre 0 et t} \quad (II.23)$$

- Espérance mathématique :

$$E(t) = 1/\lambda = MTBF \quad (II.24)$$

- Écart-type :

$$\sigma = \frac{1}{\lambda} \quad (II.25)$$

II.5.1.2 Pour la maintenabilité

Elle se retrouve à partir de la loi fondamentale de la fiabilité avec constant. La probabilité de survie entre 0 et t devient :

$$M(t) = 1 - e^{-\mu t} \quad (II.26)$$

- Densité de probabilité :

$$g(t) = \mu e^{-\mu t} \quad (II.27)$$

- Espérance mathématique :

$$E(t) = \frac{1}{\mu} = MTTR \quad (II.28)$$

- Écart-type :

$$\sigma = \frac{1}{\mu} \quad (II.29)$$

II.5.2 Représentations graphiques de la loi exponentielle

La représentation linéaire de la loi exponentielle s'obtient sur un simple papier semi-logarithmique. L'anamorphose permet une représentation linéaire de la fiabilité, la droite ayant pour pente $\lambda/2,3$. En effet, si $R(t) = e^{-\lambda t}$

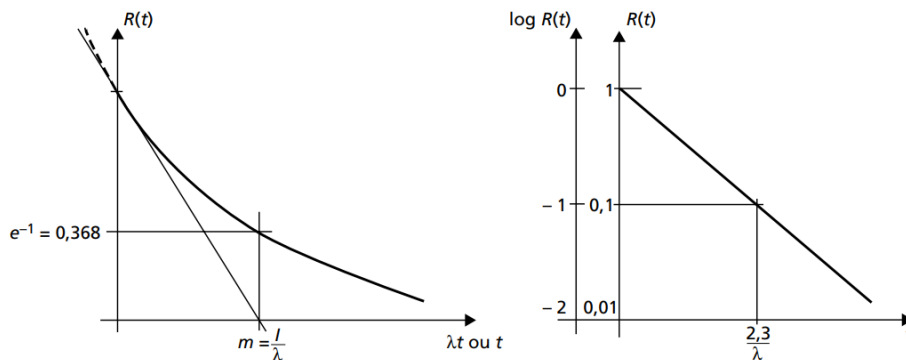


Fig II. 4 : Représentation graphique de la loi exponentielle. [11]

II.6 Analyse de la fiabilité et la maintenabilité à partir du modèle de Weibull [11]

La loi de Weibull à trois paramètres est très « souple », ce qui lui permet de s'ajuster à un grand nombre d'échantillons prélevés au long de la vie d'un équipement. Elle couvre les cas de taux de défaillance variables, décroissants (période de jeunesse) ou croissants (période de vieillesse).

Son exploitation fournit :

- Une estimation de la MTBF de la population de défaillances et la MTTR ;
- Les équations de $R(t)$ et de $\lambda(t), M(t), \mu(t)$ ainsi que leurs variations sous forme graphique ;
- Mais aussi le paramètre de forme β qui peut orienter un diagnostic, sa valeur étant caractéristique de certains modes de défaillance.

Les graphes de la fonction de distribution $f(t)$ et du taux de défaillance $\lambda(t)$ de la figure 5.21 illustrent ce polymorphisme sous la seule influence du paramètre de forme β

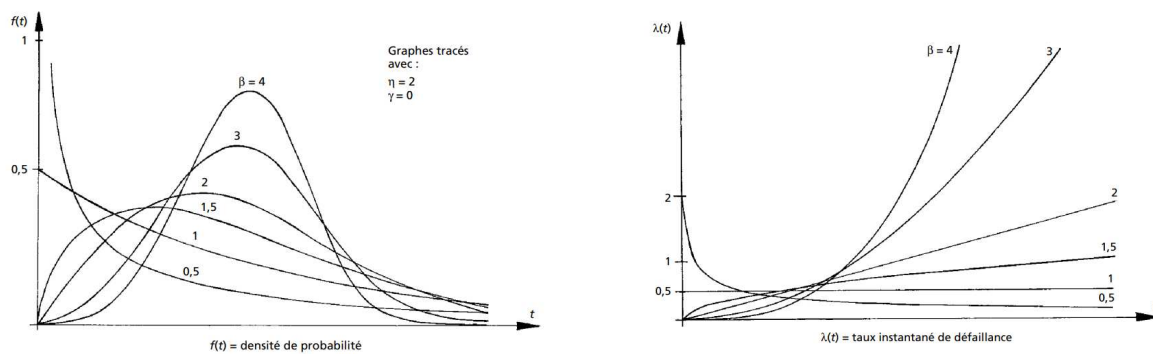


Fig II. 5 : variabilité de la loi de weibull sous l'influence de β [11]

β est appelé paramètre de forme ($\beta > 0$) : suivant que β est supérieur, égal ou inférieur à 1, la loi de Weibull correspond à un taux de défaillance instantané croissant, constant ou décroissant.

η est appelé paramètre d'échelle ($\eta > 0$) : parfois nommé « caractéristique de vie », c'est un simple paramètre de changement d'échelle de temps.

γ est appelé paramètre de position ($-\infty < \gamma < +\infty$) : il définit un changement d'origine dans l'échelle des temps. Par exemple, si $\gamma > 0$, il y a une survie totale (pas de défaillance) entre 0 et γ .

II.6.1 Expressions mathématiques caractéristiques

II.6.1.1 Pour la fiabilité

La densité de probabilité

$$f(t) = \frac{\beta}{\eta} \left(\frac{t - \gamma}{\eta} \right)^{\beta-1} e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \text{ avec } t \geq \gamma \quad (II.30)$$

Fonction de répartition $F(t)$ et fiabilité $R(t)$

$$F(t) = 1 - e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \text{ et } R(t) = e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \quad (II.31)$$

Expression du taux instantané de défaillance $\lambda(t)$

$$\lambda(t) = \frac{\beta}{\eta} \left(\frac{t - \gamma}{\eta} \right)^{\beta-1} \text{ avec } t \geq \gamma, \beta > 0, \eta > 0 \quad (II.32)$$

Espérance mathématique (MTBF) et variance :

$$MTBF = A\eta + \gamma \quad (II.33)$$

$$\sigma = B\eta \quad (II.34)$$

II.6.1.2 Pour la maintenabilité

La densité de probabilité

$$g(t) = \frac{\beta}{\eta} \left(\frac{t-\gamma}{\eta} \right)^{\beta-1} e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \text{ avec } t \geq \gamma \quad (II.35)$$

Fonction de la maintenabilité

$$M(t) = 1 - e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \quad (II.36)$$

Expression du taux instantané de réparation $\mu(t)$

$$\mu(t) = \frac{\beta}{\eta} \left(\frac{t-\gamma}{\eta} \right)^{\beta-1} \text{ avec } t \geq \gamma, \beta > 0, \eta > 0 \quad (II.37)$$

Espérance mathématique (MTTR) et variance :

$$MTTR = A\eta + \gamma \quad (II.38)$$

$$\sigma = B\eta \quad (II.39)$$

II.6.2 Structure du papier de Weibull

II.6.2.1 Description du papier fonctionnel de Weibull

Ce papier « log-log » porte quatre axes :

- L'axe A est l'axe des temps sur lequel nous porterons les valeurs t_i de durées de bon fonctionnement.
- L'axe B porte $F(t)$ sur lequel nous porterons les valeurs $F(i)$ calculées par approximation (rangs moyens ou médians). Nous estimerons la fiabilité en prenant le complément : $R(t) = 1 - F(t)$.
- L'axe a correspond à $\ln(t)$.
- L'axe b correspond à $\ln \ln (1/1 - F(t))$. Cet axe permettra d'évaluer la valeur de β .

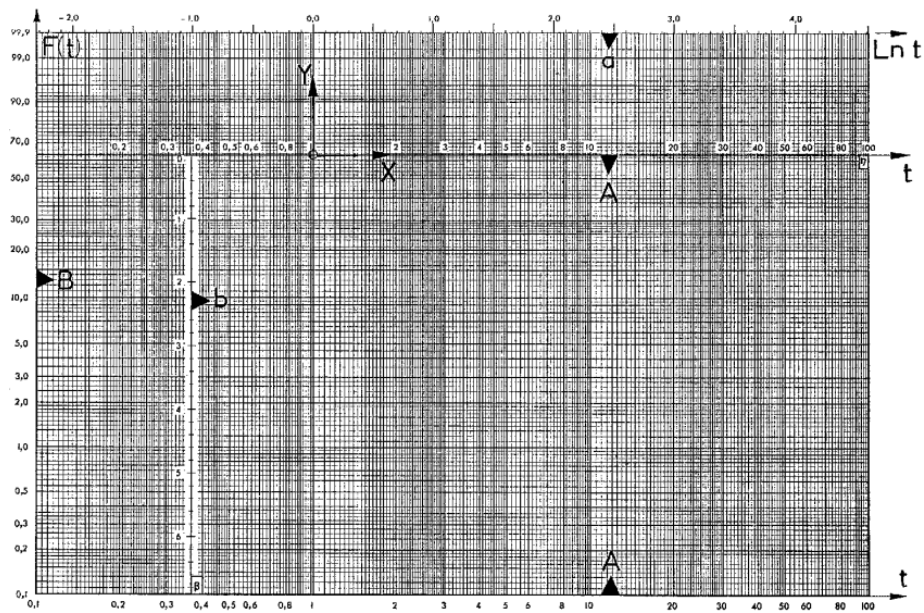


Fig II. 6 : Description du papier Weibull. [11]

II.6.2.2 Justification mathématique de la structure du papier de Weibull

$$F(t) = 1 - e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \quad (II.40)$$

$$1 - F(t) = e^{-\left(\frac{t-\gamma}{\eta}\right)^\beta} \quad (II.41)$$

Nous pouvons prendre le log népérien des deux membres car $R(t) \leq 1$

$$\frac{1}{R(t)} = \frac{1}{1 - F(t)} \geq 1 \quad (II.42)$$

$$\ln\left(\frac{1}{1 - F(t)}\right) \geq 0 \quad (II.43)$$

L'équation devient :

$$\ln\left(\frac{1}{1 - F(t)}\right) = \left(\frac{t-\gamma}{\eta}\right)^\beta \quad (II.44)$$

$$\ln\left[\ln\left(\frac{1}{1 - F(t)}\right)\right] = \beta \ln\left(\frac{t-\gamma}{\eta}\right) \quad (II.45)$$

$$\ln \left[\ln \left(\frac{1}{1 - F(t)} \right) \right] = \beta \ln(t - \gamma) - \beta \ln \eta \quad (II.46)$$

Qui est de la forme :

$$Y = \beta X + C \quad (II.47)$$

Nous obtenons une relation linéaire entre X et Y telle que la droite D1 de régression du nuage de points (F_i, t_i) a pour pente le paramètre β .

Au point $t = \eta = 1$, origine du repère (X, Y), $\ln \eta = 0$, donc $C = 0$.

La droite D2 parallèle à D1 passant par l'origine de (X, Y) a pour équation $Y = \beta X$. Elle permet de déterminer la valeur de β sur l'axe b.

II.6.3 Méthodologie de l'analyse de fiabilité

II.6.3.1 Préparation des données

Nous devons faire un tableau mis en évidence la structure générale d'un tableau de données en fonction de la taille N de l'échantillon. Par exemple les TBFs (durées de bon fonctionnement d'un composant) ou TTRs traité avec la formule des rangs médians.

Les N la taille de l'échantillon de durées de bon fonctionnement TBF ou TTR obtenues en exploitation seront classées par valeurs croissantes selon un ordre i allant de $i = 1$ à $i = N$. L'ordre $i = n(t)$ représente la i ème défaillance de l'échantillon. $F(i)$ est la probabilité cumulative de défaillance sur $(0, t)$. Suivant la taille N de l'échantillon.

- Si $N > 50$:

$$F(i) = \frac{i}{N} \quad (II.48)$$

- Si $50 > N > 20$, formule des rangs moyens :

$$F(i) = \frac{i}{N + 1} \quad (II.49)$$

- Si $20 > N$, formule des rangs médians :

$$F(i) = \frac{i - 0.3}{N + 0.4} \quad (II.50)$$

II.6.3.2 Tracé du nuage de points (ti, Fi) et la droite dite « de Weibull » D1

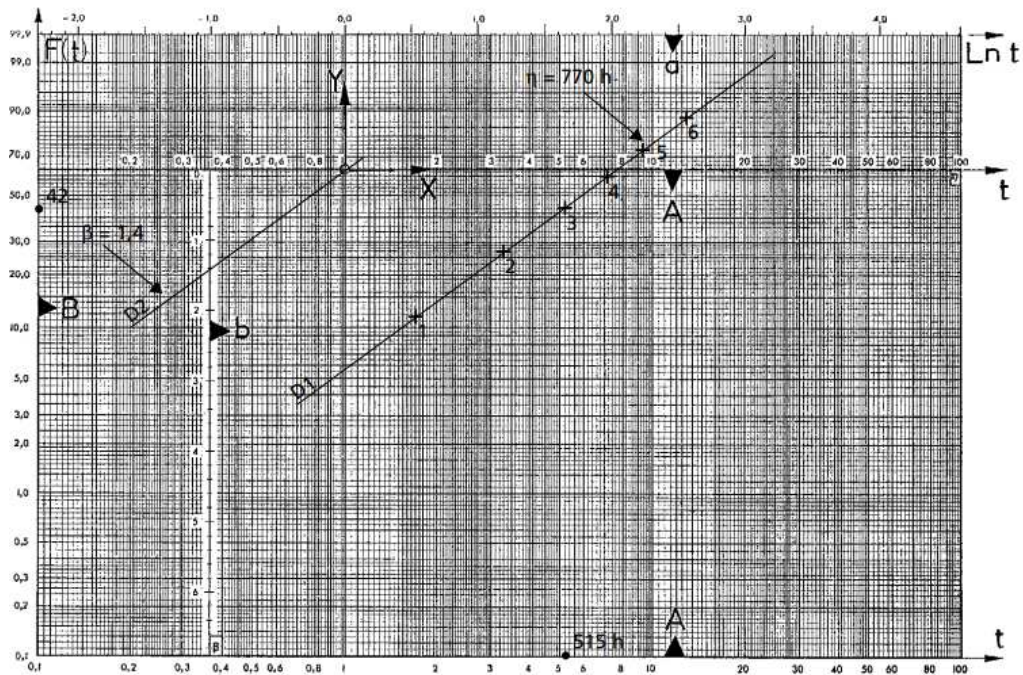


Fig II. 7 : Exemple du traçage du nuage de points et du droit de weibull. [11]

II.6.3.3 Détermination des valeurs des trois paramètres β , η , γ

La droite D1 de régression coupe l'axe A (t, η) à l'abscisse $t = \eta$ (justification : quand $Y = 0, \ln(t - \gamma) = \ln t = \ln \eta$ donc $t = \eta$). β est la pente de D1. Pour obtenir sa valeur, nous traçons la droite D2 parallèle à D1, passant par le point $\eta = 1$ (origine de X, Y). La droite D2 coupe l'axe b en un point qui donne la valeur de β (justification : D2 est la droite d'équation $Y = \beta X$ avec $X = \ln t$).

