

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire



وزارة التعليم العالي
والبحوث العلمي
Ministère de l'Enseignement Supérieure
et de la Recherche Scientifique



Université d'Oran 2
Institut de Maintenance et de Sécurité Industrielle
Département de Maintenance en électromécanique

Thèse

Pour l'obtention du diplôme de Doctorat L.M.D
Doctorat Électromécanique

Intitulé

Contribution à la mise en œuvre et surveillance d'une cellule robotisée de parachèvement de pièces

Présentée et soutenue publiquement par :

Zohra MEHAR

Devant le jury composé de :

Nom et Prénom	Grade	Établissement	Qualité
Mme. Djilalia GUENDOZ	Professeur	Université d'Oran 2	Présidente
Mr. Rachid NOUREDDINE	Professeur	Université d'Oran 2	Directeur de thèse
Mr. Farid NOUREDDINE	Maître de Conférences HDR	ENIT – UTTOP (France)	Co-Directeur de thèse
Mr. Reda YAHIAOUI	Maître de Conférences A	Université d'Oran 2	Examineur
Mme. Hadria FIZAZI	Professeur	USTO-MB	Examinatrice
Mr. Tawfik BENABDALLAH	Professeur	ENPO-MA	Examineur

21 Janvier 2025

Dédicace

Je dédie cette thèse à

...

À ma très chère mère Amied Ghania

Tu représentes pour moi le symbole de bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a cessé de m'encourager et de prier pour moi. Ta prière et ta bénédiction m'ont été d'un grand secours pour mener à bien mes études. Je te dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout-puissant, te préserver et t'accorder la santé, une longue vie et bonheur.

À mon très cher frère Haithem

Mon cher frère qui m'est le père, les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour vous. Mon ange gardien et mon fidèle compagnon dans les moments les plus délicats de cette vie mystérieuse. Je vous dédie ce travail avec tous mes vœux de bonheur, de santé et de réussite.

À mon très cher frère Assil

Mon cher petit frère présent dans tous mes moments d'examens par son soutien moral et ses belles surprises sucrées. Je te souhaite un avenir plein de joie, de bonheur, de réussite et de sérénité. Je t'exprime à travers ce travail mes sentiments de fraternité et d'amour.

À tous ceux qui, par un mot, m'ont donné la force de continuer ...

Zohra MEHAR

Remerciements

Cette thèse marque la fin des années de recherche au sein du Laboratoire Génie de Production et Maintenance Industrielle (LGPMI) de l'Institut de Maintenance et de Sécurité Industrielle (IMSI), à l'université d'Oran 2 Mohamed Ben Ahmed (UO2/MBA). Cette thèse a été pour moi une très grande expérience.

Tous mes remerciements s'adressent tout d'abord au tout puissant *Allah*, d'avoir guidé mes pas vers le chemin du savoir, et de m'avoir donné le courage, la volonté et la patience de mener à terme ce présent projet de recherche.

Je remercie très chaleureusement mes deux directeurs de thèse, *Pr. Rachid Noureddine*, Directeur du LGPMI, Professeur à l'IMSI de l'université d'Oran 2 Mohamed Ben Ahmed et *Dr. Farid Noureddine*, Maître de Conférences Habilité à Diriger des Recherches du Laboratoire Génie de Production (LGP) de l'École Nationale d'Ingénieurs de Tarbes (ENIT-UTTOP), pour avoir assuré l'encadrement de cette thèse. Leurs compétences complémentaires m'ont permis de mener à bien ce projet de recherche. Merci pour leur disponibilité, leurs précieux conseils et pour m'avoir guidée dans chacune des étapes de cette thèse, et pour m'avoir encouragée et m'avoir donné l'envie de devenir enseignant-chercheur.

Je tiens aussi à remercier chaleureusement les *Membres du Jury (Mme Djilalia Guendouz, Mr. Reda Yahiaoui, Mme Hadria Fizazi, Mr. Tawfik Benabdallah)* de thèse qui ont gracieusement consacré leur temps à la révision de ce travail et qui m'ont également honoré par leur acceptation de l'évaluer, et surtout, de m'avoir donné l'opportunité de bénéficier de leur expertise pour progresser et développer mes perspectives.

Mes remerciements s'adressent aussi à *Mme Annie Noureddine*, Enseignante associée à l'ENIT pour son aide et sa contribution à ce projet de recherche.

Merci à l'ensemble des personnes du LGP-ENIT pour leur accueil et leur disponibilité ; Merci aussi à l'ensemble des personnes du LGPMI-IMSI-UO2/MBA.

Je tiens à remercier ceux qui m'entourent depuis toujours, ma mère *Ghania* et mes frères *Haithem* et *Assil* pour leur soutien et m'avoir encouragée au cours de ces années de recherche.

Enfin, je remercie tous mes collègues ; les docteurs et les doctorants, qui ont su m'apporter leur soutien tout au long de ces années et me changer les idées lorsque j'en avais besoin.

À vous tous, un grand MERCI !

« Ce travail est un cadeau à ma mère *Ghania AMIED* »

Contribution à la mise en œuvre et Surveillance d'une Cellule Robotisée de Parachèvement de Pièces

Résumé

Le parachèvement des pièces regroupe toutes les opérations de finition et notamment les opérations de découpage, sciage, perçage, détournage, surfaçage, voire aussi des opérations de meulage, ponçage, ébarbage, et polissage. Une robotisation de ces opérations est maintenant envisageable. Cette transition vers l'industrie 4.0 soulève cependant des défis liés à la fiabilité et à la maintenance des systèmes robotiques. Cette thèse propose une nouvelle approche pour améliorer la surveillance des cellules robotisées d'usinage. Cette approche est basée sur l'utilisation d'un Réseau de Petri implémenté à partir de l'arbre des fautes de la cellule robotisée d'usinage étudiée et, permet de simuler différents scénarios de défaillances et d'identifier les symptômes associés. Cette démarche permet de détecter rapidement les défaillances et minimiser les temps d'arrêt et d'optimiser la disponibilité de la cellule robotisée. Les résultats obtenus, interprétés et discutés, permettent d'envisager une suite encourageante.

Mots Clefs

Arbre des fautes, Cellule robotisée d'usinage, Détection, Diagnostic, Fautes, Parachèvement de pièces, Réseau de Petri, Surveillance.

Contribution to the Implementation and Monitoring of a Robotic Cell for Part Finishing

Abstract

Finishing includes all finishing operations, including cutting, sawing, drilling, trimming, surfacing, and even grinding, sanding, deburring, and polishing operations. The automation of these operations through robotics is now feasible. However, this transition to Industry 4.0 presents challenges related to the reliability and maintenance of robotic systems. This thesis proposes a new approach to improve the monitoring of machining robot cells. This approach is based on the Petri net implemented from the fault tree of the machining robot cell studied, enabling the simulation of different failure scenarios and identification of associated symptoms. This methodology allows the rapid detection of failures, minimizing downtime, and optimizing the availability of the robotic cell. The results obtained, interpreted, discussed, suggest promising future developments.

Keywords

Fault Tree, Fault Detection and Diagnosis, Machining Robot Cell, Monitoring, Part finishing, Petri Net, Robotic machining.

المساهمة في إنجاز ومراقبة خلية روبوتية لاستكمال القطع

Résumé (Arabe)

ملخص

يتضمن استكمال القطع جميع العمليات النهائية، بما في ذلك عمليات التقطيع، النشر، التثقيب، التوجيه، التسوية، بالإضافة إلى عمليات الطحن والصفرة وإزالة النتوءات والتلميع. أصبح من الممكن الآن إنجاز هذه العمليات بواسطة الروبوتات. ومع ذلك، فإن هذا التحول إلى الصناعة 4.0 يثير تحديات تتعلق بموثوقية وصيانة الأنظمة الروبوتية. تقترح هذه الأطروحة نهجاً جديداً لتحسين مراقبة خلايا التصنيع الروبوتية. يعتمد هذا النهج على استخدام شبكة بتري التي تم تنفيذها من شجرة الأخطاء في خلية التصنيع الروبوتية المدروسة وتسمح بمحاكاة سيناريوهات الأعطال المختلفة وتحديد الأعراض المرتبطة بها. يتيح هذا النهج الكشف السريع عن الأعطال، بتقليل أوقات التوقف وتحسين توافر الخلية الروبوتية. إن النتائج التي حصلنا عليها، والتي تم تفسيرها ومناقشتها، تسمح لنا بتصور استمرار مشجع.

الكلمات المفتاحية

شجرة الأعطال، خلية التصنيع الروبوتية، الكشف، التشخيص، الأعطال، استكمال القطع، شبكة بتري، المراقبة.

Table des Matières

Résumé	I
Abstract	II
Résumé (Arabe).....	II
Table des Matières	III
Liste des Figures.....	IX
Liste des Tableaux.....	X
Abréviations	XI
Nomenclature	XIII
Introduction Générale.....	1
Chapitre 1 : Usinage et Parachèvement robotisés	3
1.1 Introduction	4
1.2 Robots industriels	4
1.3 Usinage robotisé	5
1.4 Architectures des robots d'usinage.....	5
1.4.1 Architecture sérielle	5
1.4.2 Architecture parallèle.....	6
1.4.3 Architecture hybride	6
1.4.4 Avantages et Inconvénients des robots sériels et parallèles.....	7
1.4.5 Avantages et Inconvénients du robot hybride.....	8
1.5 Critères de performance des robots industriels.....	9
1.6 Opérations de parachèvement.....	9
1.6.1 Découpage.....	10
1.6.2 Sciage	10
1.6.3 Perçage	10
1.6.4 Détourage	11
1.6.5 Surfaçage.....	11
1.6.6 Meulage.....	12
1.6.7 Ponçage	12
1.6.8 Ébarbage	12
1.6.9 Polissage	13
1.7 Conclusion	13

Table des Matières

Chapitre 2 : Surveillance Industrielle	14
2.1 Introduction	16
2.2 Notions de base et définitions utilisées en surveillance	16
2.2.1 Mode de fonctionnement	16
2.2.2 Dégradation	16
2.2.3 Défaillance	16
2.2.4 Panne	16
2.2.5 Défaut	17
2.2.6 Risque	17
2.2.7 Erreur	17
2.2.8 Faute	17
2.2.9 Tolérance aux fautes	17
2.2.10 Reconfiguration	17
2.2.11 Sûreté de fonctionnement	17
2.2.11.1 Fiabilité	18
2.2.11.2 Maintenabilité	18
2.2.11.3 Disponibilité	18
2.2.11.4 Sécurité	18
2.2.12 Maintenance industrielle	18
2.2.12.1 Maintenance Corrective	19
2.2.12.1.1 Maintenance palliative	20
2.2.12.1.2 Maintenance curative	20
2.2.12.2 Maintenance Préventive	20
2.2.12.2.1 Maintenance systématique	20
2.2.12.2.2 Maintenance conditionnelle	21
2.2.12.2.3 Maintenance prédictive	21
2.2.12.2.4 Maintenance proactive	22
2.2.12.3 Autre classification des activités de maintenance	22
2.3 Surveillance industrielle	22
2.3.1 Détection	23
2.3.2 Diagnostic	23
2.4 Classification des Méthodes et Modèles de Surveillance	23
2.4.1 Méthodes de Surveillance avec Modèles	24
2.4.1.1 Méthodes par Modélisations Fonctionnelle et Matérielle	24
2.4.1.1.1 Analyse des Modes de Défaillance, de leurs effets et de leur Criticité (AMDEC) ..	24
2.4.1.1.2 Arbre de défaillance (AdD) / Arbre des Fautes (AdF)	25
2.4.1.1.3 Analyse Préliminaire des Risques (APR)	26

Table des Matières

2.4.1.1.4	Diagramme de Fiabilité (DF)	26
2.4.1.1.5	Méthode de l'Espace des États (MEE)	26
2.4.1.1.6	Réseaux de Petri (RdP)	27
2.4.1.1.7	HAZard and Operability Study (HAZOP)	27
2.4.1.1.8	Arbre d'Évènements (AE)	27
2.4.1.1.9	Méthode du Diagramme Causes-Conséquences (DCC)	28
2.4.1.1.10	Méthodes de la Table de Vérité (TV) et de la Table de Décision (TD)	28
2.4.1.2	Méthodes de Surveillance par Modélisation Physique	28
2.4.1.2.1	Redondances Physiques et Analytiques	29
2.4.1.2.2	Estimation Paramétrique	29
2.4.2	Méthodes de Surveillance sans Modèles	29
2.4.2.1	Surveillance basée sur des Outils Statistiques du Traitement du Signal	29
2.4.2.1.1	Test de Franchissement de Seuil	30
2.4.2.1.2	Test de Moyenne	30
2.4.2.1.3	Test de Variance	30
2.4.2.2	Surveillance par Intelligence Artificielle (IA)	30
2.4.2.2.1	Modèles Comportementaux	30
2.4.2.2.1.1	Automates d'États Finis (AEF)	31
2.4.2.2.1.2	Réseaux de Petri (RdP)	31
2.4.2.2.2	Méthodes de Reconnaissance de Formes	31
2.4.2.2.2.1	Système Expert	31
2.4.2.2.2.2	Outil Statistique	32
2.4.2.2.2.3	Raisonnement à Partir de Cas (RàPC)	32
2.4.2.2.2.4	Réseaux de neuronaux (RNA)	32
2.4.2.2.2.5	Logique Floue	33
2.4.2.2.2.6	Système Neuro-Flou	33
2.4.2.2.3	Méthodes Explicatives	34
2.4.2.2.3.1	Graphes Causaux	34
2.4.2.2.3.2	Graphes Contextuels	34
2.5	Conclusion	35
Chapitre 3 : Théorie des Réseaux de Petri		36
3.1	Introduction	38
3.2	Historique	38
3.3	État de l'art RdP et Surveillance Industrielle	38
3.4	Réseaux de Petri	40
3.5	Notions de base des réseaux de Petri	40
3.6	Propriétés Mathématiques	42

Table des Matières

3.7	Propriétés Génériques.....	44
3.7.1	RdP borné.....	44
3.7.2	RdP sauf.....	45
3.7.3	Vivacité.....	45
3.7.4	Blocage.....	45
3.7.5	Réinitialisabilité.....	45
3.7.6	Conflit.....	45
3.7.7	Accessibilité.....	45
3.7.8	Sûreté.....	46
3.8	Analyse des réseaux de Petri.....	46
3.8.1	Analyse par Graphes des Marquages.....	46
3.8.2	Analyse par Arbre de Couverture et Graphe de Couverture.....	46
3.8.3	Analyse par Algèbre Linéaire.....	47
3.8.4	Analyse par Réduction.....	47
3.9	Structures particulières des RdP.....	47
3.9.1	Graphe d'États.....	48
3.9.2	Graphe d'Événement.....	48
3.9.3	RdP sans Conflit.....	48
3.9.4	RdP Conflit Structurel.....	48
3.9.5	RdP à Choix Libre.....	49
3.9.6	RdP Simple.....	49
3.9.7	RdP Pur.....	49
3.9.8	RdP sans Boucle.....	49
3.9.9	RdP Généralisé.....	49
3.9.10	RdP à Capacité.....	49
3.9.11	RdP Autonome.....	50
3.9.12	RdP non Autonome.....	50
3.9.13	RdP à Priorité.....	50
3.9.14	RdP à Arcs Inhibiteurs.....	50
3.10	Avantages et inconvénients des réseaux de Petri généralisés et particuliers.....	51
3.10.1	Avantages.....	51
3.10.2	Inconvénients.....	51
3.11	Réseaux de Petri de Haut Niveau.....	52
3.11.1	Réseaux de Petri Synchronisés (RdPS).....	52
3.11.2	Réseaux de Petri Temporisés (RdPT).....	52
3.11.2.1	Réseaux de Petri Temporisés.....	52
3.11.2.1.1	Réseaux de Petri T-temporisés (RdPTt).....	52

Table des Matières

3.11.2.1.2	Réseaux de Petri P-temporisés (RdPPt).....	53
3.11.2.2	Réseaux de Petri Temporels.....	53
3.11.2.2.1	Réseaux de Petri T-temporels.....	53
3.11.2.2.2	Réseaux de Petri P-temporels.....	53
3.11.2.2.3	Réseaux de Petri arc-pt-temporels.....	54
3.11.3	Réseaux de Petri Stochastiques (SPN).....	54
3.11.3.1	Réseaux de Petri Stochastiques Généralisés (GSPN).....	55
3.11.3.2	Réseaux de Petri Stochastiques et Déterministes (DSPN).....	55
3.11.3.3	Réseaux de Petri Stochastiques Étendus (ESPN).....	55
3.11.4	Réseaux de Petri Colorés (RCP).....	55
3.11.5	Réseaux de Petri Interprétés (RdPI).....	55
3.11.6	Réseaux de Petri Algébriques (APN).....	56
3.11.7	Réseaux de Petri Flous (RdPF).....	56
3.11.7.1	RdPF basés sur des règles floues.....	57
3.11.7.2	RdPF basés sur la logique linéaire floue.....	57
3.11.8	Réseaux de Petri Hiérarchiques (HPN).....	57
3.11.9	Réseaux de Petri Continus (RdPC).....	57
3.11.10	Réseaux de Petri Hybrides (RdPH).....	58
3.12	Conclusion.....	58
Chapitre 4 : Applications – Usinage robotisé en parachèvement.....		59
4.1	Introduction.....	60
4.2	Problématique.....	60
4.3	Transfert AdF - RdP.....	62
4.4	Étude de cas 1 - RdP.....	63
4.4.1	Analyse par Arbre des Fautes.....	63
4.4.2	Implémentation RdP - LabView.....	64
4.4.3	Simulations et résultats.....	66
4.5	Étude de cas 2 - RdP λ	67
4.5.1	Formalisation RdP λ	67
4.5.2	Paramétrage de la cellule d'usinage de parachèvement.....	67
4.5.3	Analyse par AdF.....	69
4.5.3.1	Analyse par RdP λ	70
4.5.4	Implémentation LabView.....	73
4.5.4.1	Implémentation AdF – LabView.....	73
4.5.4.2	Implémentation RdP λ – LabView.....	74
4.5.5	Simulations et résultats.....	76
4.5.5.1	Taux de défaillance.....	76

Table des Matières

4.5.5.2	Simulation LabView	76
4.5.5.3	Marquage	78
4.5.5.4	Propriétés génériques du RdP λ	79
4.5.5.5	Matrice d'incidence Avant du RdP λ	79
4.5.5.6	Matrice d'incidence Arrière du RdP λ	79
4.5.5.7	Matrice d'incidence du RdP λ	79
4.5.5.8	Vecteur de marquage du RdP λ	80
4.6	Conclusion	81
	Conclusion Générale	83
	Bibliographie	85
	Publications	92
	Petri Net Modeling of a Machining Robot Cell	93
	Diagnostic by Fault Tree and Petri Nets of a Robotic Machining Cell	109

Liste des Figures

Figure 1. 1. Robot sériel anthropomorphe, Robot KUKA KR 120	5
Figure 1. 2. Robot sériel cartésien, (Barka & Bentrar, 2018).....	6
Figure 1. 3. Robot parallèle, Robot Delta,(Paccot, 2009)	6
Figure 1. 4. Robot sériel intégrant un mécanisme parallèle, Robot ABB IRB 6660, (Dumas, 2011).....	7
Figure 1. 5. Robot parallèle intégrant un mécanisme sériel, Robot Tricept T606, (Moussavi, 2016).....	7
Figure 2. 1. Architecture globale et différentes stratégies de maintenance.....	19
Figure 2. 2. Classification des méthodes et modèles de surveillance.....	24
Figure 3. 1. Structure Correcte et incorrecte d'un RdP	41
Figure 3. 2. Exemple RdP 1	42
Figure 3. 3. RdP quadruplet {P, T, Pré, Post}	43
Figure 3. 4. Exemple RdP 2	44
Figure 3. 5. Exemple RdP 3	47
Figure 3. 6. Graphe d'états	48
Figure 3. 7. Graphe d'événement	48
Figure 3. 8. Conflit structurel	49
Figure 3. 9. RdP à capacité.....	50
Figure 3. 10. Exemple de RdP avec arc inhibiteur	51
Figure 4. 1. Schéma fonctionnel du système de surveillance.....	60
Figure 4. 2. Étude de Cas 1 : Cellule robotisée d'usinage de parachèvement de pièces parallélépipédiques.....	61
Figure 4. 3. Étude de Cas 2 : Cellule robotisée d'usinage de parachèvement de pièces cylindriques	61
Figure 4. 4. Transformation de la porte logique « ET » de l'AdF en RdP	62
Figure 4. 5. Transformation de la porte logique « OU » de l'AdF en RdP	63
Figure 4. 6. Arbre des Fautes de la cellule robotisée.....	63
Figure 4. 7. Block Face avant du LabView du système RdP- Modélisation sans faute.....	64
Figure 4. 8. Block diagramme du LabView - une partie du système RdP	65
Figure 4. 9. Block diagramme du LabView - exemple d'une commande	65
Figure 4. 10. Block Face avant du LabView du système RdP- Modélisation avec faute	66
Figure 4. 11. AdF de la cellule robotisée, correspondant à F	70
Figure 4. 12. RdP avant le franchissement.....	71
Figure 4. 13. RdP λ avant le franchissement.....	72
Figure 4. 14. RdP λ après le franchissement.....	72
Figure 4. 15. Transformation de la porte logique « ET » de l'AdF en RdP λ	74
Figure 4. 16. Transformation de la porte logique « OU » de l'AdF en RdP λ	74
Figure 4. 17. Block Face avant du LabView du système RdP λ - Modélisation sans faute.....	75
Figure 4. 18. Block diagramme du LabView du système RdP λ	75
Figure 4. 19. Étapes du système de diagnostic proposé	76
Figure 4. 20. Block Face avant sous LabView du système RdP λ - Modélisation avec faute	78

Liste des Tableaux

Tableau 1. 1. Avantages et inconvénients des robots sériels et parallèles	7
Tableau 1. 2. Comparaison entre l'architecture sérielle et parallèle	8
Tableau 4. 1. Signification des places	65
Tableau 4. 2. Paramètres d'usinage adoptés dans les travaux	68
Tableau 4. 3. Signification des places	73
Tableau 4. 4. Taux de défaillance du niveau 2 – AdF – RdP λ	76
Tableau 4. 5. Taux de défaillance du niveau 1 et de l'événement redouté – AdF – RdP λ	76

Abréviations

<i>SdF</i>	Sûreté de Fonctionnement
<i>GMAO</i>	Gestion de la maintenance assistée par ordinateur
<i>ERP</i>	Enterprise Resource Planning
<i>SCADA</i>	Supervisory Control And Data Acquisition
<i>MP</i>	Maintenance préventive
<i>OSA-CBM</i>	Open Systems Architecture for Condition-Based Maintenance
<i>PdM</i>	Maintenance prédictive
<i>PRM</i>	Maintenance proactive
<i>MTBF</i>	Mean Time Between Failures
<i>MC</i>	Maintenance corrective
<i>AMDEC</i>	Analyse des Modes de Défaillance, de leurs Effets et de leurs Criticité
<i>AdD</i>	Arbres de Défaillances
<i>AdF</i>	Arbre des Fautes
<i>APR</i>	Analyse Préliminaire des Risques
<i>DF</i>	Diagramme de Fiabilité
<i>MEE</i>	Méthode de l'Espace des États
<i>RdP</i>	Réseaux de Petri
<i>HAZOP</i>	HAZard and Operability study
<i>AE</i>	Méthodes de l'arbre d'évènements
<i>DCC</i>	Méthode du diagramme causes-conséquences
<i>TV</i>	Méthodes de la table de vérité
<i>TD</i>	Table de décision
<i>ER</i>	Événement redouté
<i>APD</i>	Analyse Préliminaire des Dangers
<i>AMDE</i>	Analyse des Modes de Défaillance et de leurs Effets
<i>MAC</i>	Méthode de l'Arbre des Causes
<i>MACQ</i>	Méthode de l'Arbre des conséquences ou des arbres d'Événements
<i>AC</i>	Arbre des conséquences
<i>IA</i>	Surveillance par Intelligence Artificielle
<i>AEF</i>	Automates d'États Finis
<i>MLP</i>	Réseaux neuronaux perceptron multicouches
<i>CNN</i>	Réseaux neuronaux convolutifs
<i>RNN</i>	Réseaux neuronaux récurrents
<i>GAN</i>	Réseaux neuronaux génératifs adversatifs
<i>SNN</i>	Réseaux de neurones à impulsions
<i>DNNs</i>	Réseaux de neurones profonds
<i>K</i>	Conflit
<i>RdPS</i>	RdP Synchronisé
<i>RdPT</i>	Réseaux de Petri Temporisés
<i>RdPTt</i>	Réseaux de Petri T-temporisés
<i>RdPPt</i>	Réseaux de Petri P-temporisés
<i>SPN</i>	Réseaux de Petri Stochastiques
<i>GSPN</i>	Réseaux de Petri stochastiques généralisés
<i>DSPN</i>	Réseaux de Petri stochastiques et déterministes
<i>ESPN</i>	Réseaux de Petri stochastiques étendus
<i>RCP</i>	Réseaux de Petri colorés

Abréviations

<i>RdPI</i>	RdP Interprétés
<i>APN</i>	Réseaux de Petri algébriques
<i>RdPF</i>	Réseaux de Petri flous
<i>HPN</i>	Réseaux de Petri Hiérarchiques
<i>RdPC</i>	Réseaux de Petri continus
<i>RdPH</i>	Réseau de Petri hybride
<i>MIT</i>	Massachussetts Institute of Technology
<i>LabView</i>	Laboratory Virtual Instrument Engineering Workbench
<i>PN</i>	Petri Net
λPN	Lambda Petri Net
<i>FT</i>	Fault Tree

Nomenclature

M	Marquage
P	Place
T	Transition
F	Ensemble des arcs orientés
M_0	Marquage initial
\mathbb{N}	Ensemble des entiers naturels
W	Matrice d'incidence
W^+	Matrice d'incidence avant
W^-	Matrice d'incidence arrière
\underline{S}	Vecteur caractéristique
M'	Marquage accessible
D	Fonction
dt_i	Unités de temps pour une transition
dp_i	Unités de temps pour une place
$d_{imin}(t_i)$	Unités de temps
$d_{imin}(p_i)$	Unités de temps
$d_{imax}(t_i)$	Unités de temps
$d_{imax}(p_i)$	Unités de temps
$d_{imin}(p_i, t_i)$	Unités de temps
$d_{imax}(p_i, t_i)$	Unités de temps
λ	Lambda

Introduction Générale

Introduction Générale

Dans l'industrie 4.0, la robotisation est souvent privilégiée lorsqu'il s'agit de substituer l'opérateur affecté à des opérations manuelles à forte pénibilité, pouvant engendrer des troubles musculo-squelettiques. Les performances actuelles des robots en termes de précision de suivi de trajectoires permettent d'envisager la robotisation en usinage et plus précisément sur des opérations de parachèvement des pièces, telles que le surfacage, le perçage, le détournage, le sciage, l'ébarbage, le meulage et le polissage.

Comme tout système technologique, une maintenance est obligatoire et nécessaire afin de garantir les performances d'une telle cellule d'usinage robotisée. Dans le cas de défaillances, la minimisation du temps de remise en route est primordiale afin de ne pas pénaliser la productivité de la cellule concernée. C'est dans ce contexte que nous proposons une stratégie de surveillance basée sur l'utilisation d'un réseau de Petri implémenté à partir de l'Arbre des Fautes de la cellule robotisée d'usinage avancée du Laboratoire Génie de Production (LGP) de l'École Nationale d'Ingénieurs de Tarbes-France (ENIT), en collaboration avec le Laboratoire Génie de Production et Maintenance Industrielle (LGPMI) de l'Institut de Maintenance et de Sécurité Industrielle - Oran.

Les multiples reconfigurations et la complexité des cellules robotisées d'usinage exigent de concevoir des systèmes d'aide à la surveillance de plus en plus performants. Dans le cadre de la gestion de la cellule robotisée d'usinage en parachèvement, implantée au LGP, nous nous intéressons à la modélisation des changements dans la dynamique du système lorsqu'une ou plusieurs défaillances surviennent.

Cette thèse est structurée en quatre chapitres :

- Dans le premier chapitre, nous mettons en lumière l'importance des robots dans le secteur industriel. Nous présentons tout d'abord, un historique des robots industriels, ainsi que l'usinage robotisé. Vient ensuite la présentation des différentes architectures des robots d'usinage (sérielle, parallèle, hybride) illustrées par des exemples explicatifs, accompagnées de leurs avantages et inconvénients. Et puis nous détaillons les différents types d'opérations de finition de parachèvement.
- Dans le second chapitre, nous présentons tout d'abord la maintenance industrielle, puis un état de l'art et une classification des méthodes et modèles de surveillance industrielle. Ce chapitre met en lumière l'importance de la surveillance industrielle dans le contexte complexe actuel de l'industrie et offre une méthodologie de choix des techniques et modèles disponibles.
- Dans le chapitre 3, nous mettons en lumière l'importance des RdP dans le contexte complexe actuel de l'industrie. Nous présentons tout d'abord un historique et les notions de base des réseaux de Petri illustrées par des exemples explicatifs. Ensuite, nous détaillons les propriétés mathématiques et génériques des RdP : leurs analyses par graphe des marquages, par arbre de couverture, par graphe de couverture, par l'algèbre linéaire et par réduction. Aussi leurs structures particulières ont été présentées, accompagnées de leurs avantages et inconvénients, suivies des différents types de RdP de haut niveau.

Introduction Générale

- Dans le chapitre 4, deux études de cas ont été déployées, en collaboration entre le LGPMI/IMSI/Univ-Oran 2 et le LGP/ENIT/UTTOP, sur la cellule robotisée d'usinage avancée. La première étude ne prend en compte que l'aspect qualitatif de l'analyse effectuée sur la cellule de parachèvement de pièces parallélépipédiques, tandis que la seconde étude prend en compte les aspects qualitatifs et quantitatifs de l'analyse effectuée sur la cellule de parachèvement de pièces cylindriques. Dans les deux cas d'études, nous avons effectué des simulations de diagnostic de fautes après implémentation sous environnement LabView.

Chapitre 1 : Usinage et Parachèvement robotisés

1.1	Introduction.....	4
1.2	Robots industriels	4
1.3	Usinage robotisé	5
1.4	Architectures des robots d'usinage.....	5
1.4.1	Architecture sérielle	5
1.4.2	Architecture parallèle	6
1.4.3	Architecture hybride.....	6
1.4.4	Avantages et Inconvénients des robots sériels et parallèles.....	7
1.4.5	Avantages et Inconvénients du robot hybride.....	8
1.5	Critères de performance des robots industriels.....	9
1.6	Opérations de parachèvement.....	9
1.6.1	Découpage.....	10
1.6.2	Sciage.....	10
1.6.3	Perçage.....	10
1.6.4	Détourage.....	11
1.6.5	Surfaçage.....	11
1.6.6	Meulage.....	12
1.6.7	Ponçage.....	12
1.6.8	Ébarbage.....	12
1.6.9	Polissage.....	13
1.7	Conclusion	13

Chapitre 1 : Usinage et parachèvement robotisés

1.1 Introduction

L'usinage robotisé et le parachèvement sont deux étapes clés dans le processus de fabrication moderne et leur relation est devenue de plus en plus étroite avec l'avènement des technologies de fabrication avancées. L'usinage robotisé utilise des robots pour effectuer des tâches d'usinage telles que la découpe, le fraisage, le perçage et le tournage avec une précision et une répétabilité élevée. Ces étapes de fabrication précédant le parachèvement apportent une forte valeur ajoutée à la pièce. Au contraire, les opérations de parachèvement sont en majorité effectuées manuellement et peuvent engendrer des taux de rebuts importants et, par conséquent, des pertes non négligeables pour l'entreprise. Dans ce contexte, la robotisation des opérations de parachèvement peut entraîner une augmentation de la compétitivité des entreprises.

Ce premier chapitre met en lumière l'importance des robots dans le secteur industriel. Nous présentons tout d'abord un historique des robots industriels, ainsi que l'usinage robotisé. S'ensuit la présentation des différentes architectures des robots d'usinage (sérielle, parallèle, hybride) illustrées par des exemples explicatifs, accompagnées de leurs avantages et inconvénients. Ensuite, nous définissons le parachèvement et nous détaillons les différents types d'opérations de de parachèvement.

1.2 Robots industriels

Le premier robot industriel, nommé Unimate, a été développé en 1959 par la société américaine Unimation, créé par George Devol et Joseph Engelberger (Devol, 1961). Ce manipulateur sériel à six axes, actionné par des vérins hydrauliques a été utilisé pour la première fois en 1961 dans l'industrie chez General Motors, pour la manipulation des pièces métalliques à haute température. Depuis, l'utilisation des robots en industrie s'est étendue à une grande variété d'applications telles que l'assemblage, le soudage et la peinture. Les robots permettent d'effectuer des tâches répétitives ou dangereuses à des vitesses élevées et protègent les opérateurs humains des travaux pénibles et à risque. Une étude menée par Graetz et Michaels (Graetz & Michaels, 2015) entre les années 1993 et 2007 sur 17 pays a montré que l'utilisation des robots en industrie a contribué à augmenter le taux annuel de productivité du travail et le produit intérieur brut, (Mejri, 2016).

Il existe de nombreuses définitions des robots industriels, mais la plupart font référence aux notions d'automatisme et de multi-usages. Nous avons repris la norme (ISO 8373, 2021) qui définit un robot industriel comme un manipulateur multi-application, reprogrammable, et commandé automatiquement, programmable sur trois axes ou plus. Ce robot peut être fixe ou mobile et est destiné à des applications d'automatisation industrielle, (Mejri, 2016), (Dumas, 2011) :

- Multi-application : Le robot peut être adapté à différentes applications avec des modifications physiques.
- Reprogrammable : Les mouvements programmés ou les fonctions auxiliaires peuvent être modifiés sans altération physique.
- Axe : Une direction utilisée pour spécifier le mode de mouvement du robot en rotation ou translation (trois axes minimum).

Le robot industriel se distingue généralement par sa structure articulée, composée d'un ou plusieurs bras mobiles terminés par un poignet. Son unité de commande utilise notamment un dispositif de mémoire pour stocker des programmes et des données et, potentiellement, des capteurs pour la perception de l'environnement. Ces machines polyvalentes sont généralement conçues pour accomplir une tâche spécifique de manière cyclique, mais leur flexibilité permet

Chapitre 1 : Usinage et parachèvement robotisés

de les reprogrammer pour réaliser d'autres fonctions sans modification permanente de leur structure matérielle, (Dumas, 2011), (ISO 8373, 2021).