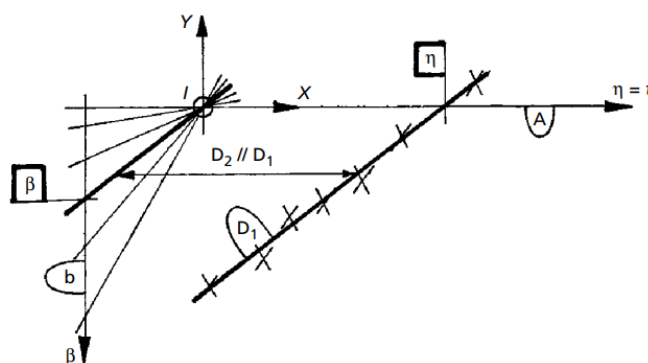


Fig II. 8 : La Détermination des valeurs des trois paramètres β, η, γ [11]

Pour $t = 1$, $\ln t = 0$ donc $X = Y = 0$. Pour $\ln t = -1$ (sur l'échelle a), $Y = \beta \ln t = -\beta$. Comme l'axe b portant β est orienté vers le bas, $Y = -b$ donc $b = \beta$.

Pour la détermination de γ : les techniques de redressement

Si le nuage des points (t_i, F_i) fait apparaître une courbure convexe ou concave, alors le paramètre γ a une valeur non nulle à déterminer après avoir effectué un « redressement ». Ce redressement peut se faire « au jugé » ou par application d'une formule de redressement

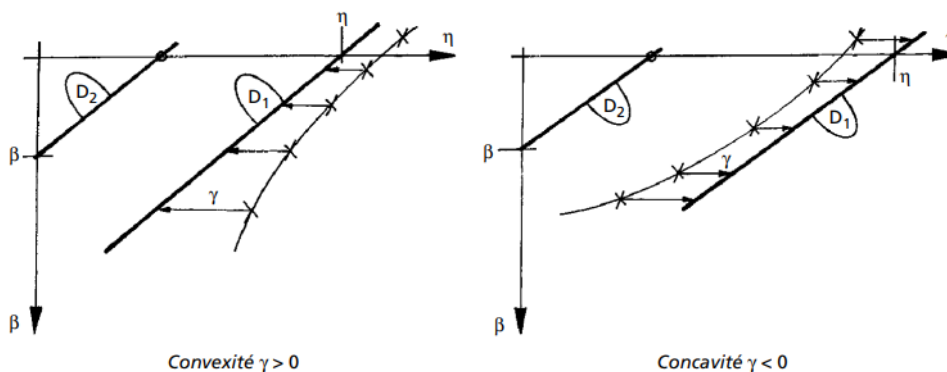


Fig II. 9 : Les techniques de redressement. [11]

Redressement au jugé Il faut translater tous les points d'une même valeur γ dont on augmente la valeur de proche en proche : la concavité (ou convexité) diminue progressivement jusqu'à l'obtention d'une droite : la valeur de translation finale est γ

$$\gamma = \frac{t_2^2 - t_1 t_3}{2t_2 - t_3 - t_3} \quad (II.50)$$

Prenons trois points A1, A2 et A3 sur la courbe C1 tels que $(a_1, a_2) = (a_2, a_3) = \Delta$. Il est conseillé de prendre des points espacés mais non extrêmes. Nous lisons les valeurs t_1, t_2 et t_3 sur l'axe des t (axe A). Il reste à appliquer la formule ci-dessus pour trouver γ et pour tracer la droite D1. L'intersection de D1 avec l'axe η donne la valeur du paramètre η

Puis L'équation de la loi de Weibull (et représentation graphique éventuelle). Détermination de la MTBF ou MTTR et σ .

II.7 Analyse de la fiabilité à partir du modèle de log-normal

La modélisation de la fiabilité à l'aide de la loi log-normale est une méthode statistique utilisée pour prédire la durabilité et la fiabilité des produits et des systèmes. La loi log-normale est particulièrement utile pour modéliser des données qui sont asymétriques vers la droite, ce qui est souvent le cas dans les études de fiabilité où l'on s'intéresse à la durée de vie des composants ou des systèmes [12].

II.7.1 Expressions mathématiques caractéristiques

II.7.1.1 Pour la fiabilité

- La fonction de la fiabilité

$$R(t) = 1 - \Phi\left(\frac{\log(t) - \mu}{\sigma}\right) \quad (II.51)$$

- La fonction de répartition

$$F(t) = \Phi\left(\frac{\log(t) - \mu}{\sigma}\right) \quad (II.52)$$

- La densité de probabilité

$$f(t) = \frac{1}{t\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2} \quad (II.53)$$

- Le taux de défaillance

$$\lambda(t) = \frac{e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2}}{t \int_0^{\infty} \sigma\sqrt{2\pi}f(t)dt} \quad (II.54)$$

- Espérance mathématique

$$MTBF = e^{\mu + \frac{\sigma^2}{2}} \quad (II.55)$$

II.7.1.2 Pour la maintenabilité

- La fonction de la maintenabilité

$$M(t) = \Phi\left(\frac{\log(t) - \mu}{\sigma}\right) \quad (II.56)$$

- La densité de probabilité

$$g(t) = \frac{1}{t\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2} \quad (II.57)$$

- Le taux de défaillance

$$\mu(t) = \frac{e^{-\frac{1}{2}\left(\frac{\log(t)-\mu}{\sigma}\right)^2}}{t \int_0^{\infty} \sigma \sqrt{2\pi} f(t) dt} \quad (II.58)$$

- Espérance mathématique

$$MTTR = e^{\mu + \frac{\sigma^2}{2}} \quad (II.59)$$

Avec les paramètres de signification

σ : l'écart type

μ : c'est la moyenne

II.7.2 Représentation graphique :

Voici des exemples des courbes de la fonction de densité et la répartition pour une loi log-normale avec la variation de μ et σ

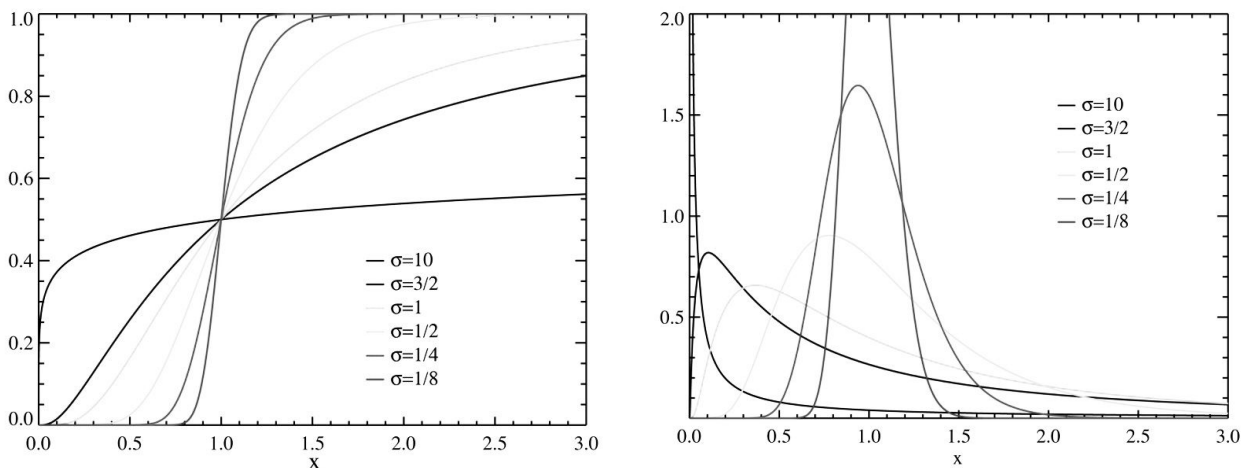


Fig II. 10 : variabilité de la loi de Log-normal sous l'influence de σ [12].

II.8 Approche mathématique de la disponibilité [11]

Modélisation de la disponibilité instantanée. Formule générale Nous nous plaçons dans l'hypothèse exponentielle, avec les deux taux λ et μ supposés constants et indépendants du temps :

Taux de défaillance :

$$\lambda = \frac{1}{MTBF} \quad (II.60)$$

Taux de réparation :

$$\mu = \frac{1}{MTTR} \quad (II.61)$$

La disponibilité instantanée d'un système réparable est de la forme :

$$D(t) = \frac{\mu}{\mu + \lambda} + \frac{\lambda}{\mu + \lambda} e^{-(\lambda + \mu)t} \quad (II.62)$$

Expressions de la disponibilité asymptotique :

$D(t)$ tend vers une limite asymptotique D_{∞} quand $t \rightarrow \infty$. Nous trouvons alors les formules usuelles équivalentes :

$$D_{\infty} = \frac{\mu}{\mu + \lambda} \text{ ou } \frac{MTBF}{MTBF + MTTR} \quad (II.63)$$

II.9 Test Kolmogorov Smirnov : [13]

En statistiques, le test de Kolmogorov-Smirnov est un test d'hypothèse utilisé pour déterminer si un échantillon suit bien une loi donnée connue par sa fonction de répartition continue, ou bien si deux échantillons suivent la même loi.

Avant la validation de toutes les lois de fiabilité, il est nécessaire de tester l'hypothèse pour savoir si nous devons accepter ou rejeter le modèle proposé par le test de K-S avec un seuil de $\alpha = 5\%$. Ce test consiste à calculer l'écart entre la fonction de répartition théorique $F(t)$ et la fonction de répartition réelle $F_n(t)$ et prendre le maximum en valeur absolue D_n . Cette valeur est comparée avec $D_{n, \alpha}$ qui est donnée par la table de Kolmogorov Smirnov. Dans le cas où $D_n < D_{n, \alpha}$, on accepte l'hypothèse et si $D_n > D_{n, \alpha}$, on refuse l'hypothèse.

N	alpha 0.001	alpha 0.01	alpha 0.02	alpha 0.05	alpha 0.1	alpha 0.15	alpha 0.2
1		0.99500	0.99000	0.97500	0.95000	0.92500	0.90000
2	0.97764	0.92930	0.90000	0.84189	0.77639	0.72614	0.68377
3	0.92063	0.82900	0.78456	0.70760	0.63604	0.59582	0.56481
4	0.85046	0.73421	0.68887	0.62394	0.56522	0.52476	0.49265
.
.
.
45	0.28482	0.23780	0.22621	0.19842	0.17881	0.16626	0.15673
50	0.27051	0.22585	0.21460	0.18845	0.16982	0.15790	0.14886

Tableau II. 1 : L'hypothèse de test Kolmogorov-Smirnov

Conclusion :

L'analyse FMD se révèle être un outil indispensable pour évaluer la performance des systèmes, des équipements et des infrastructures. En combinant les concepts de fiabilité, de maintenabilité et de disponibilité, elle nous offre une vision holistique des performances des actifs et permet de prendre des décisions éclairées pour optimiser leur gestion.

Les lois de probabilité, telles que les lois exponentielles, de Weibull et log-normale, jouent un rôle crucial dans cette démarche. Elles nous permettent de modéliser les temps de défaillance et d'estimer les taux de panne, éléments essentiels pour comprendre les caractéristiques des défaillances et pour planifier des stratégies de maintenance efficaces.

L'objectif ultime de l'analyse FMD est d'accroître l'efficacité opérationnelle, de minimiser les coûts de maintenance et de maximiser l'utilisation des actifs. En exploitant la puissance de la programmation Python et de ses bibliothèques spécialisées, nous pouvons transformer les données brutes en informations précieuses, présentées sous forme de tableaux et de graphiques clairs et concis, qui guident nos décisions stratégiques.

En conclusion, l'analyse FMD, couplée aux lois de probabilité et aux outils de programmation, s'impose comme un pilier essentiel pour une gestion optimisée des actifs. Elle permet aux entreprises de prendre des décisions éclairées qui contribuent à améliorer la performance, la rentabilité et la longévité de leurs équipements.

Chapitre III :

La réalisation du Programme

Introduction :

Les chapitres précédents ont présenté une introduction générale à l'analyse de la fiabilité, de la maintenance et de la disponibilité (FMD). Ce chapitre se concentre sur la réalisation concrète du programme, en abordant les étapes clés du développement, le choix du langage de programmation et les bibliothèques utilisées pour la réalisation de ce programme que je nomme « 317_FMD ».

III.1 Choix du Langage de Programmation : Python

Le choix de Python comme langage de programmation pour le développement de ce programme s'appuie sur plusieurs avantages majeurs [14] :

III.1.1 Polyvalence et puissance :

Python est un langage de programmation généraliste largement utilisé dans divers domaines, y compris le calcul scientifique, l'analyse de données et le développement de logiciels. Sa polyvalence permet de traiter efficacement les tâches de manipulation de données, de calcul statistique et de visualisation graphique requises pour la modélisation FMD.

III.1.2 Bibliothèques spécialisées

Python dispose d'un riche écosystème de bibliothèques spécialisées dans le calcul scientifique et l'analyse de données, telles que NumPy, SciPy, Pandas et Matplotlib. Ces bibliothèques offrent des fonctions et des outils puissants pour effectuer des opérations mathématiques complexes, analyser des ensembles de données volumineux et créer des visualisations informatives, ce qui facilite considérablement le développement de ce programme.

III.1.3 Lisibilité et Simplicité :

Python est connu pour sa syntaxe claire et intuitive, ce qui le rend facile à apprendre et à utiliser, même pour les débutants en programmation. Cette simplicité permet aux développeurs de se concentrer sur la logique métier de ce programme plutôt que sur les complexités syntaxiques du langage.

III.1.4 Large Communauté et Ressources :

Python bénéficie d'une communauté active et vaste de développeurs et d'utilisateurs, ce qui signifie qu'il existe une abondance de ressources disponibles, telles que des tutoriels, des exemples de code et

des forums de discussion, pour aider les développeurs à résoudre des problèmes et à améliorer leurs programmes.

III.1.5 Flexibilité et évolutivité :

Python est un langage flexible et évolutif qui peut s'adapter à des projets de différentes tailles et complexités. Cela permet de faire évoluer facilement ce programme au fur et à mesure que les besoins et les exigences changent.

III.2 Les Bibliothèques Utilisées dans le Programme

Le programme développé en Python utilise plusieurs bibliothèques importantes pour accomplir ses fonctionnalités. Voici une description détaillée de chaque bibliothèque, en mettant en évidence ses principales fonctionnalités et son rôle dans le programme 317_FMD :

III.2.1 Tkinter

Tkinter est une bibliothèque d'interface utilisateur graphique (GUI) intégrée à Python. Elle permet de créer des interfaces conviviales pour les applications Python, ce qui facilite l'interaction des utilisateurs avec ce programme. Dans le cadre de ce programme, Tkinter est principalement utilisé pour [14]:

- Créer une interface graphique intuitive : Tkinter permet de concevoir des fenêtres, des boutons, des menus et d'autres éléments d'interface graphique pour permettre aux utilisateurs d'importer leurs données TBF (temps entre pannes) et TTR (temps de réparation) à partir de fichiers Excel.
- Gérer les interactions des utilisateurs : Tkinter permet de capturer les actions des utilisateurs, telles que les clics de souris et les entrées au clavier, afin de guider le programme FMD dans ses processus d'analyse et de visualisation.
- Afficher les résultats et les visualisations : Tkinter peut être utilisé pour afficher des tableaux de données, des graphiques et d'autres éléments visuels pour présenter les résultats de l'analyse FMD aux utilisateurs.

III.2.2 NumPy

NumPy est une bibliothèque de calcul scientifique pour Python. Elle offre des outils puissants pour effectuer des opérations mathématiques sur des tableaux de données volumineux, ce qui est crucial

pour le traitement et l'analyse des données TBF et TTR dans ce programme. Les fonctionnalités clés de NumPy utilisées dans ce programme incluent [15] :

- Manipulation de tableaux : NumPy permet de créer, de modifier et de gérer des tableaux multidimensionnels, qui sont essentiels pour stocker et organiser les données TBF et TTR.
- Opérations mathématiques : NumPy fournit un large éventail de fonctions mathématiques, telles que la somme, la moyenne, l'écart-type, le calcul de puissances et d'exponentielles, qui sont utilisées pour calculer des statistiques importantes sur les données de défaillance.
- Fonctions statistiques : NumPy offre des fonctions statistiques avancées, telles que la distribution normale, la distribution exponentielle et la distribution Weibull, qui sont utilisées pour ajuster des modèles de fiabilité aux données TBF et TTR.

III.2.3 SciPy et SciPy.stats

SciPy est une bibliothèque de calcul scientifique avancée pour Python. Elle complète NumPy en fournissant des fonctions spécialisées pour l'ajustement de modèles statistiques, l'optimisation et d'autres tâches scientifiques. SciPy.stats, un sous-module de SciPy, est particulièrement utile pour le programme FMD car il offre des fonctionnalités spécifiques à l'analyse de la fiabilité et de la maintenance. Les fonctionnalités clés de SciPy et SciPy.stats utilisées dans ce programme incluent [16] :

- Ajustement de modèles de fiabilité : SciPy.stats permet d'ajuster des modèles de fiabilité courants, tels que les modèles exponentiels, Weibull et lognormaux, aux données TBF et TTR. Cela permet d'estimer les paramètres clés de la fiabilité, tels que le temps moyen entre pannes (MTBF) et le taux de défaillance.
- Optimisation des paramètres : SciPy.stats fournit des algorithmes d'optimisation pour trouver les valeurs optimales des paramètres des modèles de fiabilité ajustés, ce qui garantit la meilleure adéquation possible des modèles aux données.
- Calcul de mesures de fiabilité : SciPy.stats offre des fonctions pour calculer des mesures de fiabilité importantes, telles que la fonction de fiabilité ($R(t)$), la fonction de risque ($\lambda(t)$) et la fonction de répartition ($F(t)$), qui permettent d'évaluer la probabilité de défaillance des équipements au fil du temps.

III.2.4 Pandas

Pandas est une bibliothèque de manipulation et d'analyse de données pour Python. Elle offre des outils puissants pour importer, nettoyer, prétraiter et analyser des ensembles de données volumineux, ce qui est essentiel pour gérer efficacement les données TBF et TTR dans ce programme. Les fonctionnalités clés de Pandas utilisées dans ce programme incluent [17] :

- Lecture de données Excel : Pandas permet de lire des données tabulaires à partir de fichiers Excel, ce qui facilite l'importation des données TBF et TTR dans le programme.
- Nettoyage et préparation des données : Pandas offre des outils pour nettoyer les données, gérer les valeurs manquantes et formater les données de manière

III.2.5 Matplotlib

Matplotlib est une bibliothèque de visualisation pour la création de divers types de graphiques et de diagrammes. Dans ce programme, elle est utilisée pour [18] :

- Visualiser la fiabilité et la maintenabilité, la disponibilité : Création de graphiques de survie $(R(t))$ et $(\lambda(t))$ pour illustrer la probabilité de survie ou de défaillance des équipements. Génération de graphiques CDF des temps de réparation pour évaluer la probabilité d'une réparation rapide $M(t)$. Création de graphiques de disponibilité pour représenter la probabilité que l'équipement soit opérationnel à un moment donné $D(t)$.
- Visualiser les données : Création de nuages de points, d'histogrammes et de courbes pour explorer les relations entre les variables, examiner les distributions de données et suivre les tendances.

III.3 Les Objectifs et Spécifications du Programme FMD

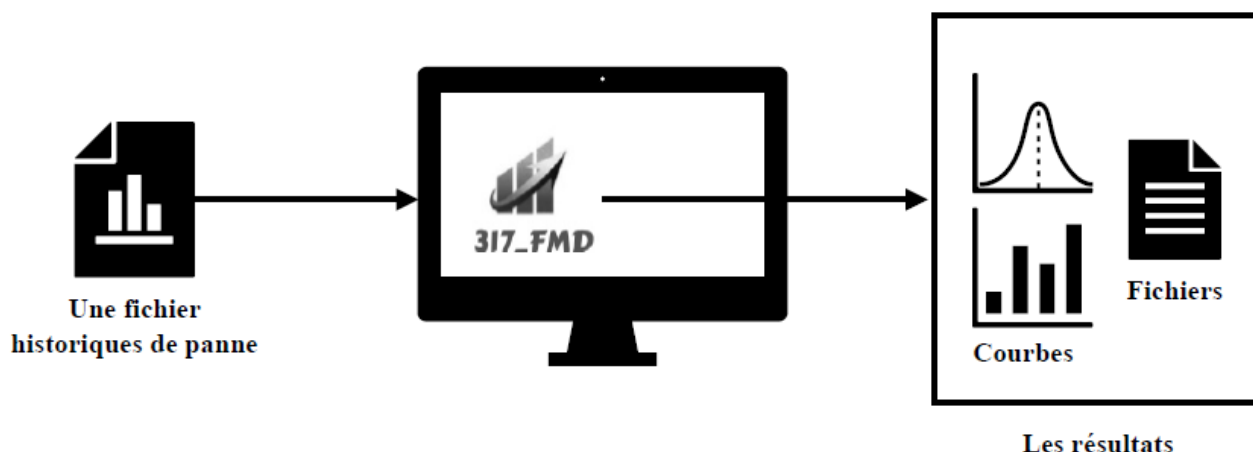


Fig III. 1 : l'objectif de la programme

III.3.1 Objectifs :

Le programme vise à fournir un outil complet pour l'analyse de la fiabilité, de la maintenabilité et de la disponibilité des équipements. Il doit répondre aux besoins des utilisateurs en offrant les fonctionnalités suivantes :

III.3.1.1 Importation de données :

Le programme doit pouvoir importer des données TBF (temps entre pannes) et TTR (temps de réparation) à partir de divers formats de fichiers, tels que CSV, Excel ou bases de données.

III.3.1.2 Analyse de la fiabilité :

Le programme doit implémenter des méthodes statistiques pour analyser les données TBF et estimer des paramètres clés de fiabilité comme le MTBF (temps moyen entre pannes), le taux de défaillance et la fonction de survie.

III.3.1.3 Modélisation de la fiabilité :

Le programme doit permettre l'ajustement de modèles de fiabilité courants, tels que les modèles exponentiels, Weibull et lognormaux, aux données TBF pour identifier le modèle qui correspond le mieux aux caractéristiques de défaillance des équipements.

III.3.1.4 Analyse de la maintenabilité :

Le programme doit fournir des outils pour analyser les données TTR et évaluer la maintenabilité des équipements. Cela peut inclure le calcul de la moyenne, de la médiane et du quartile des temps de réparation, ainsi que l'analyse de la distribution des temps de réparation.

III.3.1.5 Analyse de la disponibilité :

Le programme doit permettre de calculer des mesures de disponibilité importantes, telles que la fonction de disponibilité ($D(t)$) et la disponibilité moyenne, pour évaluer la probabilité que les équipements soient opérationnels à un moment donné.

Visualisation des résultats : Le programme doit générer des visualisations claires et informatives pour présenter les résultats des analyses de fiabilité, de maintenabilité et de disponibilité. Cela peut inclure des graphiques de survie, des histogrammes, des courbes de densité de probabilité et des graphiques de disponibilité.

III.3.1.6 Génération de rapports :

Le programme doit avoir la capacité de générer des rapports complets résumant les résultats des analyses, y compris les paramètres de fiabilité estimés, les modèles ajustés, les mesures de maintenabilité et de disponibilité, et les visualisations correspondantes.

III.3.2 Spécifications :

III.3.2.1 Interface utilisateur :

Le programme doit disposer d'une interface utilisateur graphique (GUI) conviviale et intuitive pour faciliter l'interaction des utilisateurs. Tkinter peut être utilisé pour créer une interface graphique simple et efficace.

III.3.2.1 Gestion des données :

Le programme doit pouvoir gérer efficacement les données TBF et TTR, en s'assurant de leur intégrité, de leur cohérence et de leur formatage approprié. Pandas est une bibliothèque puissante pour la manipulation et l'analyse des données tabulaires.

III.3.2.2 Calculs statistiques :

Le programme doit implémenter des fonctions mathématiques et statistiques robustes pour effectuer les calculs nécessaires à l'analyse de la fiabilité, de la maintenabilité et de la disponibilité. NumPy et SciPy sont des bibliothèques scientifiques essentielles pour ce type de calculs.

III.3.2.1 Visualisation de données :

Le programme doit utiliser des bibliothèques de visualisation comme Matplotlib pour créer des graphiques et des diagrammes de haute qualité qui communiquent efficacement les résultats des analyses.

III.4 Le Code Source pour le Programme FMD :

Le développement de ce programme implique la création d'une interface utilisateur (frontend) et d'un backend qui gère les calculs et la logique métier.

III.4.1 Développer le Frontend avec Tkinter :

Le code présenté implémente une interface graphique (GUI) pour L'application FMD. La GUI est structurée en sections distinctes pour la saisie des données, les options de calcul, l'affichage des résultats et le contrôle du mode de fonctionnement :

Voici le code avec des commentaires en vert décrivant chaque paragraphe du programme de frontend ;

```
# Créer la fenêtre principale
fenetre = tk.Tk()
fenetre.title('317_FMD')
fenetre.iconbitmap('c:\Memoire fin\MEMOIRE PROGRAMME\IMG_8501.ico')

# Définir la taille et la position de la fenêtre
l = 600
h = 400
left = int((fenetre.winfo_screenwidth() - l) / 2)
top = int((fenetre.winfo_screenheight() - h) / 2)
fenetre.geometry(f'{l}x{h}+{left}+{top}')
fenetre.minsize(500, 400)

# Appliquer un style à la fenêtre (thème sombre par défaut)
style = ttk.Style(fenetre)
fenetre.tk.call("source", "forest-light.tcl") # Thème clair (facultatif)
fenetre.tk.call("source", "forest-dark.tcl") # Thème sombre
style.theme_use("forest-dark")

# Créer un cadre principal pour organiser les éléments de l'interface
frame = ttk.Frame(fenetre)
frame.pack(expand=True, fill="both")
```

```
# Créer un cadre pour les résultats en bas de la fenêtre
sortie_frame_0 = ttk.Frame(frame)
sortie_frame_0.pack(side='bottom', fill="both")

# Créer un cadre avec étiquette pour la section d'insertion de fichier
frame_inser_fichier = ttk.LabelFrame(frame, text="Insérer un fichier")
# frame_inser_fichier.grid(row=0,column=0 ,padx = 5, pady = 15)
frame_inser_fichier.pack(side='left', fill="both", padx=5, pady=10)

# Variable pour stocker le chemin du fichier
chemine = tk.StringVar()

# Champ d'entrée pour le chemin du fichier
entre_1 = ttk.Entry(frame_inser_fichier, textvariable=chemine)
entre_1.insert(0, "Le chemin du fichier")
entre_1.bind("<FocusIn>", lambda e: entre_1.delete('0', 'end'))
entre_1.grid(row=0, column=0, columnspan=2, sticky="ew", padx=5, pady=5)

# Bouton pour afficher le contenu du fichier (à implémenter)
bouton_de_affichage = ttk.Button(frame_inser_fichier, text='Afficher',
command=ok)
bouton_de_affichage.grid(row=1, column=0, sticky="ew", padx=5, pady=5)

# Bouton pour importer les données du fichier (à implémenter)
bouton_importer = ttk.Button(frame_inser_fichier, text="Importer",
command=importer_fichier)
bouton_importer.grid(row=1, column=1, sticky="ew", padx=5, pady=5)

# Espace réservé pour les boutons de calcul des fonctions
place_texte = ttk.Label(frame_inser_fichier, text='Fonctions à calculer')
place_texte.grid(row=2, column=0, sticky="ew", pady=5, padx=20)

# Séparateur horizontal
separator = ttk.Separator(frame_inser_fichier)
separator.grid(row=2, column=1, sticky="ew", pady=5, padx=5)

# Menu déroulant pour les fonctions de fiabilité
button1 = ttk.Menubutton(frame_inser_fichier, text="Fiabilité")
button1.grid(row=3, column=0, sticky="ew", columnspan=2, pady=5, padx=5)

# Séparateur horizontal
separator_2 = ttk.Separator(frame_inser_fichier)
separator_2.grid(row=4, column=0, sticky="ew", columnspan=2, pady=5, padx=25)

# Menu déroulant pour les fonctions de maintenabilité
button2 = ttk.Menubutton(frame_inser_fichier, text="Maintenabilité")
button2.grid(row=5, column=0, sticky="ew", columnspan=2, pady=5, padx=5)
```

```
# Séparateur horizontal
separator_3 = ttk.Separator(frame_inser_fichier)
separator_3.grid(row=6, column=0, sticky="ew", columnspan=2, pady=5, padx=25)

# Menu déroulant pour les fonctions de disponibilité
button3 = ttk.Menubutton(frame_inser_fichier, text="Disponibilité")
button3.grid(row=7, column=0, sticky="ew", columnspan=2, pady=5, padx=5)

# Séparateur horizontal
separator_3 = ttk.Separator(frame_inser_fichier)
separator_3.grid(row=8, column=0, sticky="ew", columnspan=2, pady=5, padx=5)

# Bouton bascule pour changer de mode (fonction à définir)
mode_switch = ttk.Checkbutton(frame_inser_fichier,
                              text="Mode", style="Switch",
                              command=mode)
mode_switch.grid(row=9, column=0, sticky="nsew", columnspan=2, pady=5,
                 padx=5)

# Cadre pour l'arborescence de résultats
treeview_frame = ttk.Frame(frame)
treeview_frame.pack(side='right', expand=True, fill="both")

# Barre de défilement verticale pour l'arborescence
scrollbar = ttk.Scrollbar(treeview_frame)
scrollbar.pack(side="right", fill="y")

# Arborescence de résultats (hauteur de 12 lignes)
tree = ttk.Treeview(treeview_frame, height=12, yscrollcommand=scrollbar.set)
tree.pack(side='left', expand=True, fill="both", padx=5, pady=12)

# Configurer la barre de défilement pour suivre le contenu de l'arborescence
scrollbar.config(command=tree.yview)

# Variable pour stocker le texte de sortie
sortier = tk.StringVar()

# Cadre pour afficher les résultats
Sortie_frame = ttk.LabelFrame(sortie_frame_0, text="Les résultats:")
Sortie_frame.pack(side="left", expand=True, fill="both", padx=5, pady=10)

# Zone d'affichage des résultats (police Calibri 10)
place_texte_2 = ttk.Label(Sortie_frame, textvariable=sortier, font="calibri
10")
sortier.set("")
place_texte_2.pack(expand=True, fill="both", padx=5)
```

```
# Cadre pour le bouton d'enregistrement
boutton5_frame = ttk.Frame(sortie_frame_0)
boutton5_frame.pack(side="right", fill="both", padx=5, pady=15)

# Bouton pour enregistrer les résultats dans un fichier Excel (à définir)
boutton5 = ttk.Button(boutton5_frame, text="Enregistrer",
command=enregistrer_excel)
boutton5.pack(side="top", fill="x", padx=5)
```