1.3 Usinage robotisé

Les premières études sur les robots d'usinage remontent aux années 90 et, depuis, la recherche s'est concentrée sur le potentiel d'applications des robots dans le domaine de l'usinage. Pour les industries manufacturières, les robots offrent un avantage en termes de gain de temps en travaillant de manière efficace avec un cycle de travail constant. L'usinage robotisé permet des économies en utilisant le même robot pour différentes productions et applications. Les acteurs industriels ont identifié un besoin croissant de robots d'usinage sur le marché en pleine expansion de la robotique industrielle, (Moussavi, 2016), (Moussavi et al., 2014).

1.4 Architectures des robots d'usinage

Des robots industriels dans une multitude de domaines d'application ont stimulé l'innovation au niveau de leur architecture, visant à optimiser leurs performances et leur efficacité. On peut distinguer trois classes principales d'architectures de robots d'usinage industriels : l'architecture sérielle, l'architecture parallèle et l'architecture hybride, voir (Mejri, 2016), (Moussavi, 2016), (Dumas, 2011), (Bouzgarrou, 2001), (Khalil & Dombre, 1999).

1.4.1 Architecture sérielle

L'organe terminal du robot, communément appelé effecteur, est relié à la base par une chaîne cinématique ouverte unique, (Pierre, 1997). Parmi les architectures sérielles les plus répandues, on trouve la structure anthropomorphe, imitant la morphologie humaine, avec une épaule, un bras et un poignet. L'architecture d'un robot sériel est constituée d'un ensemble de corps rigides articulés successivement entre eux par différentes liaisons, généralement des liaisons prismatiques (de type glissière) et/ou rotoïdes (de type pivot). Ces liaisons sont souvent actionnées par des moteurs, permettant de contrôler chaque corps de manière indépendante, ce qui offre au robot une multitude de configurations possibles. Cette chaîne ouverte simple présente l'avantage de faciliter la commande du manipulateur. Les robots industriels dotés de cette architecture sont les plus utilisés dans le secteur industriel, notamment les structures anthropomorphes (figure 1.1) et cartésiennes (figure 1.2), voir (Mejri, 2016), (Moussavi, 2016), (Dumas, 2011).



Figure 1. 1. Robot sériel anthropomorphe, Robot KUKA KR 120, (LGP, 2023)



Figure 1. 2. Robot sériel cartésien, (Barka & Benrar, 2018)

1.4.2 Architecture parallèle

Les robots à structure parallèle se caractérisent par leur chaîne cinématique fermée, qui relie la base du robot à l'organe terminal par un ensemble de chaînes sérielles élémentaires, communément appelées jambes. En général, le nombre des jambes correspond au nombre des degrés de liberté du robot et chaque jambe est contrôlée par un actionneur individuel, (Merlet, 1996). Cette architecture parallèle (figure 1.3) est particulièrement adaptée aux simulateurs de vol, (Stewart, 1965), aux tâches de déplacement, (Pick and Place), (Clavel, 1991), et aux cellules d'usinage, (Mejri, 2016), (Moussavi, 2016), (Dumas, 2011), (Bonnemains, 2009).



Figure 1. 3. Robot parallèle, Robot Delta, (Paccot, 2009)

1.4.3 Architecture hybride

Les limitations inhérentes aux architectures sérielles et parallèles ont conduit le développement d'architectures hybrides, qui combinent les avantages de ces deux approches. Ces architectures hybrides se déclinent en deux formes principales : un robot sériel intégrant un mécanisme parallèle (figure 1.4), ou un robot parallèle intégrant un mécanisme sériel (figure 1.5). La structure des robots hybrides résulte de la combinaison d'une structure sérielle et d'une structure parallèle. L'organe terminal est relié à la base par une chaîne cinématique complexe, communément appelée jambes complexes, (Mejri, 2016), (Moussavi, 2016), (Gogu, 2008).



Figure 1. 4. Robot sériel intégrant un mécanisme parallèle, Robot ABB IRB 6660, (Dumas, 2011)



Figure 1. 5. Robot parallèle intégrant un mécanisme sériel, Robot Tricept T606, (Moussavi, 2016)

1.4.4 Avantages et Inconvénients des robots sériels et parallèles

Le tableau 1.1 présente les avantages et les inconvénients des architectures sérielle et parallèle.

Tableau 1. 1. Avantages et inconvénients des robots sériels et parallèles

Robot	Avantages	Inconvénients
Sériel	<ul style="list-style-type: none"> ➤ Espace de travail : Grâce à leur architecture en chaîne ouverte, les robots sériels offrent un espace de travail vaste pour un encombrement au sol minimal. ➤ Flexibilité : Leur structure leur confère une grande flexibilité, permettant une grande facilité et simplicité à positionner et orienter l'effecteur dans l'espace de travail. 	<ul style="list-style-type: none"> ➤ Rapport charge utile/poids du robot défavorable: La charge maximale admissible est limitée par le poids du robot lui-même, ce qui se traduit par un faible ratio charge utile/poids du robot. ➤ Précision restreinte : Les erreurs d'articulation s'accroissent le long de la chaîne cinématique, réduisant la précision globale du robot. Les

Chapitre 1 : Usinage et parachèvement robotisés

	<ul style="list-style-type: none"> ➤ Modélisation simplifiée : Leur architecture simple, inspirée d'un bras humain, facilite la modélisation et, par conséquent, nécessite une commande moins complexe que celle des robots parallèles. Cette simplicité de conception a contribué à leur développement. 	<p>articulations en porte-à-faux doivent supporter des efforts importants générés par le poids de la structure, ce qui amplifie les erreurs d'articulation et diminue la précision de l'effecteur.</p> <ul style="list-style-type: none"> ➤ Inertie élevée : La masse importante des différents éléments du robot, y compris les corps et les actionneurs, limite ses capacités dynamiques, notamment sa vitesse et son accélération.
	En raison de ces limitations, les robots à structure sérielle sont mieux adaptés aux applications nécessitant un grand espace de travail et une précision modérée.	
Parallèle	<ul style="list-style-type: none"> ➤ Rapport charge utile/masse du robot élevé : La conception parallèle permet une charge utile importante par rapport à la masse du robot lui-même, résultant en un ratio charge utile/masse du robot favorable. ➤ Raideur et précision accrues : La fermeture des chaînes cinématiques dans les robots parallèles confère une raideur et une précision supérieures par rapport aux robots sériels, réduisant les déflexions et améliorant la précision de positionnement. ➤ Caractéristiques dynamiques: La faible masse des éléments en mouvement dans les robots parallèles leur confère des caractéristiques dynamiques remarquables, notamment une accélération et une vitesse élevées. 	<ul style="list-style-type: none"> ➤ Modélisation et commande complexes : La modélisation et la commande des robots parallèles sont plus complexes que celles des robots sériels en raison de la structure cinématique non linéaire. ➤ Volume de travail restreint : Le volume de travail accessible aux robots parallèles est généralement limité par rapport à leur encombrement global, ce qui restreint leur champ d'application. ➤ Flexibilité limitée : La flexibilité des robots parallèles est limitée par leur structure cinématique spécifique, ce qui restreint le nombre d'opérations pouvant être effectuées sur ces machines.
	En raison de leurs caractéristiques, les robots parallèles sont mieux adaptés aux tâches nécessitant une dynamique élevée, une raideur importante et une précision accrue sur un volume de travail relativement restreint (généralement inférieur à 1 m ³).	

Le tableau 1.2 présente une comparaison comparative des avantages et des inconvénients des architectures sérielle et parallèle.

Tableau 1. 2. Comparaison entre l'architecture sérielle et parallèle

Architecture	Sérielle	Parallèle
Espace de travail	+	-
Rigidité	-	+
Inertie	+	-
Rapport charge utile / poids	-	+
Rapport espace de travail / dimensions	+	-
Précision	-	+

1.4.5 Avantages et Inconvénients du robot hybride

L'architecture hybride offre un compromis entre les avantages et les inconvénients des architectures sérielle et parallèle, la rendant particulièrement adaptée aux applications d'usinage. Elle combine la rigidité améliorée de la sous-structure parallèle avec le volume de travail supérieur du mécanisme à poignet sériel, ce qui permet d'accéder à des formes complexes et d'offrir une plus grande flexibilité dans la configuration et l'orientation de l'outil. Cette capacité spatiale étendue permet d'accéder à des formes complexes, d'offrir une plus grande flexibilité dans la configuration et l'orientation de l'outil et de travailler sur plusieurs postes.

Chapitre 1 : Usinage et parachèvement robotisés

1.5 Critères de performance des robots industriels

Les robots industriels sont évalués en fonction des critères de performances définis par les normes (ISO 9283, 1998) et (ANSI/RIA R15, 1990). Ces performances permettent de déterminer les capacités des robots industriels et leur adaptation à des applications spécifiques. Elles sont dérivées des données techniques fournies par les constructeurs, (Mejri, 2016), (Dumas, 2011) :

- Exactitude de positionnement : Capacité du robot à atteindre une position donnée avec précision.
- Répétabilité de positionnement : Capacité du robot à reproduire une position donnée de manière cohérente.
- Espace de travail : Volume dans lequel le robot peut déplacer son effecteur.
- Nombre d'axes avec la plage angulaire de chaque axe : Nombre d'articulations du robot et leur amplitude de mouvement.
- Vitesses et accélérations articulaires maximales : Vitesses et accélérations les plus élevées que chaque articulation peut atteindre.
- Vitesse et accélération maximales de l'organe terminal : Vitesse et accélération les plus élevées que l'extrémité du bras du robot peut atteindre.
- Résolution de mouvement des articulations : Précision minimale avec laquelle les articulations du robot peuvent se déplacer.
- Charge utile : Poids maximal que le robot peut manipuler.
- Conditions générales de travail : Conditions environnementales dans lesquelles le robot peut fonctionner (positionnement, température, environnement, ...).

En usinage robotisé, la précision est d'une importance capitale pour respecter les tolérances dimensionnelles des pièces à usiner. Elle dépend de trois paramètres que sont l'exactitude de positionnement, la répétabilité et la résolution. Selon les études de (Belchior et al., 2013), les deux premiers paramètres peuvent être influencés par la position et la configuration du robot dans son espace de travail.

1.6 Opérations de parachèvement

Dans l'usinage des pièces mécaniques, une ou des opérations de finition sont généralement nécessaires. Ces opérations de finition sont généralement désignées comme opérations de parachèvement. Le parachèvement regroupe une multitude de techniques de finition selon l'objectif recherché (Dumas, 2011), (Davim & Jain, 2008) :

- Mécaniques : Augmentation de la dureté superficielle par écrouissage (l'écrouissage induit une déformation plastique du matériau, ce qui renforce la surface et améliore sa résistance à l'usure et à la fatigue) ; Élimination des effets de bord (les effets de bord, tels que les bavures et les déformations localisées, peuvent fragiliser la pièce et nuire à son bon fonctionnement).
- Géométriques : Mise à la longueur et à l'épaisseur précises (les opérations de finition permettent de respecter les tolérances dimensionnelles requises pour un assemblage correct et un fonctionnement optimal des pièces) ; Obtention d'un état de surface particulier (l'état de surface, caractérisé par la rugosité et la texture, est crucial pour des propriétés telles que la friction, la lubrification et la résistance à la corrosion) ; Création de trous de fixations (les trous de fixations sont nécessaires à l'assemblage des différentes pièces d'un mécanisme, les opérations de finition assurent la précision et la qualité de ces trous).
- Esthétiques : Obtention d'un aspect visuel particulier selon des exigences esthétiques (surface lisse, brillante, etc..).

Chapitre 1 : Usinage et parachèvement robotisés

Nous présentons dans ce qui suit les principales opérations de parachèvement.

1.6.1 Découpage

Le découpage permet de découper une pièce selon une forme spécifique. Les techniques de découpage les plus courantes sont (Charles, 2015), (Dumas, 2011), (Caillaud et al., 2009) :

- Découpage par fraisage : Cette méthode utilise une fraise rotative pour découper la pièce, permet une découpe d'une grande variété de matériaux jusqu'à une épaisseur de 25 mm. L'effort exercé sur la pièce est minimal et la zone affectée thermiquement est réduite (environ 0.5 mm sur les métaux), ce qui limite les déformations. L'avantage majeur de ce procédé est qu'il permet d'atteindre des vitesses de coupe importantes.
- Découpage par laser : Cette méthode utilise un laser pour découper la pièce.
- Découpage par jet d'eau ou d'abrasif : Cette méthode utilise un jet d'eau à haute pression pour découper la pièce. Cette technique est la plus utilisée après les procédés conventionnels. Cette méthode permet de découper une grande variété de matériaux comme les métaux, pierres, marbres ou verres dans les épaisseurs allant jusqu'à 600 millimètres. Sa caractéristique principale est qu'elle assure une découpe à froid, donc sans endommagement ni déformation thermique du matériau, d'où un avantage important pour la découpe des composites. Industriellement, cette technique est limitée au perçage et à la découpe.

1.6.2 Sciage

Le sciage est un procédé d'usinage par enlèvement de matière qui permet de tronçonner une pièce en plusieurs parties. Il s'agit d'une opération courante dans divers domaines industriels, notamment la construction métallique, la menuiserie et l'usinage mécanique. Les techniques de sciage les plus courantes sont (Des Métaux, 2017) (Abdi, 2017) :

- Sciage par scie à ruban : Cette méthode utilise une lame fine et flexible en forme de ruban continu qui se déplace en boucle fermée autour de deux volants. La lame est guidée par des poulies et entre en contact avec la pièce à usiner, permettant une coupe précise et efficace. Le sciage à ruban est particulièrement adapté aux coupes de grandes dimensions et aux matériaux durs, tels que l'acier et l'aluminium.
- Sciage par scie circulaire : Cette méthode utilise une lame circulaire rotative qui attaque la pièce à usiner par sa périphérie. La lame est montée sur un arbre moteur et sa vitesse de rotation peut être ajustée en fonction du matériau à couper. Le sciage circulaire est idéal pour les coupes droites et précises dans des pièces de dimensions moyennes ou petites.
- Sciage par chaîne : Cette méthode utilise une chaîne dentée qui se déplace à grande vitesse sur un guide. La chaîne est entraînée par un moteur et ses dents arrachent les copeaux de matière pour effectuer la coupe. Le sciage par chaîne est particulièrement utile pour les coupes rapides et brutes dans des matériaux tendres ou fibreux, tels que le bois et les plastiques. Elle est utilisée dans les travaux de construction, de jardinage et de sylviculture. La tronçonneuse est un exemple typique d'outil utilisant cette technique.

1.6.3 Perçage

Le perçage permet de créer des trous précis dans une pièce. Cette technique utilisée dépend du diamètre et de la profondeur du trou ainsi que du matériau de la pièce. Parmi les méthodes les plus courantes, on trouve (Ernotte et al., 2005) :

- Perçage par torsion : Le perçage par torsion est la méthode la plus courante pour percer des trous de petit diamètre. Il utilise un foret rotatif pour enlever le matériau de la pièce. Le foret est généralement en acier trempé et sa pointe est affûtée selon un angle spécifique. La pièce

Chapitre 1 : Usinage et parachèvement robotisés

à percer est fixée dans un étau ou dans un mandrin et le foret est appliqué sur la surface de la pièce. Le foret est ensuite mis en rotation et une pression est exercée pour l'enfoncer dans la pièce. Le perçage par torsion est un processus relativement simple et rapide, mais il peut générer de la chaleur et des vibrations.

- Perçage par percussion : Le perçage par percussion est utilisé pour percer des trous dans des matériaux durs ou cassants, tels que le béton ou la pierre. Il utilise un foret rotatif et un marteau pour percer le trou. Le foret est généralement en carbure de tungstène et sa pointe est affûtée selon un angle spécifique. La pièce à percer est fixée dans un étau ou dans un mandrin et le foret est appliqué sur la surface de la pièce. Le foret est ensuite mis en rotation et le marteau frappe la tête du foret pour le propulser dans la pièce. Le perçage par percussion est un processus plus lent que le perçage par torsion, mais il permet de percer des trous dans des matériaux qui seraient difficiles à percer avec d'autres méthodes.
- Trépanage : Le trépanage est utilisé pour percer des trous de grand diamètre. Il utilise un outil à plusieurs dents appelé trépan pour enlever le matériau de la pièce. Le trépan est généralement en acier trempé et ses dents sont disposées en quinconce. La pièce à percer est fixée dans un étau ou dans un mandrin et le trépan est appliqué sur la surface de la pièce. Le trépan est ensuite mis en rotation et une pression est exercée pour l'enfoncer dans la pièce. Le trépanage est un processus lent et laborieux, mais il permet de percer des trous de grand diamètre avec une grande précision.

1.6.4 Détourage

Le détourage, une opération d'usinage essentielle dans la fabrication des pièces composites, consiste à enlever la matière et est effectué après le démoulage des pièces. Ce processus utilise des outils coupants dont les arêtes périphériques enlèvent la matière suivant une direction radiale par rapport à l'axe de rotation, générant ainsi une surface usinée parallèle à cet axe. Deux principales méthodes sont employées : l'usinage conventionnel (outils coupants, scies diamantées) et l'usinage non conventionnel (jet d'eau abrasif), (Zenia, 2017) (M. Haddad, 2013).