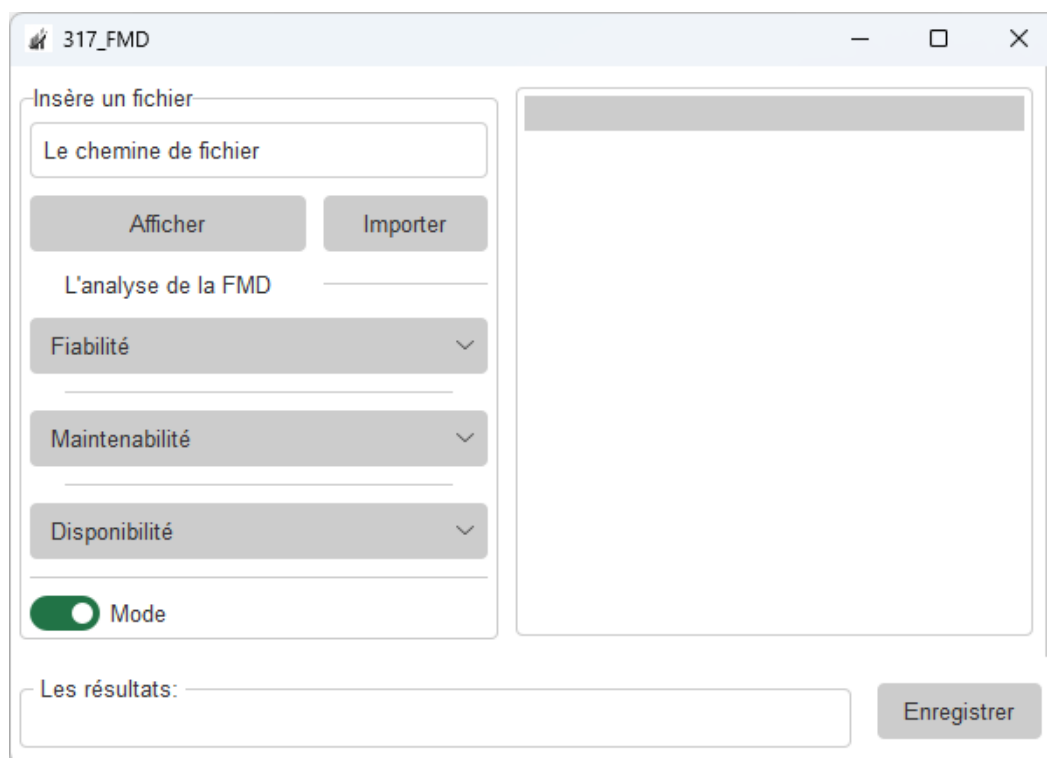


Fig III. 2 : L'interface d'utilisateur de la programme « 317_FMD »

III.4.2 Développer le Backend :

Le backend du programme FMD assure le traitement des données, les calculs complexes, la communication avec l'interface utilisateur et la génération de rapports détaillés, offrant ainsi un outil complet pour l'analyse de la fiabilité, de la maintenabilité et de la disponibilité

Voici un exemple de code qui illustre la fonction d'analyse de la fiabilité et de la maintenabilité basée sur la loi de Weibull.

III.4.2.1 Fonction de la loi de weibull :

```
def loi_weibull(nom_colonne, commande):
    """
    Cette fonction analyse une distribution de Weibull en utilisant des données
    tabulaires.
    Arguments:
    nom_colonne (str): Nom de la colonne contenant les données temporelles (TBF
    ou TTR).
    commande (str): Commande spécifiant le type d'analyse à effectuer (Fiabilité,
    Fonction de répartition, Densité de probabilité, Taux de défaillance,
    Toutes).
    Retourne:
    Un DataFrame contenant les résultats de l'analyse et les caractéristiques de
    la distribution de Weibull.
    """

    # Récupérer le chemin du fichier Excel
    chemin_0 = chemine.get()
    # Définir le chemin d'accès au fichier Excel contenant les données
    chemine_0 = "chemin/vers/votre/fichier.xlsx"
    # Lire les données du fichier Excel dans un DataFrame Pandas
    donnees = pd.read_excel(chemine_0)
    # Choisir le nome de la colonne à analyser
    nom_colonne = "nom_colonne" # Remplacer par le nom de la colonne réel
    # Trier les données par la colonne choisie
    donnees_triees = donnees.sort_values(by=nom_colonne)
    # Ajouter une nouvelle colonne "i" avec des valeurs entières séquentielles
    donnees_triees["i"] = range(1, len(donnees_triees) + 1)
    # Extraire les données triées et indexées dans un DataFrame séparé
    donnees_weibull = donnees_triees[[nom_colonne,
    "i"]].reset_index(drop=True)
    # Calculer le nombre de points de données
    N = len(donnees_weibull)
    # Définir la fonction de distribution empirique de Weibull (F(i))
    if N <= 20:
    # Pour les petits échantillons (N <= 20), utiliser la correction de Nelson-
    Aalen-Liebman
        donnees_weibull["F(i)"] = round(((donnees_weibull["i"] - 0.30) / (N +
    0.4)), 6)
    else:
    # Pour les échantillons plus grands (N > 20), utiliser le calcul standard
        donnees_weibull["F(i)"] = round((donnees_weibull["i"] / (N + 1)), 6)
```

```

# Appliquer une transformation logarithmique aux données pour la régression
linéaire
    donnees_weibull["X"] = round((np.log(donnees_weibull[nom_colonne])), 6)
# Appliquer une double transformation logarithmique à F(i) pour la régression
linéaire
    donnees_weibull["Y"] = round(np.log(np.log(1 / (1 -
donnees_weibull["F(i)"]))))), 6)

# Extraire les données pour le traçage
TBF = donnees_weibull[nom_colonne]
TBF_values = np.linspace(min(TBF), max(TBF), 1000)
X = donnees_weibull['X']
Y = donnees_weibull['Y']
# Définition des valeurs constantes pour les paramètres de Weibull
BETTA = [0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, ..]
A = [120.0, 24.0, 9.2605, 5.0291, 3.3234, 2.4786, 2.0, 1.7024, 1.5046,..]
B = [1901.0, 199.0, 50.08, 19.98, 10.44, 6.46, 4.47, 3.35, 2.65, 2.18,..]

def calculer_valeur_beta_A_B(beta_0):
    """
    Cette fonction détermine les valeurs de A et B associées à un paramètre de
    forme  $\beta$  donné.
    Arguments:
    beta_0 (float): Paramètre de forme  $\beta$  de la distribution de Weibull.
    Retourne:
    Un tuple contenant le paramètre de forme  $\beta$ , les valeurs de A et B, et une
    chaîne de caractères résumant les résultats.
    """
    valeur_min = abs(beta_0 - BETTA[0])
    index_min = 0
    for i in range(1, len(BETTA)):
        difference = abs(beta_0 - BETTA[i])
        if difference < valeur_min:
            valeur_min = difference
            index_min = i
    Valeur_BETTA = BETTA[index_min]
    Valeur_A = A[index_min]
    Valeur_B = B[index_min]
    return Valeur_BETTA, Valeur_A, Valeur_B, f"Paramètre de forme  $\beta$  :
{Valeur_BETTA} et A : {Valeur_A} et B : {Valeur_B}\n"

# Définir la fonction pour calculer les paramètres de la droite de Weibull
(pente et ordonnée à l'origine)
def tracer_de_droit_Weibull(col1, col2):
    """
    Cette fonction trace le droit de Weibull en utilisant les données
    transformées.

```

Arguments:

coll (array): Tableau des valeurs de X ($\ln(t)$).

col2 (array): Tableau des valeurs de Y ($\ln(\ln(1/(1-F(t))))$).

retourne:

Un tuple contenant la pente (a), l'ordonnée à l'origine (b), le tableau des valeurs de X, et le tableau des valeurs de Y.

```

"""
    A1 = np.vstack([coll, np.ones(len(coll))]).T
    a, b = np.linalg.lstsq(A1, col2, rcond=None)[0]
    x_droite = coll
    y_droite = a * x_droite + b
    return a, b, x_droite, y_droite
"""

# Calculer les paramètres de la droite de Weibull
dwl = tracer_de_droit_Weibull(X, Y)

# Créer le diagramme de dispersion des points de données
plt.plot(X, Y, linestyle="--", marker='o', color="blue", label='Nuage de
points des médiane F(i) en fonction de TBF')
# Tracer la droite de Weibull
plt.plot(dwl[2], dwl[3], color="r", label='Droite de Weibull')

# Configurer les étiquettes du graphique, la grille, la légende et afficher
le graphique
plt.xlabel(r'Y = $\ln(t)$')
plt.ylabel(r'X = $\ln\left(\ln\left(\frac{1}{1-F(t)}\right)\right)$')
plt.grid(True, which='both', linestyle=':', linewidth=0.5,
color='darkgray')
plt.legend()
plt.show()

# Demander à l'utilisateur de confirmer si les données semblent suivre une
distribution de Weibull
dresse = messagebox.askquestion('', "Le nuage de points semble-t-il
suivre une droite ? (Oui/Non)")
if dresse == "yes": # Si l'utilisateur répond "oui"
    lamda = 0 # Définir la valeur de lambda à 0
    process_redressement(dresse, donnees_weibull, nom_colonne, lamda) #
Appeler la fonction de redressement avec lambda = 0

elif dresse == "no": # Si l'utilisateur répond "non"
    p = np.polyfit(X, Y, 2) # Ajuster un polynôme du second degré aux
données transformées
    premier = Y[1] # Obtenir la deuxième valeur de Y (indice 1)
    deuxieme = Y[N-2] # Obtenir la valeur de Y avant-dernière (indice N-
2)

```

```

    troisieme = (premier + deuxieme)/2 # Calculer la moyenne de premier
et deuxieme
    t_1 = TBF[1] # Obtenir la deuxième valeur de TBF (indice 1)
    t_2 = TBF[N-1] # Obtenir la valeur de TBF avant-dernière (indice N-
1)
    t_3 = np.exp(np.polyval(p,troisieme)) # Evaluer le polynôme à la
valeur moyenne (troisieme)
    lamda = ((t_3**2) - (t_1*t_2))/((2*t_3) - t_1 - t_2) # Calculer la
valeur de lambda
    def process_redressement(dresse, donnees_weibull, nom_colonne, lamda):
# Vérification du redressement des données
    if dresse == "oui":

        # Calcul des paramètres de Weibull
        betta_0 = dwl[0]
        vv = calculer_valeur_betta_A_B(betta_0)
        Valeur_BETTA = vv[0]
        beta_a_b = vv[3]
        tetta = np.exp(-dwl[1]/betta_0)
# Cette section calcule les paramètres de Weibull ( $\eta$  et  $\gamma$ ) pour le cas où le
redressement est nécessaire.
# Formatage et affichage des paramètres de Weibull
    tetta_lamda = f"Paramètre d'échelle  $\eta$ : {round(tetta,4)} \nParamètre
de position  $\gamma$ : {round(lamda,4)}\n"
    print(tetta_lamda)
# Cette section formate et affiche les paramètres de Weibull ( $\eta$  et  $\gamma$ ).
# Calcul de F(t)
    donnees_weibull['F(t)'] = round(1 - np.exp(-(((TBF-
lamda)/tetta)**Valeur_BETTA)), 6)
# Cette ligne calcule la fonction de répartition de Weibull (F(t)) et la
stocke dans une nouvelle colonne nommée "F(t)".
    # Calcul de Dn
    donnees_weibull['Dn'] =round(np.abs( donnees_weibull['F(i)'] -
donnees_weibull['F(t)']), 6)
# Cette ligne calcule l'écart entre la fonction de répartition empirique
(F(i)) et la fonction de répartition de Weibull (F(t)) et l'absolutise, puis
le stocke dans une nouvelle colonne nommée "Dn".
# Calcul de la fonction de fiabilité R(t)
    if nom_colonne =='TBF':
        F1 = 1 - np.exp(-(((TBF_values-lamda)/tetta)**Valeur_BETTA))
        donnees_weibull['R(t)'] =round((1 - donnees_weibull['F(t)']), 6)
# Cette section calcule la fonction de fiabilité R(t) et la stocke dans une
nouvelle colonne nommée "R(t)".

    R1 = 1-F1
# Cette ligne calcule la fiabilité cumulative R(t) pour chaque valeur de TBF.

```

```
# Calcul de la fonction de densité de probabilité f(t)
    donnees_weibull['f(t)'] = round((Valeur_BETTA/tetta)*(((TBF-
lamda)/tetta)**(Valeur_BETTA-1))*np.exp(-(((TBF-
lamda)/tetta)**Valeur_BETTA)), 6)
    f1 = (Valeur_BETTA/tetta)*(((TBF_values-lamda)/tetta)**(Valeur_BETTA-
1))*np.exp(-(((TBF_values-lamda)/tetta)**Valeur_BETTA))

# Cette section calcule la fonction de densité de probabilité f(t) et la
stocke dans une nouvelle colonne nommée "f(t)".

    # Calcul du taux de défaillance  $\lambda(t)$ 

    donnees_weibull[' $\lambda(t)$ '] = round((Valeur_BETTA/tetta)*(((TBF-
lamda)/tetta)**(Valeur_BETTA-1)),6)
    taux_1 = (Valeur_BETTA/tetta)*(((TBF_values-
lamda)/tetta)**(Valeur_BETTA-1))

# Cette section calcule le taux de défaillance  $\lambda(t)$  et la stocke dans une
nouvelle colonne nommée " $\lambda(t)$ ".

# Traçage des courbes selon la commande

    if commande == 'Fiabilité':
        trace_corbes(TBF_values,R1,'Fiabilité \n', 'TBF', 'R(t)')
        donnees_weibull = donnees_weibull[[nom_colonne, 'R(t)']]

# Cette section trace la courbe de fiabilité R(t) et sélectionne les
colonnes "TBF" et "R(t)" dans le jeu de données donnees_weibull.

    elif commande == 'Fonction de répartition':
        trace_corbes(TBF_values,F1,'Fonction de répartition
\n', 'TBF', 'F(t)')
        donnees_weibull = donnees_weibull[[nom_colonne, 'F(t)']]

# Cette section trace la courbe de la fonction de répartition F(t) et
sélectionne les colonnes "TBF" et "F(t)" dans le jeu de données
donnees_weibull.
    elif commande == 'Densité de probabilité':
        trace_corbes(TBF_values,f1,'Densité de probabilité
\n', 'TBF', 'f(t)')
        donnees_weibull= donnees_weibull[[nom_colonne, 'f(t)']]

# Cette section trace la courbe de la densité de probabilité f(t) et
sélectionne les colonnes "TBF" et "f(t)" dans le jeu de données
donnees_weibull.

    elif commande == 'Taux de défaillance':
```

```

        trace_corbes(TBF_values , taux_1 , 'Taux de défaillance
\n', 'TBF', r"<span class="math-inline">\lambda \ (t)\</span>")
        donnees_weibull= donnees_weibull[[nom_colonne, '\lambda(t)']]

# Cette section trace la courbe du taux de défaillance  $\lambda(t)$  et sélectionne
les colonnes "TBF" et " $\lambda(t)$ " dans le jeu de données donnees_weibull.

        elif commande == 'Toutes':
            trace_corbes(TBF_values , R1, ' Fiabilité \n', 'TBF', 'R (t)' )
            trace_corbes(TBF_values, F1, 'Fonction de répartition \n'
, 'TBF', 'F(t)')
            trace_corbes(TBF_values , f1 , ' Densité de probabilité
\n', 'TBF', r'$ f (t)$' )
            trace_corbes(TBF_

# Calcul de la fonction de maintenabilité M(t) pour le cas TTR
elif nom_colonne == 'TTR':
    M1 = 1 - np.exp(-((TBF_values-lamda)/tetta)**Valeur_BETTA))
    # Cette ligne calcule la fonction de maintenabilité M(t) pour le cas
TTR.

    # Renommage des colonnes F(i) et F(t) en M(i) et M(t)
    donnees_weibull =
donnees_weibull.rename(columns={'F(i)':'M(i)', 'F(t)':'M(t)'})
    # Cette ligne renomme les colonnes F(i) et F(t) en M(i) et M(t) pour
le cas TTR.

    # Calcul de la fonction de densité de probabilité g(t)
    donnees_weibull['g(t)'] = round((Valeur_BETTA/tetta)*((TBF-
lamda)/tetta)**(Valeur_BETTA-1))*np.exp(-((TBF-
lamda)/tetta)**Valeur_BETTA)), 6)
    g1 = (Valeur_BETTA/tetta)*(((TBF_values-lamda)/tetta)**(Valeur_BETTA-
1))*np.exp(-((TBF_values-lamda)/tetta)**Valeur_BETTA))
    # Cette section calcule la fonction de densité de probabilité g(t)
pour le cas TTR et la stocke dans une nouvelle colonne nommée "g(t)".
    # Calcul du taux de réparation  $\mu(t)$ 
    donnees_weibull[' $\mu(t)$ '] = round((Valeur_BETTA/tetta)*((TBF-
lamda)/tetta)**(Valeur_BETTA-1)), 6)
    taux_1 = (Valeur_BETTA/tetta)*((TBF_values-
lamda)/tetta)**(Valeur_BETTA-1)

    # Cette section calcule le taux de réparation  $\mu(t)$  pour le cas TTR et
la stocke dans une nouvelle colonne nommée " $\mu(t)$ ".

    # Traçage des courbes selon la commande
    if commande == 'Maintenabilité':
        trace_corbes(TBF_values, M1, 'Maintenabilite \n', 'TTR', 'M(t)')
        donnees_weibull = donnees_weibull[[nom_colonne, 'M(t)']]

```

```

        # Cette section trace la courbe de maintenabilité M(t) et
sélectionne les colonnes "TTR" et "M(t)" dans le jeu de données
donnees_weibull.

        elif commande == 'Densité de probabilité':
            trace_corbes(TBF_values,g1,'Densité de probabilité
\n','TTR','g(t)')
            donnees_weibull= donnees_weibull[[nom_colonne, 'g(t)']]

        # Cette section trace la courbe de la densité de probabilité g(t)
et sélectionne les colonnes "TTR" et "g(t)" dans le jeu de données
donnees_weibull.
        elif commande == 'Taux de réparation':
            trace_corbes(TBF_values , taux_1 , 'Taux de réparation
\n','TTR',r"<span class="math-inline"> $\mu$ (t)</span>")
            donnees_weibull = donnees_weibull[[nom_colonne, ' $\mu$ (t)']]

# Cette section trace la courbe du taux de réparation  $\mu$ (t) et sélectionne les
colonnes "TTR" et " $\mu$ (t)" dans le jeu de données donnees_weibull.

        elif commande == 'Toutes':
            trace_corbes(TBF_values , M1 , ' Maintenabilité \n','TTR','M(t)' )
            trace_corbes(TBF_values , g1 , ' Densité de probabilité \n','TTR',
r'$ f (t)<span class="math-inline">' \)
            trace_corbes(TBF_values , taux_1 , 'Taux de réparation\n','TTR'
,r"</span> $\mu$ (t)$")
            donnees_weibull = donnees_weibull[[nom_colonne, 'M(t)', 'g(t)',
' $\mu$ (t)']]
# Cette section trace les courbes de maintenabilité M(t), densité de
probabilité g(t), taux de réparation  $\mu$ (t) et sélection

        # Définition des valeurs critiques pour le test de Kolmogorov-Smirnov (K-
S)
        N__ = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
19,..]
        alpha_005 = [0.975, 0.84189, 0.7076, 0.62394, 0.56328, 0.51926,..]

        # Calcul de la valeur maximale de la différence entre les valeurs de F(t)
consécutives
        valeur_max = round(donnees_weibull['Dn'].max(), 5)

        # Détermination du nombre de valeurs dans le DataFrame
        valeur = len(donnees_weibull)

        # Calcul de la valeur critique de Dn pour le test de K-S
        if valeur <= 50:

```

```
valeur_min = abs(valeur - N__[0])
index_min = 0
for i in range(1, len(N__)):
    difference = abs(valeur - N__[i])
    if difference < valeur_min:
        valeur_min = difference
        index_min = i
Dn_Alpha = alpha_005[index_min]
else:
    Dn_Alpha = 1.36 / np.sqrt(valeur)

# Application du test de Kolmogorov-Smirnov (K-S)
if valeur_max < Dn_Alpha:
    dnn = f"La valeur maximale de Dn,max {valeur_max} est inférieure à
Dn,alpha {Dn_Alpha}. \nAlors, d'après le test de K-S avec un seuil d'alpha =
5 % , le module de Weibull est accepté."
else:
    dnn = f"La valeur maximale de Dn {valeur_max} est supérieure à
Dn_Alpha {Dn_Alpha}. \nAlors, d'après le test de K-S avec un seuil d'alpha = 5
% , le module de Weibull est rejeté."

# Mise à jour de l'arborescence
for row in tree.get_children():
    tree.delete(row)
# Cette boucle supprime tous les éléments existants de l'arborescence.

tree["columns"] = list(donnees_weibull.columns)
# Cette ligne définit les en-têtes de colonnes de l'arborescence en fonction
des noms de colonnes du jeu de données donnees_weibull.

tree["show"] = "headings"
# Cette ligne active l'affichage des en-têtes de colonnes dans
l'arborescence.

# Configuration des colonnes
for colonne in tree["columns"]:
    tree.column(colonne, width=10)

# Cette boucle définit la largeur de toutes les colonnes à 10 unités.

for colonne in tree["columns"]:
    tree.heading(colonne, text=colonne)

# Cette boucle définit le texte d'entête pour chaque colonne en utilisant son
nom.

# Remplissage de l'arborescence avec les données
```

```
for index, ligne in donnees_weibull.iterrows():
    tree.insert("", "end", text=index, values=ligne.tolist())

# Cette boucle parcourt chaque ligne du jeu de données donnees_weibull et
# insère une nouvelle ligne dans l'arborescence.
# - index correspond à l'index de la ligne dans le jeu de données.
# - ligne correspond à la ligne de données.
# - values=ligne.tolist() convertit la ligne en une liste pour l'insertion
# dans l'arborescence.

# Affichage des résultats dans la zone de texte "sortier"
sortier.set(beta_a_b + tetta_lamda + M_ecar + dnn)

def trace_corbes(X1, Y1, titre, X_titre, Y_titre):
    """
    Fonction pour tracer une courbe et afficher la figure
    Cette fonction crée un graphique en ligne personnalisé avec des options de
    mis en page et d'annotation.
    Args:
    X1 (list ou NumPy array): Données pour l'axe x.
    Y1 (list ou NumPy array): Données pour l'axe y.
    titre (str): Titre du graphique.
    X_titre (str): Étiquette de l'axe x.
    Y_titre (str): Étiquette de l'axe y.
    Returns:
    None: Affiche le graphique et ne renvoie aucune valeur.
    """
    # Création du graphique en ligne
    plt.plot(X1, Y1, linestyle="-")
    # Ajout d'une grille et personnalisation de l'apparence
    plt.grid(True, which='both', linestyle=':', linewidth=0.5,
             color='darkgray')
    # Ajout du titre et des étiquettes
    plt.title(titre, fontsize=14, fontweight='bold',
              fontdict=dict(family='Times New Roman'))
    plt.xlabel(X_titre)
    plt.ylabel(Y_titre)
    # Rotation des étiquettes des graduations de l'axe x
    plt.tick_params(axis='x', rotation=45)

    # Contrôle de la densité des graduations sur les axes
    plt.locator_params(axis='x', nbins=25)
    plt.locator_params(axis='y', nbins=20)
    # Affichage du graphique
    plt.show()
```

Les autres fonctions pour les autres lois statistiques présentent une structure similaire.

III.4.2.2 Fonction pour la loi exponentielle :

```
def exponentielle(nom_colonne, commande):  
    """  
  
    Cette fonction calcule la fiabilité et la maintenabilité en utilisant la loi  
    exponentielle, basée sur la colonne de données fournie (`nom_colonne`)  
    et d'autres paramètres ou réglages (`commande`).  
    Args:  
    nom_colonne (str): Le nom de la colonne contenant les données.  
    commande (dict): Un dictionnaire contenant des paramètres ou réglages  
    supplémentaires (facultatif).  
    Returns:  
    La valeur de fiabilité calculée.  
  
    """  
    pass
```

III.4.2.3 Fonction pour calculer la disponibilité :

```
def disponibilite():  
    """  
  
    Cette fonction calcule la disponibilité du système.  
    Returns:  
    La valeur de disponibilité calculée.  
  
    """  
    Pass
```

III.4.2.4 Fonction pour le loi log-normale :

```
def lognormal(nom_colonne, commande):  
    """  
  
    Cette fonction calcule les paramètres de la distribution et la fiabilité et  
    la maintenabilité en utilisant la loi log-normale ( $\mu$  et  $\sigma$ ) et en  
    utilisant la colonne de données fournie (`nom_colonne`) et d'autres  
    paramètres ou réglages (`commande`).  
  
    Args:  
    nom_colonne (str): Le nom de la colonne contenant les données.
```

```
commande (dict): Un dictionnaire contenant des paramètres ou réglages supplémentaires (facultatif).
```

```
Returns:
```

```
tuple: Un tuple contenant les paramètres de la distribution log-normale (mu et sigma).
```

```
"""  
Pass
```

Fonction pour gérer le clic du bouton 'ok' :

```
def ok():  
    """  
    Cette fonction gère le clic du bouton 'ok' et peut afficher des instructions ou effectuer d'autres actions.  
    """  
    # en fonction du clic du bouton 'ok'.  
    Pass
```

Fonction pour gérer le changement de mode (activer/désactiver) :

```
def mode():  
    """  
    Cette fonction gère le changement de mode de calcul en fonction de l'état du bouton bascule.  
    """  
    Pass
```

Fonction pour enregistrer les résultats dans un fichier Excel :

```
def enregistrer_excel():  
    """  
    Cette fonction enregistre les résultats calculés dans un fichier Excel.  
    """  
    Pass
```

III.5 Tests de validation :

Les tests de validation d'applications, également connus sous le nom de tests d'acceptation, constituent une étape cruciale du processus de développement logiciel. Ils visent à confirmer que

l'application répond non seulement aux exigences définies à l'avance, mais également aux attentes et aux besoins des utilisateurs finaux. En d'autres termes, on veut par ce test de garantir que l'application fonctionne comme prévu, offre une expérience utilisateur fluide et satisfaisante, et respecte les critères de qualité établis.

Pour ce test, je vais effectuer une analyse FMD de l'historique des pannes de four à billette de la presse 1600T qui est une très importante machine au sein de l'entreprise ALGAL+ sur une période de deux ans.

Le tableau suivant présente l'historique des pannes de four à billette de la presse 1600T sur une période de deux ans [1] :

Date	Eléments	TTR	TBF
01/02/2021	Four a billettes (thermocouple)	16,67	1018,33
16/03/2021	Four a billettes (table billettes)	14,91	1641,08
24/05/2021	Four a billettes (thermocouple)	8,5	567,5
17/06/2021	Four a billettes (table billettes)	14,83	1113,17
03/08/2021	Four a billettes (four de filière)	14,91	2025,08
27/10/2021	Four a billettes (chaîne de table billettes)	15	441
15/11/2021	Four a billettes (le conduit de gaz enflammé)	11,75	1620,25
22/01/2022	Four a billettes (thermocouple)	10,58	1981,42
15/04/2022	Four a billettes (four a chauffe rapide)	31,58	2056,42
11/07/2022	Four a billettes (zone de réchauffage e)	8,58	1239,42
01/09/2022	Four a billettes (four de filière)	11,83	1260,17
24/10/2022	Four a billettes (chaîne de table billettes)	3,5	428,5
11/11/2022	Four a billettes (thermocouple)	4,5	1147,5
29/12/2022	Four a billettes (Zone de réchauffage)	6,08	

Tableau III. 1 : l'historique des pannes de four à billette de la presse 1600T

III.5.1 Préparation des données

Les données historiques doivent être gradées dans une feuille Excel.

Puis définir les colonnes :

- TBF (temps entre les pannes) : cette colonne stockera la durée de fonctionnement sans panne pour chaque événement.

- TTR (temps de réparation) : cette colonne stockera le temps nécessaire pour réparer chaque panne.