1.6.5 Surfaçage

Le surfaçage est une technique visant à améliorer l'état de surface d'une pièce en éliminant les imperfections et en lui conférant la rugosité souhaitée. Cette technique peut être réalisée par différentes méthodes, telles que (Morandea et al., 2011) :

- Brochage : Le brochage utilise un outil à dents multiples appelé broche pour enlever de la matière sur toute la longueur de la pièce. La broche est généralement constituée d'acier trempé et ses dents sont disposées en quinconce. La pièce à brocher est fixée dans un étau et la broche est déplacée le long de la pièce en exerçant une pression. Le brochage permet d'obtenir une surface plane et précise, mais il peut laisser des stries sur la surface.
- Lapping : Le lapping utilise des abrasifs fins pour polir la surface de la pièce. Les abrasifs sont généralement sous forme de poudre ou de suspension et sont appliqués sur une surface plane appelée plateau de lapping. La pièce à polir est placée sur le plateau de lapping et est déplacée en cercle ou en va-et-vient. Le lapping permet d'obtenir une surface très lisse et brillante, mais il est un processus long et laborieux.
- Rodage : Le rodage utilise une pièce mobile et une pièce fixe entre lesquelles un abrasif est introduit pour obtenir un ajustement précis et une surface lisse. La pièce mobile est généralement appelée arbre et la pièce fixe est appelée alésage. L'abrasif est généralement sous forme de pâte ou de suspension et est appliqué entre les deux pièces.

Chapitre 1 : Usinage et parachèvement robotisés

- Les deux pièces sont mises en rotation et une pression est exercée pour les serrer l'une contre l'autre. Le rodage permet d'obtenir un ajustement très précis et une surface très lisse, mais il est un processus long et délicat.

1.6.6 Meulage

Le meulage est un procédé d'usinage par enlèvement de matière qui utilise une meule ou une pierre abrasive pour façonner et lisser la surface d'une pièce. Il est couramment utilisé pour affiner la finition des métaux, des plastiques et d'autres matériaux, (Gognau, 2003).

1.6.7 Ponçage

Le ponçage est un procédé d'usinage par enlèvement de matière qui permet d'obtenir un état de surface spécifique sur une pièce, généralement en vue d'une finition ultérieure ou pour améliorer ses propriétés fonctionnelles. Il consiste à frotter la surface de la pièce avec un abrasif afin d'éliminer les irrégularités, les aspérités et les imperfections. Les techniques de ponçage les plus courantes sont (Dumas, 2011), (Robin, 2007):

- Ponçage manuel : Utilise des outils simples comme du papier de verre, des brosses abrasives ou des blocs de ponçage. Adapté aux petites pièces et aux surfaces planes.
- Ponçage mécanique avec meule : Utilise une meule rotative recouverte d'abrasif pour poncer la surface de la pièce. Adapté aux surfaces planes et courbes.
- Ponçage par bande abrasive : Utilise une bande abrasive continue qui défile sur deux poulies et frotte la surface de la pièce. Adapté aux surfaces planes et aux pièces de grande taille.
- Sablage : Utilise un jet de sable ou d'autres abrasifs projetés à haute pression pour nettoyer ou décaper la surface de la pièce. Adapté aux surfaces difficiles d'accès et aux matériaux durs.

1.6.8 Ébarbage

L'ébarbage est une opération d'usinage qui consiste à éliminer les bavures spécifiques créées lors des opérations d'usinage précédentes. Il s'agit d'un processus de découpe qui permet d'obtenir des bords nets et précis. Les techniques d'ébarbage les plus courantes sont (Schoose, 2016) (Briard, 2003) :

- Ébarbage manuel : Cette méthode traditionnelle utilise des outils manuels tels que des limes, des brosses métalliques, des grattoirs et des papiers abrasifs pour éliminer les bavures une à une. L'ébarbage manuel est généralement utilisé pour des petites pièces ou pour des zones difficiles d'accès.
- Ébarbage par vibration : Cette méthode utilise des vibrations pour détacher les bavures de la pièce. La pièce est placée dans un bac vibrant contenant un média abrasif, tel que des billes ou des copeaux de bois. Les vibrations du bac provoquent le frottement du média contre la pièce, ce qui déloge les bavures. L'ébarbage par vibration est efficace pour les pièces de petite et moyenne taille, et permet de traiter plusieurs pièces simultanément.
- Ébarbage par jet d'eau : Cette méthode utilise un jet d'eau à haute pression pour éliminer les bavures de la pièce. La pression du jet d'eau permet de couper ou de casser les bavures, même dans les zones difficiles d'accès. L'ébarbage par jet d'eau est particulièrement adapté aux matériaux durs et fragiles, tels que l'acier trempé et les plastiques.

Chapitre 1 : Usinage et parachèvement robotisés

1.6.9 Polissage

Le polissage qui est une dernière étape du processus de finition consistant à affiner la surface d'une pièce, est crucial pour obtenir un résultat esthétique et fonctionnel optimal. Il s'agit d'éliminer les imperfections microscopiques et de créer une surface lisse et réfléchissante (Doche & Hihn, 2021), (Guiot, 2012).

- Polissage mécanique : le polissage mécanique utilise des outils abrasifs (pâte à polir, feutre) sur une surface en mouvement pour enlever progressivement les irrégularités.
- Polissage par flux abrasif : Un mélange d'abrasif et de liquide est projeté à haute pression sur la surface à polir. Les particules abrasives éliminent les imperfections tout en laissant une surface lisse.

1.7 Conclusion

Face aux difficultés rencontrées pour obtenir des pièces finies de qualité optimale, l'usinage et le parachèvement robotisés s'avèrent des leviers clés dans les processus de fabrication des nombreux avantages, notamment une meilleure qualité, une productivité accrue, des coûts réduits et une sécurité améliorée.

Dans la première partie de ce chapitre, nous avons montré les divers modèles de robots existants et présenté certains critères aidant à décider du meilleur choix approprié pour le montage d'une cellule robotisée dédiée. La deuxième partie du chapitre traite des principales opérations de parachèvement et montre leur large diversité ainsi que leur grande importance dans les processus de fabrication.

La suite de notre travail s'inscrit parmi les recherches qui visent à améliorer la fiabilité des procédés de parachèvement et, plus particulièrement, à développer une stratégie de surveillance sur une cellule robotisée de parachèvement des pièces mécaniques.

Chapitre 2 : Surveillance Industrielle

2.1	Introduction	16
2.2	Notions de base et définitions utilisées en surveillance	16
2.2.1	Mode de fonctionnement	16
2.2.2	Dégradation.....	16
2.2.3	Défaillance	16
2.2.4	Panne.....	16
2.2.5	Défaut.....	17
2.2.6	Risque	17
2.2.7	Erreur	17
2.2.8	Faute.....	17
2.2.9	Tolérance aux fautes	17
2.2.10	Reconfiguration.....	17
2.2.11	Sûreté de fonctionnement	17
2.2.11.1	Fiabilité.....	18
2.2.11.2	Maintenabilité.....	18
2.2.11.3	Disponibilité.....	18
2.2.11.4	Sécurité.....	18
2.2.12	Maintenance industrielle	18
2.2.12.1	Maintenance Corrective	19
2.2.12.1.1	Maintenance palliative.....	20
2.2.12.1.2	Maintenance curative.....	20
2.2.12.2	Maintenance Préventive	20
2.2.12.2.1	Maintenance systématique.....	20
2.2.12.2.2	Maintenance conditionnelle.....	21
2.2.12.2.3	Maintenance prédictive.....	21
2.2.12.2.4	Maintenance proactive.....	22
2.2.12.3	Autre classification des activités de maintenance	22
2.3	Surveillance industrielle	22
2.3.1	Détection	23
2.3.2	Diagnostic	23
2.4	Classification des Méthodes et Modèles de Surveillance.....	23
2.4.1	Méthodes de Surveillance avec Modèles	24
2.4.1.1	Méthodes par Modélisations Fonctionnelle et Matérielle.....	24
2.4.1.1.1	Analyse des Modes de Défaillance, de leurs effets et de leur Criticité (AMDEC)..	24

2.4.1.1.2	Arbre de défaillance (AdD) / Arbre des Fautes (AdF)	25
2.4.1.1.3	Analyse Préliminaire des Risques (APR)	26
2.4.1.1.4	Diagramme de Fiabilité (DF)	26
2.4.1.1.5	Méthode de l'Espace des États (MEE)	26
2.4.1.1.6	Réseaux de Petri (RdP)	27
2.4.1.1.7	HAZard and Operability Study (HAZOP)	27
2.4.1.1.8	Arbre d'Évènements (AE)	27
2.4.1.1.9	Méthode du Diagramme Causes-Conséquences (DCC)	28
2.4.1.1.10	Méthodes de la Table de Vérité (TV) et de la Table de Décision (TD)	28
2.4.1.2	Méthodes de Surveillance par Modélisation Physique	28
2.4.1.2.1	Redondances Physiques et Analytiques	29
2.4.1.2.2	Estimation Paramétrique	29
2.4.2	Méthodes de Surveillance sans Modèles	29
2.4.2.1	Surveillance basée sur des Outils Statistiques du Traitement du Signal	29
2.4.2.1.1	Test de Franchissement de Seuil	30
2.4.2.1.2	Test de Moyenne	30
2.4.2.1.3	Test de Variance	30
2.4.2.2	Surveillance par Intelligence Artificielle (IA)	30
2.4.2.2.1	Modèles Comportementaux	30
2.4.2.2.1.1	Automates d'États Finis (AEF)	31
2.4.2.2.1.2	Réseaux de Petri (RdP)	31
2.4.2.2.2	Méthodes de Reconnaissance de Formes	31
2.4.2.2.2.1	Système Expert	31
2.4.2.2.2.2	Outil Statistique	32
2.4.2.2.2.3	Raisonnement à Partir de Cas (RàPC)	32
2.4.2.2.2.4	Réseaux de neuronaux (RNA)	32
2.4.2.2.2.5	Logique Floue	33
2.4.2.2.2.6	Système Neuro-Flou	33
2.4.2.2.3	Méthodes Explicatives	34
2.4.2.2.3.1	Graphes Causaux	34
2.4.2.2.3.2	Graphes Contextuels	34
2.5	Conclusion	35

Chapitre 2 : Surveillance Industrielle

2.1 Introduction

La maintenance est une approche nécessaire dans laquelle les équipements et les machines sont surveillés et analysés. Dans ce chapitre on s'intéresse à la surveillance industrielle, notre travail s'inscrit dans la lignée d'études recensant les travaux théoriques ou empiriques dans le domaine de l'industrie. Dans ce chapitre nous avons réalisé un état de l'art et une classification des méthodes et modèles de surveillance industrielle. La surveillance peut être définie de plusieurs manières, cette surveillance est composée essentiellement de deux phases : la détection de défauts et le diagnostic de pannes. Le diagnostic se décompose en deux fonctions : la localisation et l'identification des causes. La surveillance permet de donner des informations sur l'état du système considéré. Ces informations peuvent être données aux opérateurs qui peuvent alors agir en conséquence, ou être utilisées afin de modifier, ou configurer le pilotage automatiquement.

Nous présentons dans ce chapitre, tout d'abord la Sûreté de Fonctionnement (SdF), et la maintenance, puis un état de l'art et une classification des méthodes et modèles de surveillance industrielle. Ce chapitre met en lumière l'importance de la surveillance industrielle dans le contexte complexe actuel de l'industrie et offre une méthodologie de choix des modèles disponibles.

2.2 Notions de base et définitions utilisées en surveillance

La terminologie et les définitions relatives à la surveillance et à la maintenance varient d'une communauté scientifique, ou d'un secteur industriel à l'autre. Précisons la terminologie adoptée dans la suite de nos travaux : un système technologique peut être sujet à des usures, des déviations, des défauts, des perturbations, des dysfonctionnements, des pannes, des erreurs, des défaillances, voire des fautes. Ces notions sont précisément explicitées, voir (R. Nouredine, 2008), (R. Nouredine et al., 2008), (Heng, 2002), (Monchy & Vernier, 2000), (F. Nouredine, 1996), (Arlat & Laprie, 1995), (Zwingelstein, 1995), (NF EN 13306, 2018), (AFNOR, 1994) (IEC, 1990), (AFNOR NF X60-500, 1988).

2.2.1 Mode de fonctionnement

Le mode de fonctionnement est un état dans lequel un système ou un équipement se trouve lorsqu'il est en train d'accomplir sa fonction. Il peut être normal, dégradé ou anormal.

2.2.2 Dégradation

La dégradation est un processus progressif d'altération de l'état d'une entité ou des performances d'un équipement, pouvant entraîner une défaillance immédiate.

2.2.3 Défaillance

La défaillance est l'impossibilité d'une entité à assurer une fonction requise. La défaillance est un événement soudain qui conduit à une panne. Elle peut être causée par une dégradation, une erreur de conception ou de fabrication, ou un événement extérieur, tel qu'un incendie ou une inondation.

2.2.4 Panne

La panne est l'inaptitude d'une entité (composant ou système) à assurer une fonction requise. Elle peut être temporaire ou permanente.

Chapitre 2 : Surveillance Industrielle

2.2.5 Défaut

Le défaut c'est la non-conformité d'une entité par rapport à des exigences spécifiées. Il peut être matériel, logiciel, ou organisationnel.

2.2.6 Risque

Le risque c'est la combinaison de la probabilité d'un événement et de la gravité des conséquences de cet événement. Il peut être lié à la sécurité, à la sûreté, ou à la rentabilité.

2.2.7 Erreur

Une erreur est une manifestation de la faute activée, un état (ou partie de l'état) du système pouvant mener à une défaillance.

2.2.8 Faute

La faute est un défaut présent dans un composant matériel ou logiciel qui a le potentiel de provoquer une erreur. Elle peut être causée par divers facteurs, tels qu'une erreur de conception, une défaillance matérielle, une erreur de programmation ou une erreur humaine. La faute ne produit pas d'erreur immédiatement ; elle reste dormante jusqu'à ce qu'elle soit activée par un événement spécifique, intentionnel ou accidentel.

2.2.9 Tolérance aux fautes

La tolérance aux fautes est la capacité d'un système à maintenir son fonctionnement correct en présence de fautes, c'est-à-dire de défauts dans ses composants matériels ou logiciels. Elle vise à éviter les défaillances qui pourraient résulter de ces fautes, en mettant en œuvre des mécanismes de détection des erreurs et de rétablissement du système.

2.2.10 Reconfiguration

Selon (Kharouati, 2021) et (Romain, 2009), la reconfiguration est l'action de modifier la structure d'un système défaillant de telle sorte que les composants non défaillants permettent de délivrer un service acceptable, bien que dégradé. Elle peut être effectuée pour améliorer les performances, la sécurité, ou la sûreté de l'entité.

2.2.11 Sûreté de fonctionnement

Selon (Palluat, 2006), la Sûreté de Fonctionnement (SdF), est définie comme l'ensemble des aptitudes d'une entité à satisfaire une ou plusieurs fonctions requises dans des conditions données, pendant une durée prévue, tout en évitant les dommages pour elle-même et son environnement.

Selon (IEC, 1990), la SdF est définie par l'ensemble des propriétés qui décrivent la disponibilité et les facteurs qui la conditionne : fiabilité, maintenabilité, disponibilité, sécurité et logistique de maintenance, la durabilité..., ou des combinaisons de ces aptitudes.

SdF est définie par Villemeur (Villemeur, 1988a) comme la science des défaillances et des pannes. La sûreté de fonctionnement (SdF) est définie comme :

- FMDS (Fiabilité, Maintenabilité, Disponibilité et Sécurité).
- Science des défaillances.
- Analyse de risque.

Chapitre 2 : Surveillance Industrielle

Elle se caractérise par des études structurelles statiques et dynamiques des systèmes, du point de vue prévisionnel mais aussi opérationnel et expérimental. Ces études tiennent compte des aspects probabilistes des défaillances techniques et humaines, ainsi que des conséquences de ces défaillances, (Palluat, 2006).

2.2.11.1 Fiabilité

Selon la norme (AFNOR NF X60-500, 1988), la fiabilité est l'aptitude d'une entité à accomplir une fonction requise, dans des conditions données, pendant un intervalle de temps donné.

Elle est mesurée par la probabilité $R(t)$ que l'entité soit non défaillante sur la durée $[0, t]$ sachant qu'elle n'est pas défaillante à l'instant 0.

2.2.11.2 Maintenabilité

Selon la norme (AFNOR NF X60-500, 1988), la maintenabilité est l'aptitude d'une entité à être maintenue ou rétablie, dans un état dans lequel elle peut accomplir une fonction requise, pendant un intervalle de temps donné, lorsque la maintenance est effectuée dans des conditions données, avec des procédures et des moyens prescrits.

2.2.11.3 Disponibilité

Selon la norme (AFNOR NF X60-500, 1988), la disponibilité est l'aptitude d'une entité à être en état d'accomplir une fonction requise, dans des conditions données, à un instant donné ou pendant un intervalle de temps donné.

Elle se mesure par la probabilité $A(t)$ d'être, à l'instant t , en état d'accomplir une fonction requise.

2.2.11.4 Sécurité

Selon (Villemeur, 1988a), la sécurité d'une entité est son aptitude à éviter de faire apparaître, dans des conditions données, des événements critiques ou catastrophiques.

Cette composante de la SdF est primordiale pour les entités où le risque de dommages corporels à l'utilisateur ou à l'environnement est jugé important.

2.2.12 Maintenance industrielle

Selon la norme (NF EN 13306, 2018), la maintenance est définie comme l'ensemble des activités destinées à maintenir ou à rétablir un bien dans un état ou dans des conditions données spécifiques de sûreté de fonctionnement afin qu'il puisse accomplir une fonction requise. Ces activités sont une combinaison entre des activités techniques, administratives et de management.