III.5.2 Importer les données

vous pouvez utiliser le champ “Chemin de fichier” pour spécifier l’emplacement du fichier que vous souhaitez importer. Vous pouvez entrer manuellement le chemin d’accès ou utiliser un explorateur de fichiers pour le sélectionner.

vous pouvez aussi cliquer sur le bouton “Importer” pour démarrer le processus d’importation.

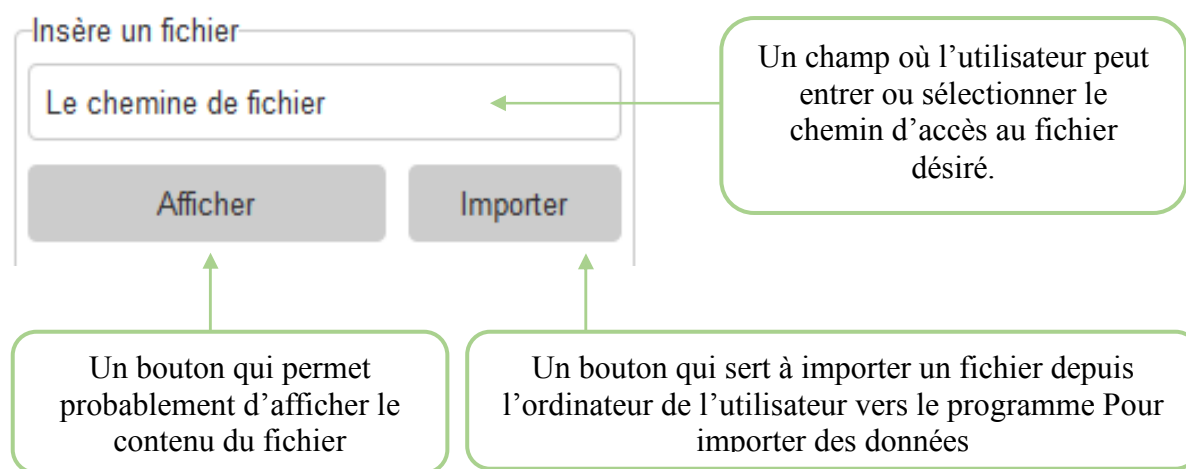


Fig III. 3 : L'importation de donnes au programme « 317_FMD »

Une fois que vous avez importé les données dans votre programme, celui-ci va les afficher à l'utilisateur. Par exemple, si nous importons un fichier d'historique des pannes, le programme va afficher ces données sous forme de tableau :

Date	Elements	TTR	TBF
2021-02-01 00:00:00	Four a billettes (thermocouple)	16.67	1018.33
2021-03-16 00:00:00	Four a billettes (table billettes)	14.91	1641.08
2021-05-24 00:00:00	Four a billettes (thermocouple)	8.5	567.5
2021-06-17 00:00:00	Four a billettes (table billettes)	14.83	1113.17
2021-08-03 00:00:00	Four a billettes (four de filière)	14.91	2025.08
2021-10-27 00:00:00	Four a billettes (chaîne de table billettes)	15.0	441.0
2021-11-15 00:00:00	Four a billettes (le conduit de gaz enflammé)	11.75	1620.25
2022-01-22 00:00:00	Four a billettes (thermocouple)	10.58	1981.42
2022-04-15 00:00:00	Four a billettes (four a chauffe rapide)	31.58	2056.42
2022-07-11 00:00:00	Four a billettes (zone de rechauffag e)	8.58	1239.42
2022-09-01 00:00:00	Four a billettes (four de filière)	11.83	1260.17
2022-10-24 00:00:00	Four a billettes (chaîne de table billettes)	3.5	428.5
2022-11-11 00:00:00	Four a billettes (thermocouple)	4.5	1147.5
2022-12-29 00:00:00	Four a billettes (Zone de rechauffag e)	6.08	nan

Fig III. 4 : L'affichage de donnée

III.5.3 Le choix d'analyse

Après avoir importé les données, vous pouvez les analyser en utilisant les fonctionnalités spécifiques du programme :

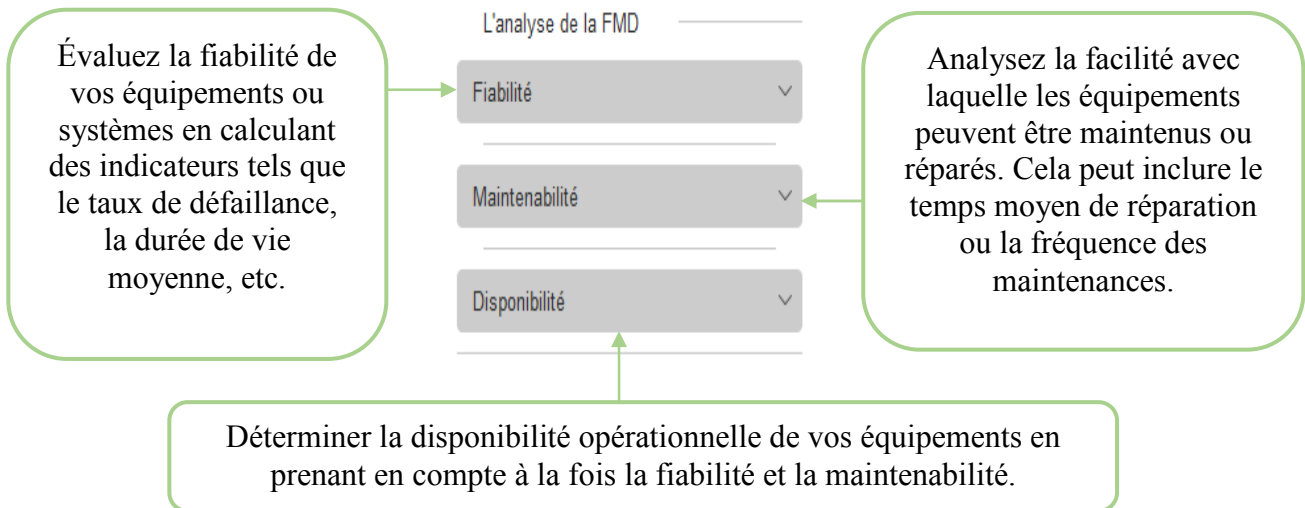


Fig III. 5 : Les choix de l'analyse

Pour procéder à l'analyse, sélectionnez d'abord la loi appropriée à votre étude, puis choisissez la fonctionnalité que vous souhaitez examiner.

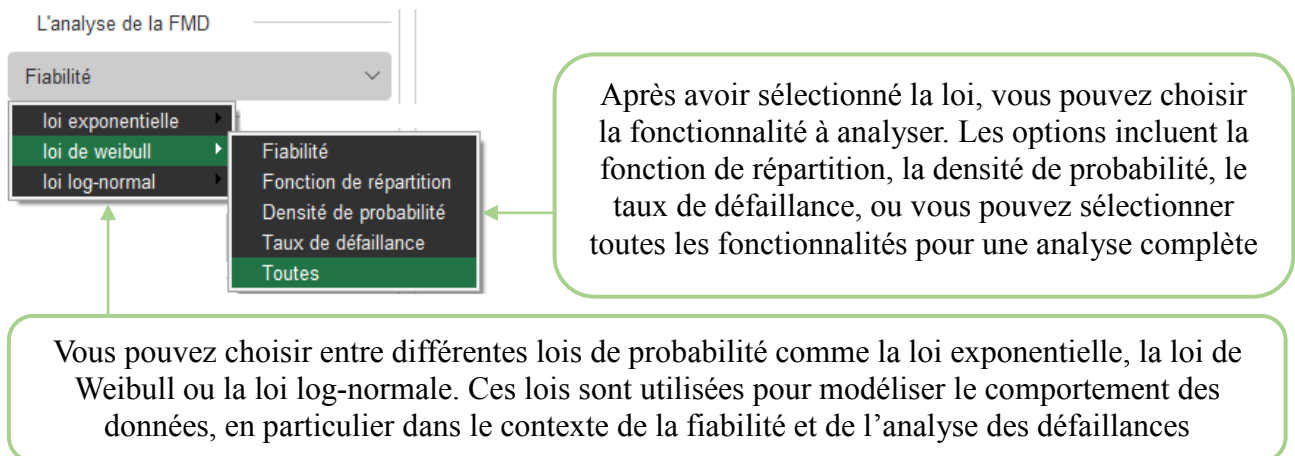


Fig III. 6 : Procédure de l'analyse

Une fois que vous avez sélectionné la loi de probabilité et la fonctionnalité, le programme va générer les courbes et les résultats de calcul correspondants.

Pour conserver ces informations, cliquez sur le bouton “Enregistrer”. Une boîte de dialogue peut apparaître vous demandant de choisir un emplacement et un format de fichier pour la sauvegarde.

III.5.3.1 l’analyse de la fiabilité de cette historique à l’aide de la loi de weibull

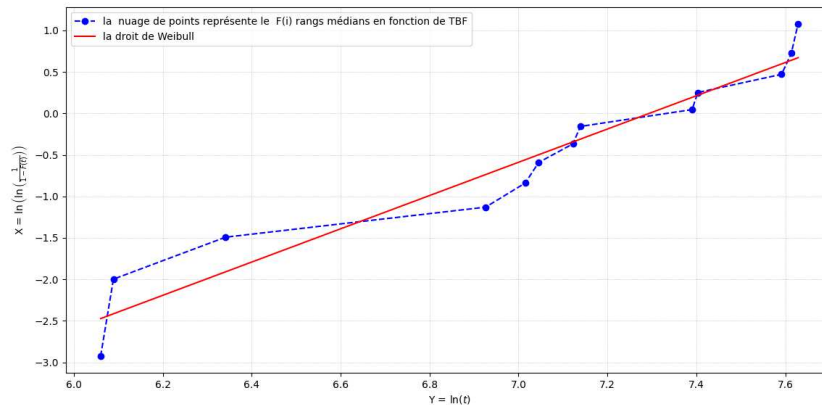


Fig III. 7 : Tracé de la droite de Weibull pour les TBFs

Les résultats :

TBF	i	F(i)	X	Y	F(t)	Dn	R(t)	f(t)	λ(t)
428.5	1.0	0.052239	6.060291	-2.925219	0.047584	0.004655	0.952416	0.000258	0.000271
441.0	2.0	0.126866	6.089045	-1.997558	0.050879	0.075987	0.949121	0.000269	0.000283
567.5	3.0	0.201493	6.341241	-1.491604	0.091641	0.109852	0.908359	0.000376	0.000414
1018.33	4.0	0.276119	6.925919	-1.129706	0.338081	0.061962	0.661919	0.000679	0.001026
1113.17	5.0	0.350746	7.014967	-0.839489	0.403796	0.05305	0.596204	0.000704	0.001181
1147.5	6.0	0.425373	7.045341	-0.590529	0.428048	0.002675	0.571952	0.000709	0.001239
1239.42	7.0	0.5	7.122399	-0.366513	0.493341	0.006659	0.506659	0.000709	0.0014
1260.17	8.0	0.574627	7.139002	-0.156901	0.508038	0.066589	0.491962	0.000707	0.001437
1620.25	9.0	0.649254	7.390336	0.046591	0.741107	0.091853	0.258893	0.000556	0.002147
1641.08	10.0	0.723881	7.40311	0.252254	0.752545	0.028664	0.247455	0.000542	0.002191
1981.42	11.0	0.798507	7.591569	0.471253	0.896865	0.098358	0.103135	0.000306	0.002967
2025.08	12.0	0.873134	7.613364	0.724948	0.909605	0.036471	0.090395	0.000278	0.003073
2056.42	13.0	0.947761	7.628722	1.082458	0.918003	0.029758	0.081997	0.000258	0.00315

Les résultats:
 Paramètre de forme β : 2.65 et A : 0.8893 et B : 0.355
 Paramètre d'échelle η: 1488.7428
 Paramètre de position γ: -47.6307
 la MTBF : 1276.31
 L'écart type σ :528.5
 R(MTBF) : 0.4806 , F(MTBF) : 0.5194 , f(MTBF) : 0.0007 , λ(MTBF) : 0.0015
 La valeur maximale de Dn,max 0.10985 est inférieure à Dn,alpha 0.36143.
 Alors, d'après le test de K-S avec un seuil d'alpha = 5 %, le module de Weibull est accepter.

Fig III. 8 : les résultats de calcul de la fiabilité à l’aide de loi de weibull afficher par le programme

Les graphes :

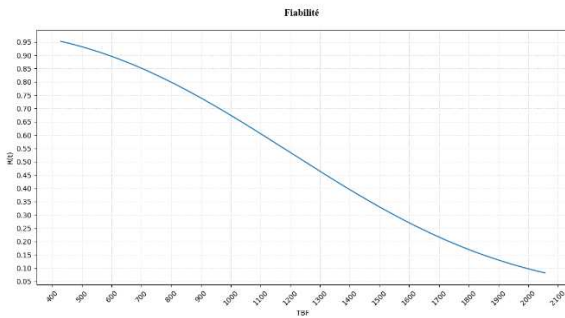


Fig III. 9 : Courbe de la fiabilité à l'aide de loi de weibull

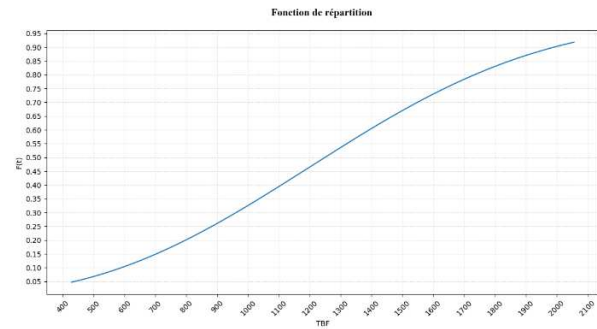


Fig III. 10 : Courbe de la fonction de répartition à l'aide de loi de weibull

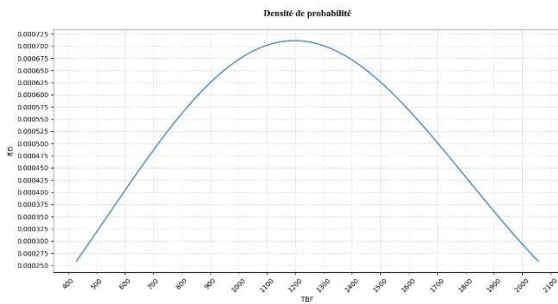


Fig III. 11 : Courbe de la densité de probabilité pour la fiabilité à l'aide de loi de weibull

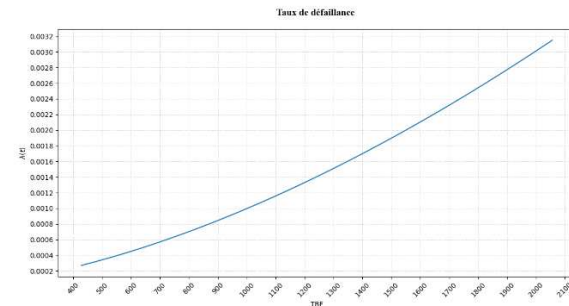


Fig III. 12 : courbe de la Taux de défaillance à l'aide de loi de weibull

III.5.3.2 l'analyse de la fiabilité de cette historique à l'aide de loi Log-normal :

TBF	i	F(i)	F(t)	Dn	f(t)	R(t)	λ(t)
428.5	1.0	0.052239	0.03323	0.019009	0.00033	0.96677	0.00034
441.0	2.0	0.126866	0.03747	0.089396	0.00035	0.96253	0.00036
567.5	3.0	0.201493	0.09637	0.105123	0.00057	0.90363	0.00063
1018.33	4.0	0.276119	0.42318	0.147061	0.00073	0.57682	0.00127
1113.17	5.0	0.350746	0.49007	0.139324	0.00068	0.50993	0.00133
1147.5	6.0	0.425373	0.51304	0.087667	0.00066	0.48696	0.00136
1239.42	7.0	0.5	0.57096	0.07096	0.0006	0.42904	0.0014
1260.17	8.0	0.574627	0.58329	0.008663	0.00059	0.41671	0.00142
1620.25	9.0	0.649254	0.75394	0.104686	0.00037	0.24606	0.0015
1641.08	10.0	0.723881	0.7615	0.037619	0.00036	0.2385	0.00151
1981.42	11.0	0.798507	0.85736	0.058853	0.00022	0.14264	0.00154
2025.08	12.0	0.873134	0.86647	0.006664	0.0002	0.13353	0.0015
2056.42	13.0	0.947761	0.87265	0.075111	0.00019	0.12735	0.00149

Les résultats:
 L'espérance μ : 7.0281 , L'écart type σ : 0.52733
 MTBF : 1296.13004
 R(MTBF) : 0.396 , F(MTBF) : 0.604 , f(MTBF) : 0.0006 , λ(MTBF) : 0.0015
 La valeur maximale de Dn,max 0.14706 est inférieure à Dn,alpha 0.36143.
 Alors, d'après le test de K-S avec un seuil d'alpha = 5 % , le module de loi log-normal est accepte.