Les activités de maintenance impliquent des actions techniques impliquant des ressources matérielles et immatérielles, et des prises de décisions complexes sont nécessaires, (Potes Ruiz, 2014), (Rachidi et al., 2013).

La maintenance industrielle est une fonction cruciale qui soutient les processus fondamentaux des organisations, (Alsyouf, 2009). Actuellement, les entreprises s'investissent de plus en plus dans les activités de gestion des connaissances afin d'améliorer leurs processus de maintenance et plus généralement leurs performances, (Potes Ruiz, 2014).

Chapitre 2 : Surveillance Industrielle

Un système de maintenance est une application de maintenance ou de fiabilité qui couvre des différentes activités de la fonction de maintenance, telles que logistique, planning des interventions, gestion des stocks (géré par la GMAO, ERP), le diagnostic et la réparation (systèmes experts, bases de données), la surveillance d'un équipement (SCADA, commande numérique sur un équipement). L'architecture de ces systèmes peut varier en fonction des objectifs visés, (Rasovska et al., 2007).

Selon les travaux suivants (Baron et al., 2023), (Motrani, 2022), (Meraghni, 2021), (Jemmali, 2021), (Vathoopan et al., 2018), (Lesobre, 2015a), (Potes Ruiz, 2014), (Vrignat, 2010a), (Cocheteux, 2010), (R. Nouredine, 2008), (Racoceanu, 2006), (Zemouri, 2003), la maintenance peut être divisée en deux grands types : la maintenance corrective (réactive) et la maintenance préventive, comme illustré à la figure 2.1.

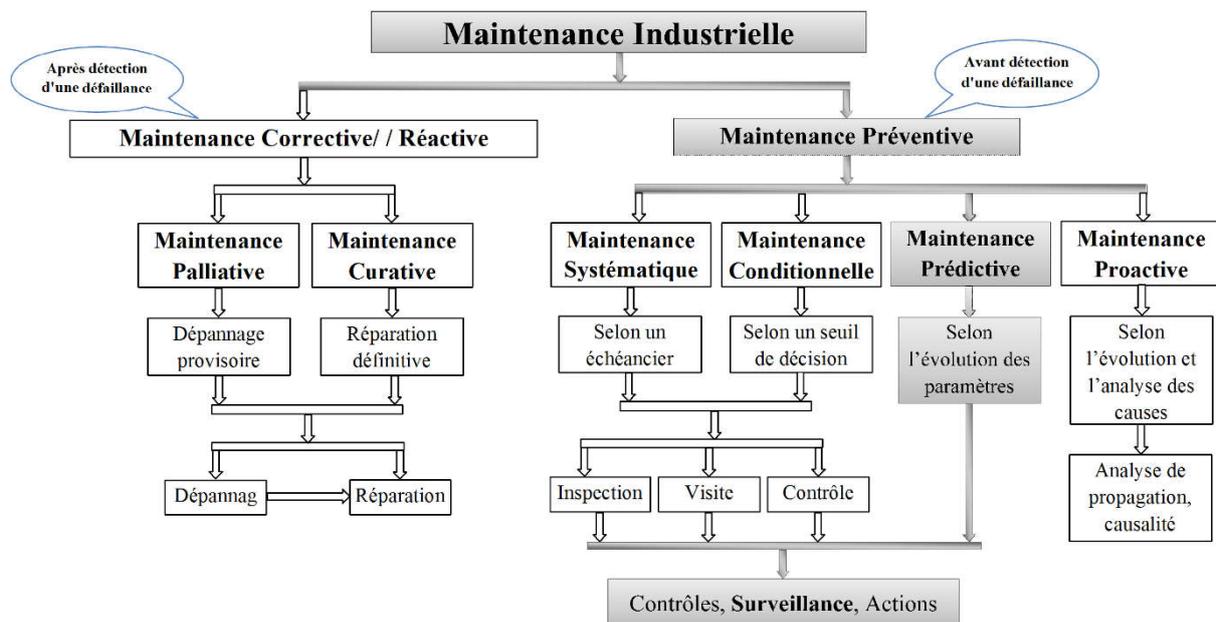


Figure 2. 1. Architecture globale et différentes stratégies de maintenance

2.2.12.1 Maintenance Corrective

Selon la norme (NF EN 13306, 2018), la maintenance corrective, appelée aussi maintenance réactive, consiste à réparer un équipement ou un système après qu'une panne ou une défaillance s'est produite, intervient après la détection et la localisation d'un défaut, a pour but de remettre le matériel en état de fonctionner.

Cette maintenance corrective est alors réalisée complément de la maintenance préventive permet de détecter la défaillance (Cousino et al., 2022). Elle est basée sur la réparation des pannes qui se sont déjà produites. Elle a lieu lorsque le processus ou la machine est à l'arrêt, ce qui peut entraîner des temps d'arrêt imprévus et des coûts de réparation plus élevés. La maintenance corrective est utilisée dans les processus où les défaillances ne sont pas critiques pour la production, (Meraghni, 2021), (Vathoopan et al., 2018).

La politique de la maintenance corrective garantit la durée de vie maximale possible du composant, mais peut entraîner des problèmes de temps d'arrêt ou de sécurité en cas de panne, (Vadeboncoeur, 2022).

La maintenance corrective peut être classée en deux catégories : palliative, curative.

Chapitre 2 : Surveillance Industrielle

2.2.12.1.1 Maintenance palliative

La maintenance palliative qui remet le système en état de fonctionnement provisoire permettant à une entité d'accomplir tout ou une partie de la fonction requise, (Lesobre, 2015a), (Vrignat, 2010a).

2.2.12.1.2 Maintenance curative

La maintenance curative qualifiée des réparations permettant à l'entité de retrouver son état initial, se concentre sur une réparation complète et durable, (Lesobre, 2015a), (Vrignat, 2010a).

2.2.12.2 Maintenance Préventive

Dans une stratégie de maintenance globale, il est important de diminuer les dépannages et les réparations au profit d'activités d'amélioration mais surtout d'entretien préventif structuré, (Vrignat, 2010b).

Selon la norme (NF EN 13306, 2018), la maintenance préventive (MP) est la 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 a pour objectif de détecter et résoudre les défaillances avant qu'ils ne surviennent, de ralentir le vieillissement du matériel et ainsi de retarder l'occurrence de prochaines défaillances. La fiabilité d'un système dépend des effets conjoints du vieillissement et de l'efficacité des maintenances (Cousino et al., 2022), (Jemmali, 2021).

La maintenance préventive peut être classée en quatre grandes catégories : systématique, conditionnelle, prédictive et proactive. Ces catégories sont définies selon la norme (NF EN 13306, 2018).

2.2.12.2.1 Maintenance systématique

Selon la norme (NF EN 13306, 2018), la maintenance systématique est une maintenance préventive qui est exécutée à des intervalles de temps préétablis ou selon un nombre d'unités d'usage prédéfini, sans contrôle préalable de l'état du bien.

La maintenance systématique exige une bonne connaissance des durées de vie et des caractéristiques de fiabilité des différents composants et sous-ensembles, car les pièces sont remplacées systématiquement. Ce type de maintenance est souvent appliqué aux pièces critiques d'un équipement dont le coût d'indisponibilité est élevé ou dont les éléments à changer ne sont pas chers ou bien difficilement accessibles, (Vrignat, 2010b).

La maintenance systématique peut contrôler certains coûts de maintenance, parce que le calendrier et les coûts des interventions sont connus a priori. Cependant, elle peut entraîner une perte de cycle de vie des pièces de rechange, augmenter la maintenance et causer des temps d'arrêt prévisibles. De plus, la maintenance systématique ne permet pas de détecter la dégradation imprévue entre les entretiens, (Vadeboncoeur, 2022). Les industriels sont basés sur l'historique des pannes, pour déterminer les fréquences des interventions sur le système. Le but

Chapitre 2 : Surveillance Industrielle

de cette maintenance est d'intervenir sur le système uniquement à des moments définis par un planning préétabli, (Latrous, 2021).

2.2.12.2.2 Maintenance conditionnelle

Selon la norme (NF EN 13306, 2018), la maintenance conditionnelle est une maintenance préventive qui est basée sur la surveillance du fonctionnement du bien et/ou des paramètres significatifs du fonctionnement de l'équipement intégrant les actions qui en découlent, tels que la température, les vibrations ou la consommation d'énergie.

La maintenance conditionnelle est généralement liée à un type d'évènement prédéterminé (diagnostic, capteur, signal ou mesure), qui est effectuée en fonction de l'état de dégradation, et le diagnostic en temps réel, (Vrignat, 2010b).

2.2.12.2.3 Maintenance prédictive

Selon la norme (NF EN 13306, 2018), la maintenance prédictive, appelée aussi maintenance prévisionnelle, est une maintenance conditionnelle exécutée en suivant les prévisions extrapolées de l'analyse et de l'évaluation de paramètres significatifs de la dégradation du bien.

La maintenance prédictive (PdM) est basée sur l'analyse de paramètres significatifs de la dégradation de l'équipement, mesurés par des capteurs. Le principe de cette maintenance est d'assurer un intervalle maximum entre les réparations et de minimiser le coût et le nombre d'opérations de maintenance programmées, (Guillaume et al., 2023), (Meraghni, 2021), (R. Nouredine, 2008). La maintenance prédictive utilise des techniques d'analyse des données, telles que l'apprentissage automatique, pour identifier les signes avant-coureurs de défaillance, (Ahn et al., 2023), (Lung, 2019), (Paolanti et al., 2018).

La PdM intègre les conditions d'utilisations et l'environnement du système. Contrairement à la maintenance conditionnelle, elle associe une prévision des indicateurs de santé pour la prise de décision de maintenance. Pour mettre en place de cette maintenance il est nécessaire de maîtriser en détail le comportement de l'entité concernée. Cette connaissance permet d'anticiper au mieux le moment où l'intervention doit être exécutée, (Lesobre, 2015b).

PdM est aujourd'hui très présente dans tous les secteurs industriels. Elle est au cœur de la maintenance du futur (Industrie 4.0), grâce notamment à l'avènement de la transformation numérique, basée sur les techniques d'intelligence artificielle. Elle s'articule autour de 7 niveaux fonctionnels, basés sur une architecture OSA-CBM (Open Systems Architecture for Condition-Based Maintenance), (Colin, 2023), et on peut la synthétiser en quatre phases, (Arena et al., 2021), (Jimenez et al., 2020), (Xu et al., 2019) :

- Collecte de données : Les données sont collectées à partir des capteurs installés sur les machines. Ces données peuvent être de nature variée, telles que les vibrations, la température, la pression, etc.
- Prétraitement des données : Les données collectées sont prétraitées pour les rendre exploitables. Ce prétraitement peut inclure des étapes telles que le filtrage, la normalisation, la segmentation, etc.
- Diagnostic et pronostic des défauts : Les données prétraitées sont ensuite analysées pour identifier les signes avant-coureurs de défaillance. Cette analyse peut être réalisée à l'aide de techniques d'apprentissage automatique, de l'intelligence artificielle, etc.
- Prise de décision : Les résultats de l'analyse des données sont utilisés pour prendre une décision sur la stratégie de maintenance à adopter. Cette décision peut concerner, par exemple, le remplacement d'une pièce, la réparation d'un composant, etc.

Chapitre 2 : Surveillance Industrielle

2.2.12.2.4 Maintenance proactive

La maintenance proactive (PRM) est une approche de maintenance qui vise à prévenir les pannes en traitant les causes primaires de défaillance. Elle implique la surveillance et la vérification continue des équipements et des installations, afin d'identifier les signes avant-coureurs de défaillance. C'est une approche de maintenance plus coûteuse à court terme que la maintenance corrective, mais elle peut être plus rentable à long terme, (Baron et al., 2023), (Cocheteux, 2010), (Racoceanu, 2006).

La maintenance proactive est un terme qui combine les principes de la maintenance préventive et prédictive. Elle utilise la maintenance prédictive pour la réparation des causes des problèmes survenant sur les machines ou les processus. Cette modalité de maintenance permet de réduire les temps d'arrêt et d'améliorer la disponibilité des équipements. Ce type de maintenance permet aux équipements de fonctionner presque sans interruption, avec un temps moyen entre les défaillances (MTBF) considérablement allongé. L'objectif principal de la maintenance proactive est l'analyse de l'évolution surveillée des paramètres significatifs de la dégradation du bien, permettant de retarder et de planifier les interventions, (Vrignat, 2010b).

2.2.12.3 Autre classification des activités de maintenance

Une deuxième classification des activités de maintenance est basée sur le degré auquel le système est remis en état de fonctionnement. Cette classification définit quatre types de maintenance, comme indiqué par (Kaddachi, 2021), (Lesobre, 2015a), (H. Wang, 2002), (Pham & Wang, 1996) :

- 1) Réparation parfaite : le système est remis à l'état neuf, avec une durée de vie et une intensité de défaillance identiques à celles d'un système neuf.
- 2) Réparation minimale : le système est remis dans l'état dans lequel il se trouvait avant la défaillance, avec la même intensité de défaillance.
- 3) Réparation imparfaite : le système est remis dans un état meilleur que celui dans lequel il se trouvait avant la défaillance, mais pas à l'état neuf.
- 4) Pire réparation : le système est remis dans un état pire que celui dans lequel il se trouvait avant la défaillance, mais il reste opérationnel.

2.3 Surveillance industrielle

Selon les travaux (Mahdaoui, 2013), (R. Nouredine et al., 2008), (Guillot & Meyer, 2008), (Cassandras & Lafortune, 2008), (Racoceanu, 2006), (Palluat, 2006) et (Palluat et al., 2005), la surveillance est un dispositif passif et/ou dynamique, informationnel. La surveillance est un processus qui consiste à analyser l'état d'un système et à fournir des informations sur son fonctionnement. Elle permet notamment de détecter et de classer les défaillances, en observant l'évolution du système, puis à les diagnostiquer en localisant les éléments défaillants et en identifiant les causes premières, (Cordeschi, 2002), (Miessner, 1912).

L'objectif de la surveillance est de recueillir en permanence tous les signaux provenant du système de production, via des indicateurs, et de suivre en temps réel son évolution. Elle permet ainsi de contrôler le comportement du système par rapport à son fonctionnement normal, et de détecter toute anomalie ou défaillance. Dans ce cas, un diagnostic est établi afin de localiser le sous-système responsable du dysfonctionnement et d'identifier la cause de la défaillance, (Valette & Künzle, 1994), (Moulières-Seban, 2017). Cela permet de spécifier les opérations de maintenance à effectuer afin de remettre le système de production en état de fonctionnement, (Moulières-Seban, 2017), (David & Alla, 2010).

Chapitre 2 : Surveillance Industrielle

La surveillance est composée essentiellement de deux phases: la détection de défauts et le diagnostic de pannes. Le diagnostic se décompose en deux fonctions : la localisation et l'identification des causes.

2.3.1 Détection

Selon (Lefebvre, 2000), la détection des défaillances nécessite de pouvoir classer les situations observables comme étant normales ou anormales. Cette classification n'est pas triviale, car les situations anormales sont souvent caractérisées par un manque d'information. Une simplification communément adoptée consiste à considérer comme anormale toute situation qui n'est pas normale.

2.3.2 Diagnostic

Selon (Racoceanu, 2006), (Dubuisson, 2001) et (Isermann & Ballé, 1997), le système de diagnostic propose à l'opérateur les causes possibles de ce problème, ainsi que des interprétations floues de ces causes. L'objectif du système de diagnostic est de rechercher les causes et de localiser les organes qui ont entraîné une observation particulière. La fonction du diagnostic se décompose en deux fonctions élémentaires complémentaires : localisation et identification.

- La localisation consiste à déterminer le sous-ensemble fonctionnel défaillant, tandis que l'identification consiste à déterminer les causes internes ou externes à l'équipement. Cette étape peut être difficile, car les défaillances peuvent se propager d'un sous-système à l'autre.
- L'identification consiste à déterminer la cause de la défaillance. Cette étape peut également être difficile, car les causes peuvent être multiples et complexes.

2.4 Classification des Méthodes et Modèles de Surveillance

Dans ce chapitre, nous décrivons les méthodologies de surveillance industrielle et de diagnostic. Plusieurs travaux se sont intéressés à la surveillance industrielle, nous pouvons citer les travaux suivants : (Chatila, 2014), (David & Alla, 2010), (Cassandras & Lafortune, 2008), (Giua & Seatzu, 2007), (Thomas, 2005), (David & Alla, 1992a), (Chu, 1995), (Capek, 1920).

Selon les travaux suivants (Alhelou, 2019), (Skliros et al., 2019), (Soualhi et al., 2019), (Gertler, 2017), (Z. Gao et al., 2015), (Hwang et al., 2010), (David & Alla, 2010), (Racoceanu & Zerhouni, 2005), (Fabre et al., 2001) et (Tromp, 2000), les méthodes de surveillance sont généralement divisées en deux groupes : les méthodes de surveillance avec modèle et les méthodes sans modèle. Ces dernières sont différenciées par plusieurs critères : l'évolution de la dynamique du système (continu, discret ou hybride), la mise en place du système de surveillance (en ligne, hors ligne), la nature de l'information (quantitative, qualitative) et sa distribution (décentralisée ou centralisée). Le principe général de ces méthodes est de comparer les données relevées au cours du fonctionnement réel avec la connaissance que l'on a du fonctionnement nominal (détection) ou des fonctionnements défaillants (diagnostic : localisation et identification), (Sallak et al., 2005). Pour avoir plus de détails, nous référons le lecteur aux travaux suivants: (Basseville, 1988), (Chu, 1995), (David & Alla, 2010).

Après l'analyse et l'étude des travaux cités, nous proposons une classification globale des méthodes et modèles utilisés en surveillance industrielle, illustrée à travers la figure 2.2.

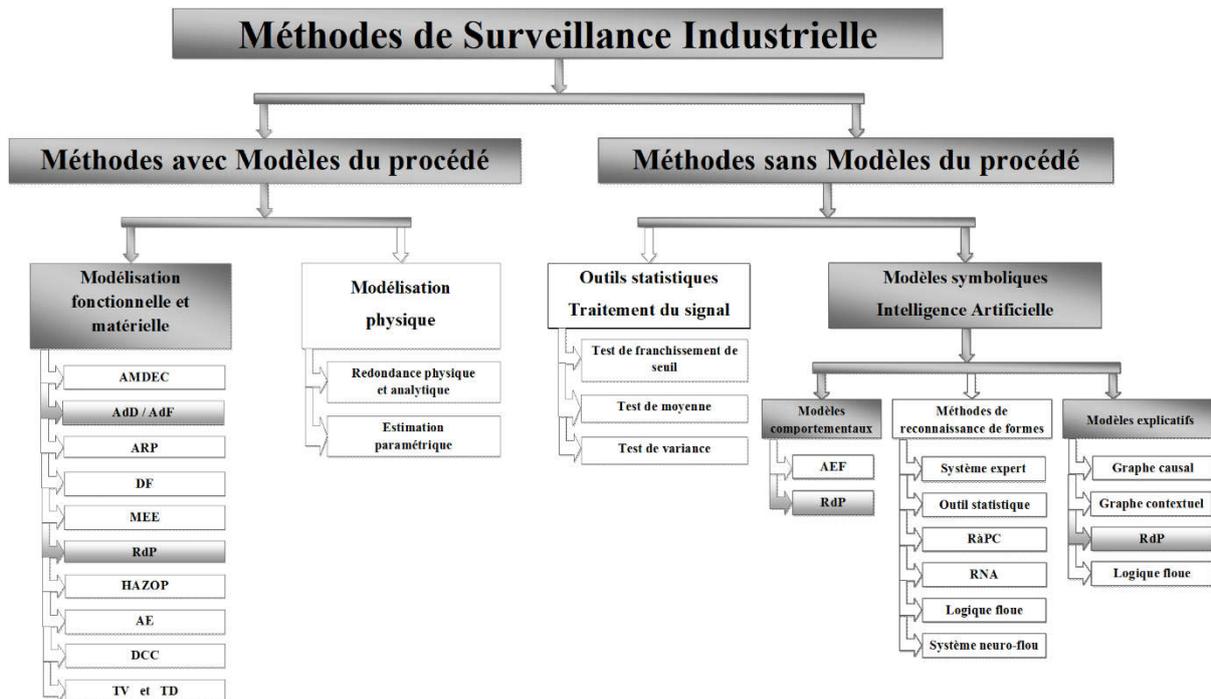


Figure 2. 2. Classification des méthodes et modèles de surveillance

2.4.1 Méthodes de Surveillance avec Modèles

Les Méthodes de surveillance avec modèles, sont les méthodes traditionnelles les plus utilisées par les industriels, elles se basent sur l'existence d'un modèle formel de l'équipement et utilisent généralement des techniques de l'automatique, (Guillot & Meyer, 2008), (Racoceanu, 2006), (Chu, 1995). On distingue généralement les méthodes par modélisation fonctionnelle et matérielle, et les méthodes de surveillance par modélisation physique.

2.4.1.1 Méthodes par Modélisations Fonctionnelle et Matérielle

Ces méthodes consistent à établir a priori et de la manière la plus complète possible, les liens entre les causes initiales des défaillances et leurs effets mesurables. Les méthodes les plus couramment rencontrées sont l'Analyse des Modes de Défaillance, de leurs Effets et de leurs Criticité (AMDEC) et les Arbres de Défaillances (AdD). Aussi il y'a d'autres méthodes utilisées: Analyse Préliminaire des Risques (APR), Diagramme de Fiabilité (DF), Méthode de l'Espace des États (MEE), Réseaux de Petri (RdP), HAZard and Operability Study (HAZOP), Méthodes de l'arbre d'Évènements (AE), Méthode du Diagramme Causes-Conséquences (DCC), Méthodes de la Table de Vérité (TV) et de la Table de Décision (TD). Ces méthodes sont principalement utilisées lorsque le nombre de données de retour d'expérience est suffisamment important, et utilisées essentiellement pour le diagnostic, (Mahdaoui, 2013).

2.4.1.1.1 Analyse des Modes de Défaillance, de leurs effets et de leur Criticité (AMDEC)

L'AMDEC (Analyse des Modes de Défaillance, de leurs effets et de leur criticité), est une méthode d'analyse préventive, qualitative et quantitative, utilisée dans la sûreté de fonctionnement qui permet d'identifier les modes de défaillances potentielles d'un produit, d'un procédé ou d'un système, d'évaluer leurs effets potentiels et de déterminer leur criticité, (Talebberrouane & Lounis, 2016), (Bukowski & Delaruelle, 2011), (R. Noureddine, 2008), (Bouissou & Dutuit, 2004).

Chapitre 2 : Surveillance Industrielle

Selon (Dutuit et al., 2018), (Mihalache, 2007a), (Garin, 1994) et (Villemeur, 1988b), l'AMDEC est une méthode complète et flexible qui peut être utilisée dans une grande variété d'applications. Dans le but d'analyser les conséquences des défaillances et à identifier les pannes dont les répercussions sur la sécurité sont importantes. L'analyse de criticité permet de classer les risques afin de prioriser les actions de réduction des risques les plus importants. L'AMDEC peut être mise en œuvre tout au long du cycle de vie d'un système. Elle est principalement utilisée comme technique d'analyse préventive pour détecter les défaillances potentielles, évaluer les risques, et susciter des actions de prévention. En fonction du système et des événements à considérer, d'autres champs peuvent être introduits dans l'AMDEC, tels que la fréquence ou la gravité. L'AMDEC est un outil qui permet l'analyse précise et exhaustive des relations de cause à effet entre les modes de défaillance des composants du système et leurs effets observables. Elle permet également d'évaluer ces relations. L'AMDEC est généralement réalisée en 5 étapes :

- 1) Identification des modes de défaillance : Cette étape consiste à identifier tous les modes de défaillance potentiels du produit, du procédé ou du système.
- 2) Analyse des effets des défaillances : Cette étape consiste à évaluer les effets potentiels de chaque mode de défaillance.
- 3) Détermination de la criticité : Cette étape consiste à déterminer la criticité de chaque mode de défaillance en fonction de sa probabilité d'occurrence, de sa sévérité et de sa détectabilité.
- 4) Planification des actions correctives : Cette étape consiste à planifier les actions correctives à mettre en œuvre pour réduire la criticité des modes de défaillance.
- 5) Suivi et mise à jour : Cette étape consiste à suivre les actions correctives mises en œuvre et à mettre à jour l'AMDEC en fonction des nouveaux résultats.

L'AMDEC est un outil puissant qui peut contribuer à améliorer la fiabilité, la sécurité et la qualité des produits, des procédés et des systèmes. Elle est particulièrement efficace pour l'analyse de défaillances locales d'éléments conduisant à la défaillance globale du système. En effet, elle permet d'identifier les modes communs de défaillances pouvant affecter le système.

2.4.1.1.2 Arbre de défaillance (AdD) / Arbre des Fautes (AdF)

L'arbre de défaillances (AdD) ou l'arbre des fautes (AdF), est une méthode d'analyse déductive qualitative et quantitative utilisée dans la sûreté de fonctionnement, (Ghostine, 2008), (R. Noureddine et al., 2005), (Villemeur, 1988b). C'est l'une des méthodes d'analyse de la fiabilité, de la disponibilité, de la maintenance et de la sécurité des systèmes la plus largement utilisée (Ruijters et al., 2019), (Dutuit et al., 2018). C'est un outil puissant qui permet d'identifier les causes profondes des défaillances permettant de développer des stratégies de maintenance préventive et par conséquent la réduction de coûts de maintenance (Gharahasanlou et al., 2014), (Purba, 2014). L'AdF est la technique la plus simple et la plus utilisée pour évaluer la fiabilité, (Bouissou & Dutuit, 2004), (Talebberrouane & Lounis, 2016), (Talebberrouane et al., 2016).

L'arbre de défaillances (AdD), permet de représenter graphiquement par un diagramme arborescent les causes et les effets d'une défaillance, et qui consiste à partir d'un événement redouté (ER) à identifier les causes qui peuvent le conduire. Ces causes sont ensuite décomposées en causes plus élémentaires, jusqu'à ce que l'on atteigne des causes qui ne peuvent pas être décomposées plus loin. Les causes sont représentées par des nœuds de l'arbre, et les relations de cause à effet entre les causes sont représentées par des portes logiques (ET, OU, NON, XOR), (Palluat, 2006). L'identification de l'événement redouté est une étape importante, car elle détermine l'étendue de l'analyse. Les causes de l'événement redouté peuvent être identifiées par une analyse des risques, une analyse fonctionnelle ou une analyse des modes de défaillance et de leurs effets (AMDEC), (Ruijters & Stoelinga, 2015), (Garin, 1994),

Chapitre 2 : Surveillance Industrielle

(Mihalache, 2007a). Au chapitre 4 de cette thèse, nous avons utilisé cet outil et lié aux travaux de (Mehar et al., 2023) et (Mehar et al., 2022).

2.4.1.1.3 Analyse Préliminaire des Risques (APR)

Selon (Bukowski & Delaruelle, 2011), (Mihalache, 2007a), (Bukowski & Delaruelle, 2011), (Zwingelstein, 1996) et (Morel et al., 1992), l'Analyse Préliminaire des Risques (APR) ou bien l'Analyse Préliminaire des Dangers (APD), est une méthode d'analyse qualitative utilisée pour identifier des risques d'un système complexe et évaluer la gravité des conséquences liées à ces risques. Elle est également utilisée aussi pour l'identification des risques au stade préliminaire de la conception d'une installation ou d'un projet, d'un produit ou d'un processus. Elle est généralement réalisée au début du cycle de vie du projet, afin d'identifier les risques les plus importants et de mettre en place des mesures de mitigation. L'ARP ne nécessite pas une connaissance approfondie et détaillée de l'installation étudiée.

Selon (Mihalache, 2007a) et (Hadj-Mabrouk, 2006), l'APR se déroule en plusieurs étapes : Identification des dangers ; Évaluation des risques ; Classification des risques ; Planification des actions correctives. L'APR est une méthode simple et rapide qui permet d'identifier rapidement les risques les plus importants d'un projet. Elle est donc un outil précieux pour la gestion des risques.

2.4.1.1.4 Diagramme de Fiabilité (DF)

Selon les travaux (Weber & Simon, 2015), (Cocozza-Thivent, 1997), (Frankel, 1988), (Villemeur, 1988a) et (Pagès & Gondran, 1980), un diagramme de fiabilité (DF) est une analyse qualitative et quantitative, qui permet d'analyser et de calculer la fiabilité d'un système ou d'un composant au cours du temps. Il est généralement utilisé pour évaluer la fiabilité d'un système avant sa mise en service ou pour suivre la fiabilité d'un système en service.

Selon (Mihalache, 2007b) et (Noyes & Pérès, 2007), le DF est construit en représentant l'architecture du système sous forme de blocs. Ces blocs peuvent représenter des composants, des sous-systèmes ou des fonctions. Les relations entre les blocs sont représentées par des flèches. Ces flèches représentent les relations de cause à effet entre les blocs. Les relations de cause à effet peuvent être représentées par des portes logiques, telles que les portes AND, OR, NOT et XOR.

Les données de fiabilité des différents composants du système sont utilisées pour calculer la fiabilité du système. Ces données peuvent être obtenues à partir de tests, de données historiques ou d'estimations. À partir des DF on peut généralement représenter des courbes. La courbe de fiabilité représente la probabilité qu'un système fonctionne sans défaillance à un instant donné. La courbe de disponibilité représente la probabilité qu'un système soit disponible pour fonctionner à un instant donné, (Habchi, 2018).

2.4.1.1.5 Méthode de l'Espace des États (MEE)

Selon les travaux (Demri, 2009), (Birolini, 1997), (Villemeur, 1988b), (Frankel, 1988) et (Pagès & Gondran, 1980), la Méthode de l'Espace des États (MEE) (ou processus de Markov) est une méthode analytique utilisée pour l'analyse de fiabilité et l'analyse de la sûreté de fonctionnement des systèmes réparables. Elle permet de décrire l'évolution du système dans le temps en fonction de ses états fonctionnels et de ses taux de transition entre ces états. La MEE est une méthode puissante qui permet d'obtenir des informations quantitatives sur la sûreté de fonctionnement des systèmes réparables. La MEE peut être appliquée à des systèmes de toute

Chapitre 2 : Surveillance Industrielle

taille et de toute complexité. Elle est cependant limitée par le fait qu'elle suppose que les taux de transition sont constants dans le temps, (Benchouia et al., 2017), (Schoenig et al., 2006), (Frankel, 1988).

Le principal avantage de cette méthode MEE est sa capacité de modéliser des systèmes réparables. Cette méthode peut être difficile à appliquer pour des systèmes de grande taille, car elle conduit à une explosion des états. La MEE se compose de quatre étapes principales : Recensement des états fonctionnels ; Définition des taux de transition ; Élaboration du diagramme d'états ; Calcul des probabilités d'état, (Munaweera et al., 2021).

2.4.1.1.6 Réseaux de Petri (RdP)

Selon (Petri, 1962) et (Brauer et al., 1985), les réseaux de Petri (RdP) sont des outils graphiques et mathématiques, qui permettent de décrire des relations entre des conditions et des événements, et de modéliser le comportement de systèmes dynamiques à événements discrets. Ils permettent l'analyse qualitative et quantitative.

Nous décrivons plus en détail la théorie des réseaux de Petri (RdP) au chapitre 3. Au chapitre 4, nous avons utilisé cet outil et lié à nos travaux de recherche (Mehar et al., 2023) et (Mehar et al., 2022).

2.4.1.1.7 HAZard and Operability Study (HAZOP)

Selon (Derbazi, 2023) et (Bukowski & Delaruelle, 2011), l'analyse des risques sur les schémas de type HAZOP pour HAZard Operability, est une méthode systématique pour identifier et évaluer les risques associés à un processus ou à un système. Elle est utilisée dans une variété d'industries, notamment la chimie, la pétrochimie, l'énergie et la fabrication.

Selon (El Hadj Yagoub, 2023) (Oeing et al., 2023), le processus HAZOP commence par la création d'un schéma détaillé du processus ou du système. Le schéma est ensuite divisé en sections, chacune représentant une étape ou un composant du processus. Pour chaque section, un groupe de travail constitué d'experts techniques identifie les écarts potentiels par rapport aux conditions nominales. Ces écarts sont appelés "déviations". L'HAZOP est une méthode centrée sur l'installation, contrairement à l'AMDE qui est centrée sur les composants. Le groupe de travail doit déterminer les causes et les conséquences potentielles de chacune de ces dérives. Il s'agit également d'identifier les moyens existants permettant de détecter, prévenir ou limiter les effets de la dérive. Le groupe de travail peut proposer des mesures correctives pour améliorer la sécurité.

L'analyse HAZOP est une méthode efficace pour identifier les risques potentiels associés aux processus et aux systèmes. Elle peut aider à prévenir les accidents et à améliorer la sécurité des personnes et des biens.

2.4.1.1.8 Arbre d'Évènements (AE)

Selon (El Hami & Eid, 2023), (Bukowski & Delaruelle, 2011) et (Mazouni, 2008), l'arbre d'évènements (AE) est une méthode d'analyse quantitative et qualitative des risques qui permet de visualiser et d'évaluer les différentes séquences d'évènements qui peuvent conduire à un accident. Il est souvent utilisé pour l'analyse de la sûreté des systèmes complexes et les événements initiateurs, tels que les centrales nucléaires, les raffineries de pétrole ou les usines chimiques. Cette méthode est une dérivée de la méthode des arbres de décision, utilisée en analyse de décision.

Chapitre 2 : Surveillance Industrielle

L'arbre d'événements est une méthode puissante qui peut être utilisée pour identifier et évaluer les risques associés aux systèmes complexes. Elle est cependant une méthode complexe qui nécessite une expertise technique.

2.4.1.1.9 Méthode du Diagramme Causes-Conséquences (DCC)

Selon (Omrane et al., 2018), (Hoq, 2014) et (Palluat et al., 2004), la méthode du Diagramme Causes-Conséquences (DCC) est une méthode d'analyse des risques qui combine les logiques inductives de l'Arbre des Événements (AE) et déductives de l'arbre des défaillances des défaillances (AdD).

Le DCC est une méthode simple et efficace qui peut être utilisée pour identifier et évaluer les risques associés à une variété d'événements. Elle est souvent utilisée pour l'analyse des risques des systèmes industriels, des bâtiments et des infrastructures. La construction d'un DCC suit les règles cumulées des AdD et AE et permet d'obtenir des résultats qualitatifs et quantitatifs. Les causes sont représentées par des flèches qui pointent vers l'événement, et les conséquences sont représentées par des flèches qui pointent depuis l'événement.