Fig III. 13 : les résultats de calcul de la fiabilité à l'aide de loi log-normal afficher par le programme

Les graphes :

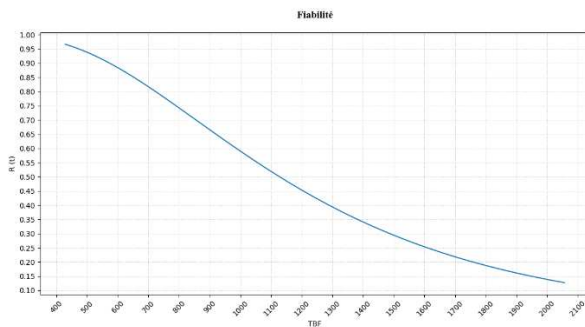


Fig III. 14 : Courbe de la fiabilité à l'aide de loi Log-normal

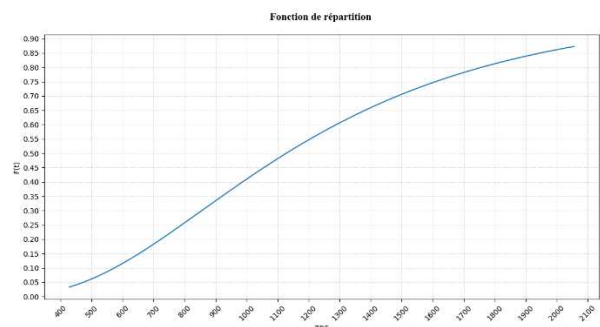


Fig III. 15 : courbe de la Taux de défaillance à l'aide de loi Log-normal

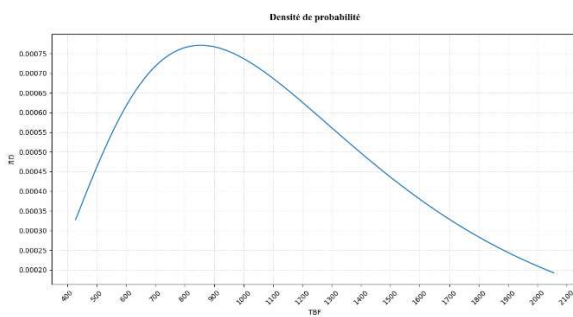


Fig III. 16 : Courbe de la densité de probabilité pour la fiabilité à l'aide de loi Log-normal

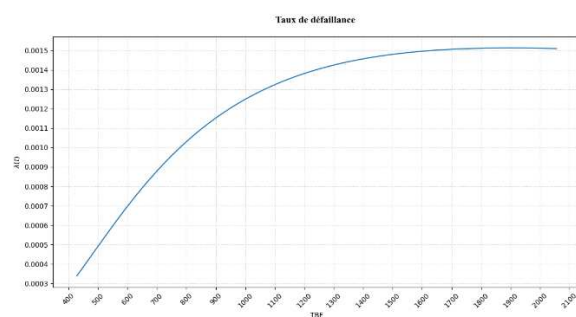


Fig III. 17 : courbe de la Taux de défaillance à l'aide de loi de Log-normal

III.5.3.3 L'analyse de la maintenabilité de cette historique à l'aide de la loi de weibull

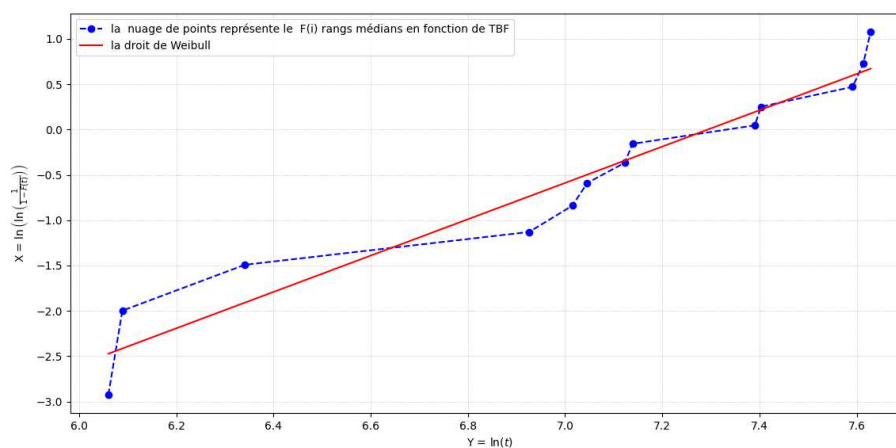


Fig III. 18 : Tracé de la droite de Weibull pour les TTRs

Les résultats :

TTR	i	M(i)	X	Y	M(t)	Dn	g(t)	$\mu(t)$
3.5	1.0	0.048611	1.252763	-2.999093	0.011736	0.036875	0.042995	0.043505
4.5	2.0	0.118056	1.504077	-2.07444	0.067001	0.051055	0.063548	0.068112
6.08	3.0	0.1875	1.805005	-1.571953	0.176632	0.010868	0.072798	0.088416
8.5	4.0	0.256944	2.140066	-1.214077	0.352111	0.095167	0.070205	0.10836
8.58	5.0	0.326389	2.149434	-0.92861	0.357717	0.031328	0.069948	0.108905
10.58	6.0	0.395833	2.358965	-0.685368	0.489898	0.094065	0.06177	0.121093
11.75	7.0	0.465278	2.463853	-0.468392	0.558897	0.093619	0.056125	0.127237
11.83	8.0	0.534722	2.470639	-0.267722	0.563371	0.028649	0.05573	0.127637
14.83	9.0	0.604167	2.696652	-0.076058	0.708432	0.104265	0.041149	0.141131
14.91	10.0	0.673611	2.702032	0.11303	0.711709	0.038098	0.040781	0.141458
14.91	11.0	0.743056	2.702032	0.306673	0.711709	0.031347	0.040781	0.141458
15.0	12.0	0.8125	2.70805	0.515202	0.715361	0.097139	0.040368	0.141823
16.67	13.0	0.881944	2.813611	0.759214	0.776616	0.105328	0.033128	0.148299
31.58	14.0	0.951389	3.452524	1.106549	0.982616	0.031227	0.003316	0.190761

Les résultats:

Paramètre de forme β : 1.339 et A : 0.917 et B : 0.687
 Paramètre d'échelle η : 10.0033
 Paramètre de position γ : 3.1366

la MTTR : 12.31
 L'écart type σ : 6.87
 M(MTTR) : 0.5895 , g(MTTR) : 0.0534 , μ (MTTR) : 0.13

La valeur maximale de Dn,max 0.10533 est inférieure à Dn,alpha 0.3489.
 Alors, d'après le test de K-S avec un seuil d'alpha = 5 % , le module de Weibull est accepter.

Fig III. 19 : les résultats de calcul de la maintenabilité à l'aide de loi de Weibull afficher par le programme

Les graphs :

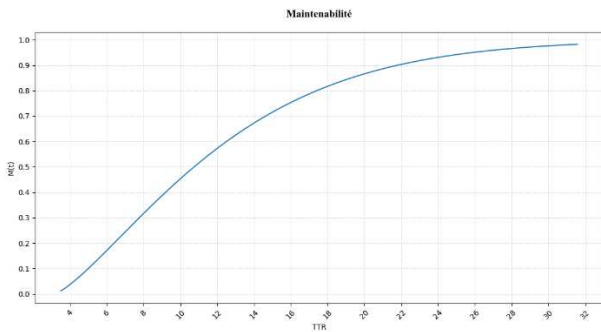


Fig III. 20 : Courbe de la maintenabilité à l'aide loi de weibull

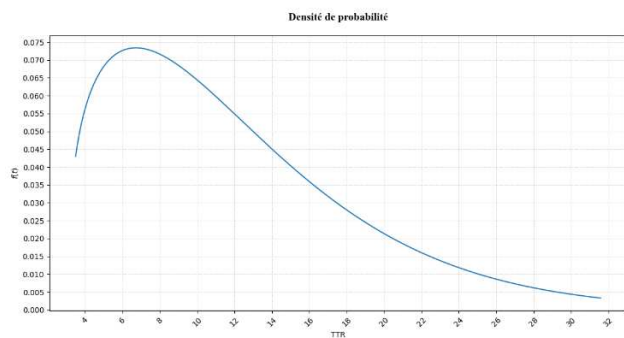


Fig III. 21 : Courbe densité de probabilité de la maintenabilité à l'aide de loi de weibull

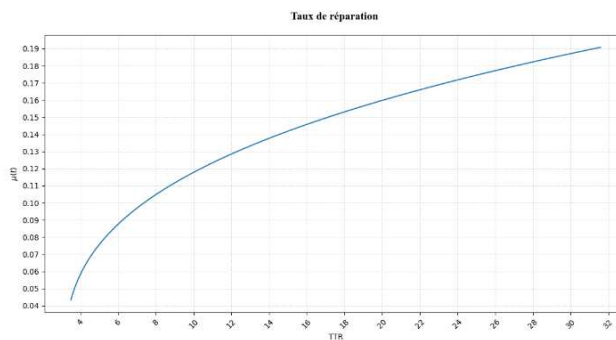


Fig III. 22 : Courbe de la taux de réparation à l'aide loi de weibull

III.5.3.4 L'analyse de la maintenabilité de cette historique à l'aide de la loi log-normal

TTR	i	F(i)	M(t)	Dn	g(t)	$\mu(t)$
3.5	1.0	0.048611	0.02101	0.027601	0.02618	0.02674
4.5	2.0	0.118056	0.05738	0.060676	0.0464	0.04922
6.08	3.0	0.1875	0.1513	0.0362	0.07002	0.0825
8.5	4.0	0.256944	0.3363	0.079356	0.07793	0.11742
8.58	5.0	0.326389	0.34253	0.016141	0.07775	0.11826
10.58	6.0	0.395833	0.48996	0.094127	0.06843	0.13417
11.75	7.0	0.465278	0.56562	0.100342	0.0608	0.13997
11.83	8.0	0.534722	0.57046	0.035738	0.06026	0.14029
14.83	9.0	0.604167	0.72169	0.117523	0.04109	0.14764
14.91	10.0	0.673611	0.72496	0.051349	0.04063	0.14772
14.91	11.0	0.743056	0.72496	0.018096	0.04063	0.14772
15.0	12.0	0.8125	0.72859	0.08391	0.04012	0.14782
16.67	13.0	0.881944	0.7882	0.093744	0.03154	0.14891
31.58	14.0	0.951389	0.97501	0.023621	0.00336	0.13445

Les résultats: _____
 L'espérance μ : 2.37284 , L'écart type σ : 0.55083
 MTTR : 12.48521
 M(MTTR) : 0.6085 , g(MTTR) : 0.0558 , μ (MTTR) : 0.1425
 La valeur maximale de Dn,max 0.11752 est inférieure à Dn,alpha 0.3489.
 Alors, d'après le test de K-S avec un seuil d'alpha = 5 % , le module de loi log-normal est accepte.