2.4.1.1.10 Méthodes de la Table de Vérité (TV) et de la Table de Décision (TD)

Selon (Jubin, 2019) et (Palluat et al., 2004), les méthodes de la table de vérité (TV) et de la table de décision (TD) sont des méthodes logiques qui permettent de représenter et d'analyser des situations complexes. Elles sont utilisées dans une variété de domaines, notamment l'ingénierie, les mathématiques, l'informatique et la science.

Selon (Jubin, 2019) et (Palluat et al., 2004), la table de vérité (TV) est une méthode qui permet d'identifier tous les états possibles (fonctionnement ou panne) d'un système à partir de comportements binaires. Elle est limitée aux cas simples pour l'analyse manuelle, mais peut être automatisée, basée sur l'algèbre booléenne. La construction de la table est réalisée après identification des AMDE des composants. Elle est construite en associant une variable à chaque état possible du système. Les variables peuvent prendre deux valeurs, 0 (fonctionnement) ou 1 (panne). La table de vérité est limitée aux systèmes comportant peu de composants, car le nombre de combinaisons possibles devient important quand le nombre de composants augmente.

Selon (Jubin, 2019) et (Palluat et al., 2004), la table de décision (TD) est une extension de la table de vérité qui permet de traiter un nombre quelconque d'états pour chaque entrée de la table. Elle peut servir à l'élaboration d'un arbre de défaillance (AdD). Elle est construite en associant une variable à chaque état possible de chaque composant du système. Les variables peuvent prendre un nombre quelconque de valeurs, correspondant aux différents états possibles du composant. Elle est utilisée dans l'analyse des risques pour identifier les différentes causes possibles d'un événement indésirable.

La relation entre les méthodes de la table de vérité (TV) et de la table de décision (TD) est une relation de parenté. La table de vérité est une méthode plus simple et plus limitée, tandis que la table de décision est une extension de la table de vérité qui permet de traiter des cas plus complexes. La table de décision est une méthode plus puissante que la table de vérité.

2.4.1.2 Méthodes de Surveillance par Modélisation Physique

Ces méthodes ont pour principe de comparer les mesures effectuées sur le système aux informations fournies par le modèle (Frank, 1990), (Zemouri, 2003), (R. Nouredine, 2008), (F. Nouredine, 1996). Tout écart est alors synonyme d'une défaillance. Les outils de la théorie

Chapitre 2 : Surveillance Industrielle

de la décision sont ensuite utilisés pour déterminer si cet écart est dû à des aléas normaux ou à une défaillance du système. Les méthodes de surveillance par modélisation physique sont les méthodes les plus familières aux automaticiens. En règle générale, elles peuvent être divisées en deux techniques : les techniques de redondances physique et analytique, et les techniques d'estimation paramétrique.

2.4.1.2.1 Redondances Physiques et Analytiques

Selon (Palluat, 2006), (Zemouri, 2003), (Zhang, 1999) et (Frank, 1990), les redondances physiques et analytiques sont deux techniques de surveillance par modélisation physique. Elles diffèrent par la manière dont elles utilisent le modèle physique. Elles sont complémentaires et peuvent être utilisées ensemble pour améliorer la fiabilité de la surveillance.

La redondance physique consiste à utiliser plusieurs capteurs pour mesurer la même variable. Si les mesures de deux capteurs différents sont différentes, cela indique qu'il y a un problème avec le système. Par exemple, un système de contrôle de température peut utiliser deux capteurs de température pour mesurer la température d'un fluide. Si les deux capteurs indiquent des températures différentes, cela indique qu'il y a un problème avec le système, tel qu'une défaillance d'un capteur ou une variation de la température du fluide, (Mahdaoui, 2013), (Zemouri, 2003).

La redondance analytique utilise un modèle physique pour prédire les valeurs des variables du système. Si les valeurs mesurées des variables sont différentes des valeurs prédites, cela indique qu'il y a un problème avec le système. Ces méthodes fournissent une estimation de l'état du système. Par exemple, un système de contrôle de vitesse peut utiliser un modèle physique pour prédire la vitesse d'un véhicule. Si la vitesse mesurée du véhicule est différente de la vitesse prédite, cela indique qu'il y a un problème avec le système, tel qu'une défaillance d'un capteur ou un changement dans les conditions de conduite, (Mahdaoui, 2013), (Zemouri, 2003).

2.4.1.2.2 Estimation Paramétrique

Selon (Mahdaoui, 2013), (Palluat, 2006), (Zemouri, 2003), (Isermann, 1984) et (Willsky, 1976), l'estimation paramétrique est une technique utilisée pour estimer les paramètres d'un modèle mathématique, décrivant le comportement d'un système ou d'un processus. Elle suppose l'existence d'un modèle paramétrique et la connaissance des valeurs de ses paramètres en fonctionnement nominal. Pour identifier les paramètres caractérisant le fonctionnement réel à partir de mesures des entrées et des sorties du système. On compare les mesures prises sur le système à ses valeurs théoriques. On obtient ainsi une estimation des paramètres du modèle.

2.4.2 Méthodes de Surveillance sans Modèles

De nombreuses applications industrielles ne disposent pas de modèle, soit parce que le système est trop complexe, soit parce qu'il est reconfiguré fréquemment. Pour ce type d'applications industrielles, les seules méthodes de surveillance opérationnelles sont les méthodes sans modèle, (Racoceanu, 2006), (Zemouri, 2003), (Dubuisson, 2001). Elles se divisent en deux familles : les outils statistiques de traitement du signal et les techniques d'intelligence artificielle (IA).

2.4.2.1 Surveillance basée sur des Outils Statistiques du Traitement du Signal

La surveillance basée sur des outils statistiques du traitement du signal est une méthode de surveillance des défaillances qui utilise des tests statistiques pour détecter les anomalies dans

Chapitre 2 : Surveillance Industrielle

les signaux mesurés. Elle repose sur l'hypothèse que les signaux des capteurs possèdent certaines propriétés statistiques, et des tests sont effectués pour vérifier la présence de ces propriétés dans un échantillon de signaux mesurés. Ces techniques sont généralement simples à mettre en œuvre, mais elles ne fournissent pas d'informations sur la nature des défaillances, voir (Senoussi, 2015), (Mahdaoui, 2013), (Palluat, 2006), (Zemouri, 2003). On distingue : le test de franchissement de seuils, le test de moyenne et le test de variance.

2.4.2.1.1 Test de Franchissement de Seuil

Selon (Palluat, 2006) et (Zemouri, 2003), le test de franchissement de seuil est une méthode de surveillance des défaillances qui consiste à comparer ponctuellement les signaux à des seuils préétablis. Le seuil est une valeur qui représente la limite acceptable pour la variable. Le franchissement d'un seuil par un des signaux génère une alarme. Dans l'industrie il existe deux types de seuils : Le seuil de pré-alarme, qui permet d'entreprendre une action de maintenance préventive. Le seuil d'alarme, qui impose l'arrêt de la production et l'engagement d'une action de maintenance corrective.

2.4.2.1.2 Test de Moyenne

Selon (Palluat, 2006) et (Zemouri, 2003), le test de moyenne est une méthode de test statistique qui permet de déterminer si la moyenne d'une population est égale à une valeur donnée. Il est basé sur la distribution normale, qui est une distribution de probabilité qui décrit la distribution de valeurs d'une variable aléatoire. Le test de comparaison est effectué sur la moyenne d'un signal contenu dans une fenêtre de n valeurs, plutôt que sur une valeur ponctuelle.

2.4.2.1.3 Test de Variance

Selon (Palluat, 2006) et (Zemouri, 2003), le test de variance est une méthode statistique qui permet de vérifier si la variance d'un signal est statistiquement différente d'une valeur attendue. La variance d'un signal peut être calculée pour vérifier si elle est conforme à une valeur attendue. Si la variance est trop élevée ou trop faible, cela peut être un signe de défaillance. Par exemple, un test de variance peut être utilisé pour détecter une défaillance qui se manifeste par une augmentation ou une diminution de la puissance d'un signal.

2.4.2.2 Surveillance par Intelligence Artificielle (IA)

Différentes catégorisations des méthodes de surveillance par l'Intelligence Artificielle (IA) existent, chacune présentant des caractéristiques adaptées à des problèmes de surveillance spécifiques, voir (Senoussi, 2015), (Mahdaoui, 2013), (Racoceanu, 2006), (Palluat, 2006) (Racoceanu & Zerhouni, 2005), (Aghasaryan, 1998) et (Basseville & Cordier, 1996). D'une manière générale, on peut classer ces méthodes selon 3 types : les modèles comportementaux, les méthodes de reconnaissance de formes et les modèles explicatifs.

2.4.2.2.1 Modèles Comportementaux

Selon les travaux (Senoussi, 2015), (Mahdaoui, 2013), (Ikhlef, 2009), (Racoceanu, 2006) et (Palluat, 2006), les modèles comportementaux sont des modèles qui simulent le comportement d'un système à partir d'une modélisation de son comportement. Contrairement aux modèles numériques basés sur la physique du système, les modèles comportementaux ne s'appuient pas sur des lois physiques mais plutôt sur une analyse des modes de fonctionnement du système. Ces méthodes reposent sur des outils tels que les réseaux de Petri et les automates d'états finis.

Chapitre 2 : Surveillance Industrielle

2.4.2.2.1.1 Automates d'États Finis (AEF)

Selon (Zemouri, 2003) et (Palluat, 2006), les automates d'états finis (AEF) sont des outils mathématiques et graphiques utilisés pour modéliser le fonctionnement d'un système global à partir de ses composants. Ils permettent de représenter le comportement du système en définissant ses états possibles et les transitions entre ces états. Cette représentation est particulièrement adaptée à la simulation et à la détection d'anomalies, et peut même être utilisée à des fins de diagnostic dans certains cas.

D'après (Ikhlef, 2009), la méthode de surveillance est donc basée sur la modélisation du système à l'aide des automates d'états finis, et sur la construction d'un diagnostiqueur correspondant également à un automate d'états finis. Cette approche permet de détecter les pannes en observant en ligne une séquence d'événements et de déduire les occurrences des pannes à partir des estimations de l'état du système et des événements non observés.

2.4.2.2.1.2 Réseaux de Petri (RdP)

Suivant (Zemouri, 2003) et (Palluat, 2006), la deuxième approche majeure utilisant les méthodes à base de modèles opérationnels est basée sur le formalisme des réseaux de Petri (RdP). Les réseaux de Petri sont des outils mathématiques et graphiques qui permettent de modéliser des systèmes à événements discrets. Ils sont utilisés pour détecter les anomalies dans les signaux mesurés et pour mieux rendre compte de la dynamique des systèmes à événements discrets. Les RdP ont été enrichis sous plusieurs aspects, tels que les RdP temporisés, stochastiques et flous. Ils sont utilisés comme modèles générateurs pour la réalisation de simulations et la construction de mécanismes de détection dans une optique d'utilisation en diagnostic de systèmes. En effet, les RdP sont considérés comme un outil puissant de modélisation pour décrire la connaissance nécessaire au diagnostic.

Nous présentons en détail les réseaux de Petri (RdP) au chapitre 3, et au chapitre 4, nous avons exploité ces outils et liés à nos travaux de recherche (Mehar et al., 2023) (Mehar et al., 2022).

2.4.2.2.2 Méthodes de Reconnaissance de Formes

Selon les travaux (Soualhi et al., 2019), (Senoussi, 2015), (Mahdaoui, 2013), (Racoceanu, 2006) et (Palluat, 2006), les méthodes de reconnaissance de forme regroupent les modèles associatifs et les méthodes de reconnaissances. Elles sont caractérisées par l'apprentissage et la reconnaissance, des termes qui s'appliquent aussi bien aux systèmes de reconnaissance de formes qu'aux systèmes à base de règles tels que les systèmes experts. Ces techniques sont plus élaborées que les simples tests statistiques et sont capables de détecter et de diagnostiquer les défaillances. Les méthodes de reconnaissance de formes sont souvent basées sur des outils particuliers tels que les systèmes experts, les outils statistiques, le raisonnement à partir de cas (RàPC), les réseaux de neurones, la logique floue, et les systèmes Neuro-Flous.

2.4.2.2.2.1 Système Expert

Suivant (Soualhi et al., 2019), (Guo et al., 2019), (Senoussi, 2015), (Palluat, 2006), (Racoceanu, 2006), (Racoceanu & Zerhouni, 2005), (Angeli & Chatzinikolaou, 2004), (Saludes et al., 2003) et (Zemouri, 2003), un système expert est un programme informatique qui simule le raisonnement d'un expert humain accomplissant une tâche intellectuelle dans un domaine précis. Il est composé de trois éléments principaux :

Chapitre 2 : Surveillance Industrielle

- La base de connaissances contient les informations sur lesquelles le système expert s'appuie pour prendre ses décisions. Elle est généralement constituée de faits, de règles et de procédures.
- Le moteur d'inférence est le composant du système expert qui utilise la base de connaissances pour prendre des décisions. Il utilise un ensemble de règles pour déduire des conclusions à partir des faits connus.
- L'interface utilisateur permet à l'utilisateur d'interagir avec le système expert. Elle permet à l'utilisateur de fournir des informations au système expert et de recevoir ses réponses.

Selon (Racoceanu, 2006), les systèmes experts sont subordonnés à une base de règles logiques qui représente l'ensemble des combinaisons importantes d'états du système analysé. Ils permettent l'exploitation d'une base de données informationnelle considérable, mais présentent le désavantage d'un comportement rigide dans un contexte variable et évolutif.

2.4.2.2.2 Outil Statistique

Selon les travaux (Senoussi, 2015), (Racoceanu, 2006), (Palluat, 2006), (Racoceanu & Zerhouni, 2005) et (Zemouri, 2003), les outils statistiques sont utilisés pour la surveillance des défaillances en se basant sur la probabilité de survenue d'une défaillance. Ces méthodes supposent que l'on connaît la distribution des données pour chaque état de fonctionnement du système. Les outils statistiques de reconnaissance de formes présentent de nombreux avantages. Ils sont souvent relativement simples à mettre en œuvre et peuvent être utilisés pour traiter des données de grande taille. Cependant, ils peuvent également être sensibles au bruit et aux variations des données.

2.4.2.2.3 Raisonnement à Partir de Cas (RàPC)

Selon les travaux (Senoussi, 2015), (Ikhlef, 2009), (Palluat, 2006), (Racoceanu, 2006), (Racoceanu & Zerhouni, 2005) et (Zemouri, 2003), le raisonnement à partir de cas (RàPC) est une technique d'intelligence artificielle qui permet de résoudre des problèmes en se basant sur l'expérience passée, en réutilisant les informations et les connaissances de situations similaires déjà résolues. Alors il constitue une méthodologie pour modéliser le raisonnement, et la pensée humaine ou pour développer des systèmes intelligents. Le RàPC est souvent utilisé dans des domaines où il est difficile ou coûteux de développer des règles générales, tels que la médecine, la finance ou le droit. Le RàPC se déroule en trois étapes principales :

- Collecte et stockage des cas : les cas sont collectés à partir de sources diverses, telles que des bases de données, des documents ou des interviews d'experts.
- Correspondance de similarité : les cas sont comparés au cas à résoudre pour identifier les cas les plus similaires.
- Solution transfert : la solution du cas le plus similaire est appliquée au cas à résoudre.

2.4.2.2.4 Réseaux de neuronaux (RNA)

Selon les travaux (Orlianges et al., 2024), (Zappone et al., 2019), (R. Liu et al., 2018), (Msaaf & Belmajdoub, 2015), (Ben Ali et al., 2015), (Srivastava et al., 2014), (Urmos et al., 2013), (Z. Yang et al., 2011), (Moshou et al., 2010), (Kurban & Beşdok, 2009), (Cottrell et al., 2009), (Racoceanu, 2006), (Palluat, 2006) et (Racoceanu & Zerhouni, 2005), les réseaux neuronaux (RNA) sont des technique d'apprentissage automatique qui permettent aux algorithmes de reconnaître des caractéristiques et des objets dans des données, telles que des images, des vidéos, et même du texte ou des signaux sensoriels. Ils sont inspirés du fonctionnement du cerveau humain et capables de traiter des informations de manière non linéaire, et d'effectuer

Chapitre 2 : Surveillance Industrielle

des opérations de perception, classification et prédiction. Les RNA sont capables d'apprendre à partir de données de manière autonome, sans programmation explicite, et sont adaptés pour la mise au point de systèmes de surveillance modernes, capables de s'adapter à un système complexe avec reconfigurations multiples. Les RNA peuvent être implémentés en circuits électroniques, offrant ainsi la possibilité d'un traitement temps réel.

L'utilisation des RNA est principalement guidée par plusieurs propriétés : la capacité d'apprentissage, la capacité de généralisation, parallélisme dans le traitement (rapidité de traitement), les RNA peuvent modéliser des relations non linéaires complexes entre les données, ce qui les rend adaptés à une large gamme de problèmes. Les réseaux neuronaux peuvent être constitués de plusieurs couches de neurones. Chaque neurone reçoit des entrées d'autres neurones et calcule une sortie en fonction de ces entrées. La sortie d'un neurone est ensuite transmise aux neurones de la couche suivante. Chaque neurone réalise une fonction simple (fonction linéaire, linéaire par morceaux, fonction à seuil, sigmoïde, gaussienne), les propriétés globales de l'outil émergent de sa structure. On distingue les types de réseaux neuronaux suivants :

- Réseaux neuronaux perceptron multicouches (MLP : Multilayer Perceptron Networks) ;
- Réseaux neuronaux convolutifs (CNN : Convolutional Neural Networks) ;
- Réseaux neuronaux récurrents (RNN : Recurrent Neural Networks) ;
- Réseaux neuronaux génératifs adversatifs (GAN : Generative Adversarial Networks) ;
- Réseaux de neurones à impulsions (SNN : Spiking Neural Networks) ;
- Réseaux de neurones profonds (DNNs : Deep neural networks).