Fig III. 23 : les résultats de calcul de la maintenabilité afficher par la programme

Les graphes :

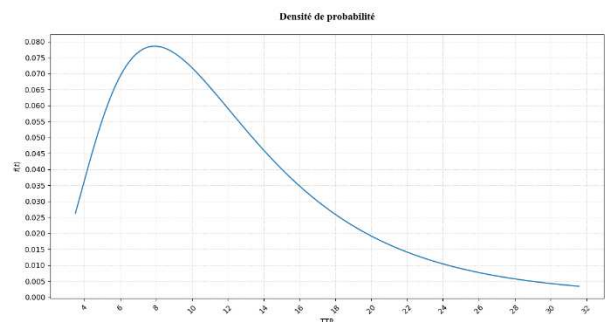
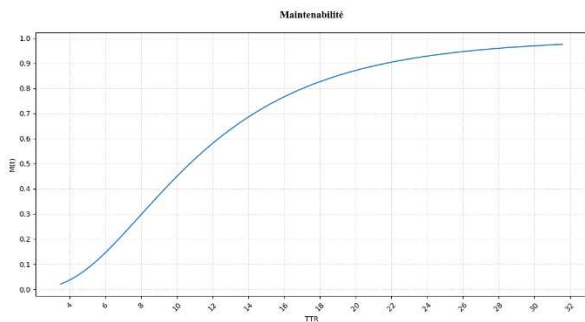


Fig III. 24 : Courbe de la maintenabilité à l'aide loi Log-normal

Fig III. 25 : Courbe densité de probabilité à l'aide de loi log-normal

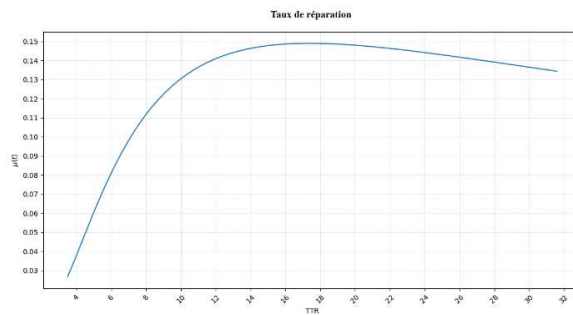


Fig III. 26 : Courbe de la taux de réparation à l'aide loi log-normal

III.5.3.5 L'analyse de la disponibilité de cette historique à l'aide de la loi exponentielle :

TTR	Di(t)
3.5	0.9976
4.5	0.997
6.08	0.9962
8.5	0.9951
8.58	0.9951
10.58	0.9943
11.75	0.994
11.83	0.9939
14.83	0.9931
14.91	0.9931
14.91	0.9931
15.0	0.9931
16.67	0.9927
31.58	0.9909

Les résultats:

MTBF = 1272.295 $\lambda = 0.0008$

MTTR = 12.373 $\mu = 0.0808$

La disponibilité intrinsèque Di = 0.9904

Fig III. 27 : les résultats de calcul de la Disponibilité afficher par la programme

Le graphe :

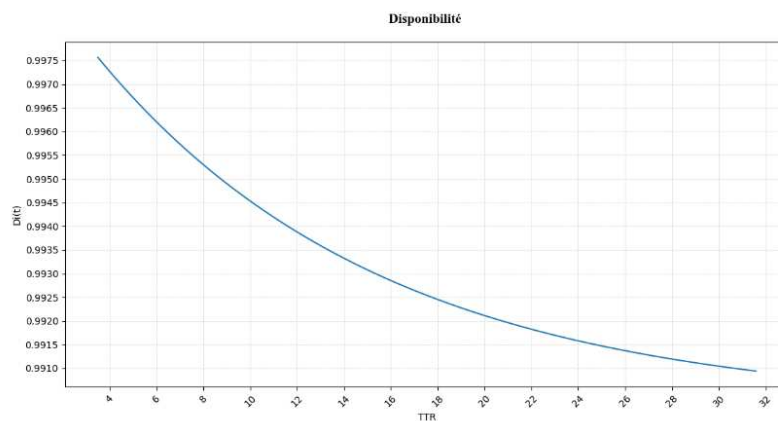


Fig III. 28 : Courbe de la disponibilité

Lorsque vous évaluez la fiabilité, la maintenabilité et la disponibilité de vos systèmes, vous pouvez identifier les domaines à améliorer

En utilisant ces analyses, vous serez en mesure de prendre des décisions éclairées pour optimiser la performance de vos équipements et assurer leur bon fonctionnement et aussi vont vous permettre d'optimiser

la performance de vos équipements, de minimiser les temps d'arrêt coûteux et d'assurer une utilisation efficace des ressources.

Conclusion :

La mise en œuvre des modèles statistiques tels que les distributions exponentielles, de Weibull et log normale dans le programme de gestion de la maintenance a démontré une capacité significative à analyser la fiabilité, la maintenance et la disponibilité des équipements. L'utilisation de Python pour développer ces modèles a permis une analyse approfondie et la visualisation des données, facilitant ainsi la prise de décisions éclairées en matière de maintenance préventive. Ce rapport a détaillé le processus de développement, soulignant l'importance d'une approche structurée et méthodique pour atteindre une efficacité optimale dans la gestion de la maintenance.

Conclusion générale
et
Perspectives

Conclusion générale

L'analyse FMD et la gestion de la maintenance s'avèrent essentielles pour garantir le bon fonctionnement des opérations industrielles. En combinant les principes fondamentaux de la maintenance avec les puissantes techniques d'analyse probabiliste, ce mémoire a démontré l'efficacité de l'analyse FMD dans l'optimisation des stratégies de maintenance et l'amélioration de la performance globale des actifs.

Ce programme développé en Python constitue un outil précieux pour les praticiens de la maintenance. En exploitant la puissance des bibliothèques spécialisées de Python, ce programme offre une analyse FMD robuste et complète, permettant aux utilisateurs de :

- Comprendre le comportement des équipements et identifier les défaillances potentielles.
- Évaluer la fiabilité, la maintenabilité et la disponibilité des actifs.
- Définir des stratégies de maintenance proactives et optimisées.
- Réduire les temps d'arrêt non planifiés et les coûts de maintenance.

Afin de valider la fiabilité et l'exactitude du programme FMD, une série de tests de validation rigoureux a été effectuée sur un échantillon de données historiques. Cet échantillon provenait d'analyses FMD réalisées par des étudiants antérieurs à l'aide du logiciel Minitab. Les paramètres de Weibull obtenus à partir du notre programme ont été comparés aux paramètres déterminés par Minitab, et les résultats ont montré une concordance remarquable. Cette validation confirme la précision et la fiabilité du programme FMD, comme en témoignent les résultats détaillés dans l'Annexe I.

De plus, le programme FMD a été validé par le Dr. DEFDAF MABRAK, un expert reconnu dans le domaine de l'analyse FMD et de la gestion de la maintenance. Le Dr. MABRAK a examiné le programme en détail et a conclu qu'il s'agissait d'un outil précieux pour les praticiens de la maintenance.

Il est important de noter que d'autres logiciels proposent des fonctionnalités similaires pour l'analyse FMD, comme Fiabobtim, Loglaala...etc. Cependant, le programme 317_FMD développé dans ce mémoire se distingue par plusieurs avantages clés :

- Large éventail d'options d'analyse : Le programme offre une gamme complète d'options d'analyse FMD, permettant aux utilisateurs de personnaliser leur analyse en fonction de leurs besoins spécifiques.
- Accès rapide aux données : Le programme offre un accès rapide et efficace aux données, ce qui permet aux utilisateurs de gagner du temps et d'améliorer leur productivité.
- Enregistrement des résultats de calcul : Le programme permet d'enregistrer facilement les résultats de calcul, ce qui facilite le partage et la consultation des données

En conclusion, l'analyse FMD et la gestion de la maintenance constituent des piliers essentiels pour les entreprises industrielles qui cherchent à atteindre l'excellence opérationnelle. Le programme FMD développé dans ce mémoire offre un outil précieux pour concrétiser ces objectifs, en permettant aux praticiens de la maintenance de prendre des décisions éclairées, d'optimiser les stratégies de maintenance et de maximiser la performance des actifs.

Perspective

Ce mémoire ouvre également la voie à de futures recherches dans le domaine de l'analyse FMD et de la gestion de la maintenance. L'exploration de techniques d'analyse plus sophistiquées, l'intégration de l'apprentissage automatique et l'extension du programme à d'autres domaines d'application ne sont que quelques exemples des avenues prometteuses qui méritent d'être explorées.

En fin de compte, l'objectif ultime est de développer des outils et des méthodologies toujours plus efficaces pour optimiser la gestion de la maintenance et garantir le bon fonctionnement des opérations industrielles, contribuant ainsi à la réussite et à la prospérité des entreprises.

Référence :

- [1] : AFNOR, NF X10-100 : Gestion des actifs physiques - Vocabulaire.
- [2] : Smith, J. R. (2020). Maintenance Management in Industrial Operations. *Journal of Industrial Engineering*, 42(3), 123-137.
- [3] : NF X60-010 , Analyse de la fiabilité – Vocabulaire
- [4] : The Oxford Talking Dictionary, The Learning Company, 1998
- [5] : Zeineb Ben Houria. Optimisation de la gestion du service de maintenance biomédicale. Ingénierie biomédicale. Université de Lyon; Université de Sfax (Tunisie), 2016. Français
- [6] : Anthony Kelly, Maintenance and the industrial organization. *Plant Maintenance Management Set 2006 ; 3(1) : 3-8.*
- [7] : Romain Lesobre, Modélisation et optimisation de la maintenance et de la surveillance des Systèmes multi- composants- Applications à la maintenance et à la conception de véhicules industriels 196p, 2015.
- [8] : Mme Benaïcha Halima, « Analyse des stratégies de maintenance des systèmes de production industrielle », Thèse En Vue De L'obtention Du Diplôme De Doctorat En-science, Université des Sciences et de la Technologie d'Oran Mohammed Boudiaf, 2015.
- [9] : Nakajima S., La maintenance productive totale (TPM), nouvelle vague de la production industrielle, Afnor Gestion, 1987.
- [10] : Alin Gabriel Mihalache. Modélisation et évaluation de la fiabilité des systèmes mécatroniques : application sur système embarqué. *Sciences de l'ingénieur [physics]*. Université d'Angers, 2007. Français
- [11] : François Monchy Jean-Pierre Vernier Maintenance Méthodes et Organisations 3^e édition
- [12] : La loi log-normale dans l'analyse de fiabilité. support.minitab.com
- [13] : Donald E. Knuth, The Art of Computer Programming, vol. 2, 3^e éd., Addison-Wesley Professional, 784 p. (ISBN 0-201-89684-2), p. 48–55
- [14] : <https://docs.python.org/>

[15] : <https://numpy.org/doc/>

[16] : <https://docs.scipy.org/doc/scipy/>

[17] : <https://pandas.pydata.org/docs/>

[18] : <https://matplotlib.org/>

Résumé :

Ce mémoire offre une analyse complète et éclairante de l'aspect crucial de la gestion de la maintenance que constitue l'analyse de la fiabilité, de la maintenabilité et de la disponibilité (FMD). En combinant des concepts théoriques avec des applications pratiques.

Ce mémoire a pour objectif de développer un programme informatique performant dédié à l'analyse FMD. Ce programme a pour but d'assister les étudiants et chercheurs, techniciens et gestionnaires dans le domaine de la gestion efficace de la maintenance des équipements.

Summary :

This dissertation provides a comprehensive and insightful analysis of the crucial aspect of maintenance management that is reliability, maintainability and availability (RMA) analysis. By combining theoretical concepts with practical applications.

This dissertation aims to develop a high-performance computer program dedicated to RMA analysis. This program aims to assist students and researchers, technicians and managers in the field of effective equipment maintenance management.

ملخص

توفر هذه الأطروحة تحليلاً شاملاً ومتعمقاً للجانب الحاسم لإدارة الصيانة وهو تحليل الموثوقية وقابلية الصيانة والتوافر (م ص ت). من خلال الجمع بين المفاهيم النظرية والتطبيقات العملية.

تهدف هذه الأطروحة إلى تطوير برنامج حاسوبي عالي الأداء مخصص لتحليل (م ص ت). يهدف هذا البرنامج إلى مساعدة الطلاب والباحثين والفنيين والمديرين في مجال الإدارة الفعالة لصيانة المعدات.

Annexe I :

Les résultats réalisés par les anciens étudiants à l'aide de logiciel Minitab :

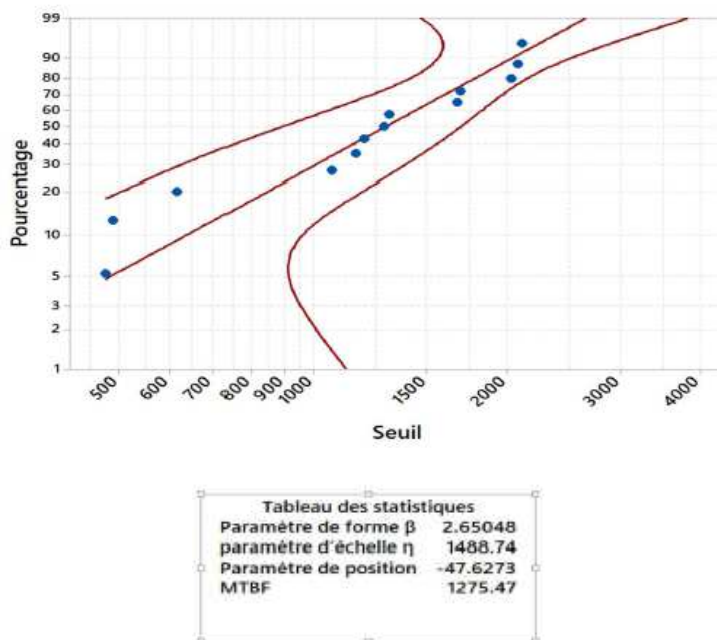


Figure 1 : Estimation des paramètres de loi de weibull

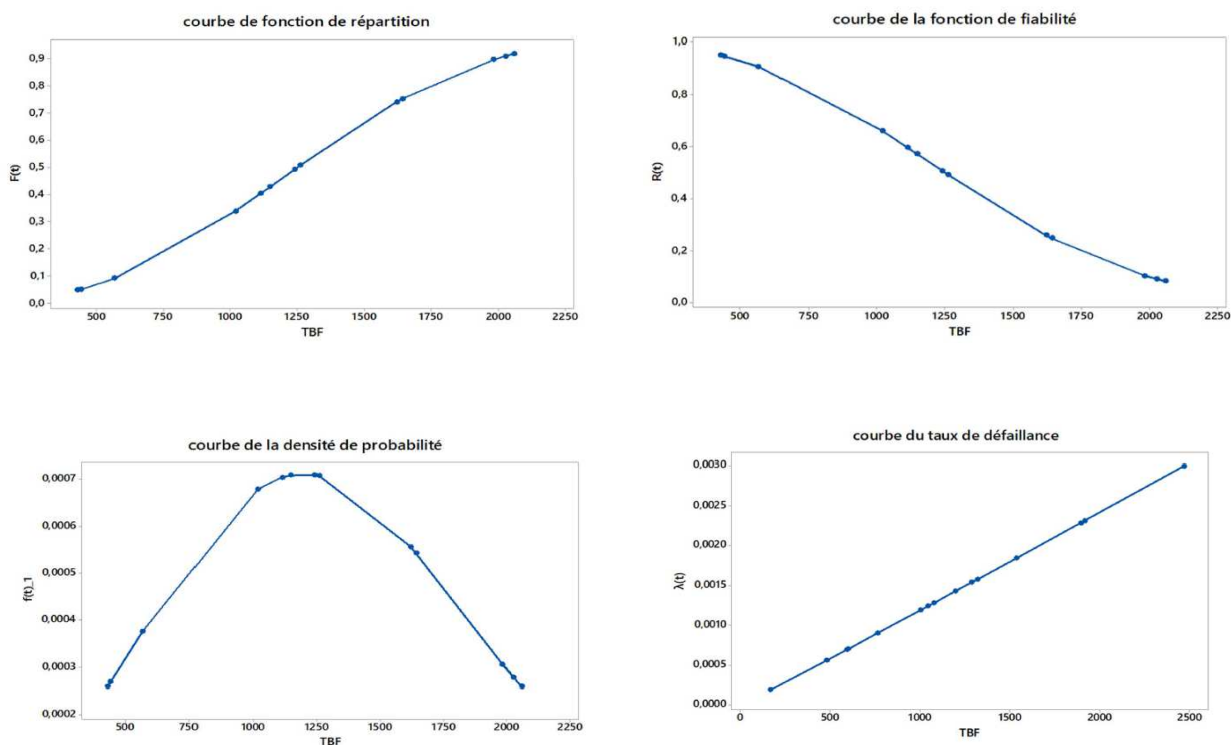


Figure 2 : Les graphes présentent les fonction de la fiabilité