2.4.2.2.2.5 Logique Floue

Selon les travaux (Bouchon-Meunier & Marsala, 2003), (Racoceanu, 2006), (Palluat, 2006), (Bouchon-Meunier & Marsala, 2003), (Mellouli & Bouchon-Meunier, 2000), (Cheng, 2000) et (Rahamani et al., 1998), la logique floue est une forme de logique qui permet de représenter des connaissances imprécises ou approximatives. Elle offre la possibilité de traiter des systèmes d'une grande complexité, notamment ceux dans lesquels sont présents des facteurs humains. Elle est souvent utilisée dans des domaines où l'incertitude est importante, tels que l'aide à la décision ou le diagnostic. La reconnaissance des formes par logique floue est une technique de reconnaissance des formes qui permet de représenter et de reconnaître des formes de manière plus flexible que les techniques traditionnelles.

2.4.2.2.2.6 Système Neuro-Flou

Selon les travaux (Racoceanu, 2006), (Palluat, 2006), (J.-S. Wang & Lee, 2002), (Uppal et al., 2002), (Palade et al., 2002), (Ould Abdeslam, 2002), (C. S. G. Lee & Wang, 2001), (C. S. G. Lee & Wang, 2000) et (Nauck & Kruse, 1998), un système neuro-floue est un type de système d'intelligence artificielle qui combine les avantages des techniques de logique floue et de réseaux de neurones, et offrent de nombreuses possibilités d'applications. Les systèmes neuro-flous sont des techniques prometteuses dans la modélisation, la reconnaissance de formes, la classification, la prise de décision, le diagnostic et le contrôle. Ils permettent de traiter des données imprécises ou approximatives aussi.

Les systèmes neuro-flous sont capables de modéliser des systèmes dynamiques non linéaires complexes, même si le modèle mathématique du système est inconnu. Ils sont également capables d'obtenir des résultats possédant un certain niveau d'interprétation. La plupart des applications des systèmes neuro-flous sont basées sur l'établissement d'un diagnostic à partir de la classification de résidus. Cependant, cette approche nécessite de pouvoir établir un modèle

Chapitre 2 : Surveillance Industrielle

du système. De plus, elle nécessite un réseau par résidus, ce qui rend le système relativement complexe et ne permet de diagnostiquer qu'un nombre restreint de défauts. Une solution à ces limitations serait d'employer des systèmes Neuro-Flou sans modèle du système à diagnostiquer. Cela permettrait de simplifier la conception des systèmes et de les rendre plus adaptables à de nouveaux défauts.

Différentes combinaisons de ces deux techniques de l'IA (la logique floue et les réseaux neuronaux) existent et mettent en avant des propriétés différentes. On peut identifier les combinaisons suivantes :

- Réseau flou neuronal.
- Système neuronal/flou simultanément.
- Modèles Neuro-Flous coopératifs.
- Modèles Neuro-Flous hybrides.

2.4.2.2.3 Méthodes Explicatives

Selon les travaux (Mahdaoui, 2013), (Racoceanu, 2006), (Palluat, 2006) et (Basseville & Cordier, 1996b), les modèles explicatifs qui sont principalement basés sur la représentation des relations entre les différents états de pannes et leurs effets, nécessitent une analyse approfondie du système, pour exprimer ces relations de cause à effet. Certains modèles obtenus permettent une approche adductive, remontant aux causes des pannes à partir des observations des symptômes. Ce sont des modèles qui donnent une représentation d'une analyse causale du système à diagnostiquer. On distingue ici les graphes causaux et les graphes contextuels, mais aussi les réseaux de Petri et la logique floue.

2.4.2.2.3.1 Graphes Causaux

Selon les travaux (Y. J. Liu et al., 2016), (F. Yang et al., 2012), (Hua et al., 2011), (D. Gao et al., 2010), (Racoceanu, 2006), (Grosclaude, 2001a), (Grosclaude et al., 2001b), (Grosclaude & Quiniou, 2000), (Brusoni et al., 1997), (Basseville & Cordier, 1996a) et (Brusoni et al., 1995), les graphes causaux sont un formalisme bien adapté à la représentation des liens causaux dans une utilisation de diagnostic. Ils permettent d'exprimer les enchaînements causaux régissant le fonctionnement du système à surveiller en cas de panne. Ce sont des graphes orientés acycliques, dont les nœuds correspondent aux causes et aux effets, et les arcs orientés aux relations causales entre les deux.

Les graphes causaux sont des outils à base de modèles explicatifs. Ils fournissent une explication à des observations d'un fonctionnement anormal d'un système. Ils représentent des connaissances profondes sur le système, capable de donner les relations causales entre les différents états de pannes. La construction du modèle causal repose sur une expertise, telle que des catalogues de pannes, des AMDEC ou des AdD, qui expriment pour une panne l'ensemble des symptômes observables.

2.4.2.2.3.2 Graphes Contextuels

Selon les travaux (Racoceanu, 2006), (Palluat, 2006), (Brézillon, 2003), (Pasquier, 2002) et (Brézillon et al., 2001), les graphes contextuels sont une technique d'apprentissage automatique et un formalisme de modélisation des connaissances permettant de représenter et d'analyser des données relationnelles, et aussi de représenter les différentes méthodes permettant d'atteindre un but en fonction du contexte dans lequel ils s'appliquent. Ils sont utilisés dans de nombreux domaines, notamment la médecine, la finance, le marketing, la gestion d'incidents dans les transports et l'industrie. Les graphes contextuels sont un formalisme bien adapté pour l'aide à

Chapitre 2 : Surveillance Industrielle

la décision dans de nombreux domaines. Ils permettent de modéliser de manière explicite le contexte dans lequel la décision est prise, de représenter les différentes méthodes pour atteindre un but et d'être flexibles et adaptables.

2.5 Conclusion

La surveillance industrielle est une phase essentielle et impérative pour la sûreté de fonctionnement et en particulier pour la maintenance conditionnelle et prédictive. Elle contribue à améliorer la performance et la fiabilité des équipements.

Dans ce chapitre, après les notions de base en maintenance, on a présenté un état de l'art et une classification des méthodes et modèles de surveillance industrielle. On a abordé les techniques les plus usitées dans ce domaine. L'utilisation des techniques avec modèles, qui sont des techniques puissantes et très précises, s'avère difficile, coûteuse et offrant de moins en moins de satisfaction vis-à-vis des systèmes de plus en plus complexes. L'utilisation des techniques sans modèles, plus simple et moins coûteuses, s'avère plus adaptée à cette problématique et, plus particulièrement les techniques liées à l'intelligence artificielle. Ces techniques sont prometteuses et les recherches futures dans ce domaine pourraient contribuer à développer des systèmes de surveillance encore plus performants.

Chapitre 3 : Théorie des Réseaux de Petri

3.1	Introduction	38
3.2	Historique	38
3.3	État de l'art RdP et Surveillance Industrielle	38
3.4	Réseaux de Petri	40
3.5	Notions de base des réseaux de Petri	40
3.6	Propriétés Mathématiques	42
3.7	Propriétés Génériques	44
3.7.1	RdP borné.....	44
3.7.2	RdP sauf.....	45
3.7.3	Vivacité	45
3.7.4	Blocage	45
3.7.5	Réinitialisabilité	45
3.7.6	Conflit	45
3.7.7	Accessibilité	45
3.7.8	Sûreté	46
3.8	Analyse des réseaux de Petri	46
3.8.1	Analyse par Graphes des Marquages	46
3.8.2	Analyse par Arbre de Couverture et Graphe de Couverture.....	46
3.8.3	Analyse par Algèbre Linéaire	47
3.8.4	Analyse par Réduction	47
3.9	Structures particulières des RdP	47
3.9.1	Graphe d'États	48
3.9.2	Graphe d'Événement.....	48
3.9.3	RdP sans Conflit	48
3.9.4	RdP Conflit Structurel.....	48
3.9.5	RdP à Choix Libre	49
3.9.6	RdP Simple	49
3.9.7	RdP Pur	49
3.9.8	RdP sans Boucle	49
3.9.9	RdP Généralisé.....	49
3.9.10	RdP à Capacité.....	49
3.9.11	RdP Autonome.....	50
3.9.12	RdP non Autonome.....	50
3.9.13	RdP à Priorité.....	50

3.9.14	RdP à Arcs Inhibiteurs	50
3.10	Avantages et inconvénients des réseaux de Petri généralisés et particuliers.....	51
3.10.1	Avantages.....	51
3.10.2	Inconvénients	51
3.11	Réseaux de Petri de Haut Niveau	52
3.11.1	Réseaux de Petri Synchronisés (RdPS).....	52
3.11.2	Réseaux de Petri Temporisés (RdPT)	52
3.11.2.1	Réseaux de Petri Temporisés	52
3.11.2.1.1	Réseaux de Petri T-temporisés (RdPTt)	52
3.11.2.1.2	Réseaux de Petri P-temporisés (RdPPt).....	53
3.11.2.2	Réseaux de Petri Temporels	53
3.11.2.2.1	Réseaux de Petri T-temporels	53
3.11.2.2.2	Réseaux de Petri P-temporels	53
3.11.2.2.3	Réseaux de Petri arc-pt-temporels	54
3.11.3	Réseaux de Petri Stochastiques (SPN).....	54
3.11.3.1	Réseaux de Petri Stochastiques Généralisés (GSPN)	55
3.11.3.2	Réseaux de Petri Stochastiques et Déterministes (DSPN).....	55
3.11.3.3	Réseaux de Petri Stochastiques Étendus (ESPN).....	55
3.11.4	Réseaux de Petri Colorés (RCP).....	55
3.11.5	Réseaux de Petri Interprétés (RdPI).....	55
3.11.6	Réseaux de Petri Algébriques (APN)	56
3.11.7	Réseaux de Petri Flous (RdPF)	56
3.11.7.1	RdPF basés sur des règles floues.....	57
3.11.7.2	RdPF basés sur la logique linéaire floue	57
3.11.8	Réseaux de Petri Hiérarchiques (HPN).....	57
3.11.9	Réseaux de Petri Continus (RdPC).....	57
3.11.10	Réseaux de Petri Hybrides (RdPH).....	58
3.12	Conclusion	58

Chapitre 3 : Théorie des Réseaux de Pétri

3.1 Introduction

Les réseaux de Petri sont un formalisme qui permet de modéliser et d'analyser des systèmes concurrents et distribués. Ils sont particulièrement utiles pour représenter les comportements dynamiques de systèmes complexes, tels que les systèmes de production, les réseaux de communication, les systèmes de gestion de la chaîne d'approvisionnement, etc. Les réseaux de Petri sont également utilisés pour étudier les propriétés de ces systèmes, telles que la couverture, la synchronisation, la sécurité, etc. Les réseaux de Petri sont donc un outil puissant pour concevoir, analyser et optimiser les systèmes complexes, ce qui en fait une technique importante dans l'ingénierie.

L'étude d'un système à l'aide d'un réseau de Petri (RdP) se déroule selon trois étapes clés :

- Modélisation du système en termes de places et de transitions du RdP.
- Analyse du modèle obtenu et déduction des propriétés, telles que les propriétés de sûreté et de vivacité.
- Révision des propriétés pour validation du système obtenu. Cette étape est un essentielle pour obtenir une évaluation précise.

Ce chapitre met en lumière l'importance des RdP dans le contexte complexe actuel de l'industrie. Nous présentons tout d'abord, un historique et les notions de base des réseaux de Petri illustrées par des exemples explicatifs. Ensuite nous détaillons les propriétés mathématiques et génériques des RdP ; leurs analyses par graphe des marquages, par arbre de couverture, par graphe de couverture, par l'algèbre linéaire et par réduction. Ainsi leurs structures particulières ont été présentées, accompagnées de leurs avantages et inconvénients, suivi des différents types de RdP de haut niveau.

3.2 Historique

Les réseaux de Petri ont été introduits par Carl Adam Petri en 1962 (Petri, 1962), dans sa thèse de doctorat, intitulée *Kommunikation mit Automaten*, en Allemagne. Il a été créé la théorie des réseaux (General Net Theory). Dans cette thèse, Petri a présenté les fondements des réseaux de Petri, y compris leur définition, leur syntaxe et leur sémantique. Petri a été inspiré par les travaux de Konrad Zuse sur les machines à calculer pour développer les réseaux de Petri. Il a été intéressé par la façon dont les systèmes complexes peuvent être modélisés et analysés. Les réseaux de Petri offrent un moyen de représenter les interactions entre les différents composants d'un système. La thèse de Petri a été publiée en 1968 sous le titre *Communication with Automata*. Ce livre a contribué à populariser les réseaux de Petri auprès de la communauté scientifique, (Adams, 1969). Dans les années 70, les réseaux de Petri ont été étudiés par un groupe de chercheurs dirigé par (Holt et al., 1970). Ce groupe a développé de nouvelles techniques pour l'analyse des réseaux de Petri, notamment la méthode des places et transitions colorées. En 1975, la première conférence sur les réseaux de Petri et les méthodes relationnelles a été organisée au Massachusetts Institute of Technology (MIT).

Le reste de cette partie comprend une étude théorique des réseaux de Petri, leurs définitions, exemples, propriétés et extensions, (Ghenai, 2017).

3.3 État de l'art RdP et Surveillance Industrielle

(Lajmi et al., 2020) proposent une nouvelle approche pour la modélisation et l'analyse des défaillances. Ils combinent la représentation graphique fournie par les réseaux de Petri et la logique floue. Ils ont utilisé les réseaux de Petri dans l'analyse des défaillances permet de remplacer les fonctions de portes logiques dans les arbres de défaillances. La logique floue

Chapitre 3 : Théorie des Réseaux de Pétri

permet des descriptions en langage naturel des entités de processus ainsi ils appliquent ces règles pour diagnostiquer les pannes.

(Talebberrouane et al., 2019) ont identifié deux formalismes distincts pour analyser les scénarios de défaillance et la disponibilité des systèmes considérés. Ces réseaux de Petri stochastiques généralisés (GSPN) et les processus de Markov pilotés par arbre de défaillance (FTDMP).

(DONCHI et al., 2022) proposent une méthodologie permettant d'évaluer la probabilité de défaillance d'origine électrique d'une vanne d'évacuateur de crues dans le temps par l'utilisation de l'arbre de défaillance et l'implémentation des réseaux de Petri stochastiques. Les auteurs ont construit le réseau de Petri stochastique avec réparation modélisant les différents états de fonctionnement de la vanne.

(Blaga et al., 2021) proposent comment des modèles de réseaux de Petri ont été développés pour une grande variété de systèmes de fabrication. Le système modélisé était la cellule robotisée pour la préparation et l'analyse d'échantillons métallographiques. Le modèle très complexe avec de nombreux éléments. Ils ont fait la simplification du modèle qu'ils aient obtenu en passant à des réseaux de Petri colorés (RdPC).

(Djellal & Lakel, 2017) proposent un modèle de réseaux de Petri pour valider les techniques de recherche dans les graphes, pour améliorées la méthode Worst-Case, grâce à une optimisation dynamique du nombre d'agents impliqués.

(Chouchane & Declerck, 2022) proposent une approche algébrique pour étudier le diagnostic classes de défauts dans un système réseaux de Petri étiquetés (RdPE) basée sur l'estimation d'état sur une fenêtre glissante de longueur h et la détection de défauts a été effectuée par la résolution du problème programmation linéaire entière (PLE).

(M'halla, 2023) a développé une stratégie de surveillance à base de réseaux de Petri Hybrides, pour améliorer la qualité de produits d'une machine retordeuse, exactement la qualité des fils et de la forme géométrique de la bobine.

(Minca et al., 2008) ont proposé un nouvel outil de surveillance de modélisation des systèmes à événements discrets avec fonctionnement non autonome, intégrant la logique floue dans un outil de type RdP Temporisés. Les connaissances incertaines associées à cette activité exigent des raisonnements spécifiques et des méthodes de modélisation adaptées à une logique différente de la logique des connaissances précises. La logique floue fournit le cadre pour l'exploitation des connaissances floues, qui du point de vue qualitatif donne des résultats plus fins que la logique classique.

(Minca et al., 2003) ont proposé un nouvel outil appelé réseau de Petri flou pour la surveillance (RdPFS). Cet outil permet de modéliser le spectre des défauts et des dégradations par une approche temporelle floue. Il permet d'évaluer les défauts possibles par des radiographies temporelles dans le RdPFS, construit sur l'ensemble des règles logiques correspondant à l'arbre de défaillance du système surveillé. L'étude de cet article est basée sur une modélisation du système surveillé par un réseau de Petri temporel (RdPT). Les deux réseaux RdPT et RdPFS sont reliés par des mécanismes qui modélisent les symptômes des défauts. L'approche floue développée utilise des opérateurs flous particuliers.

Chapitre 3 : Théorie des Réseaux de Pétri

3.4 Réseaux de Petri

Les réseaux de Petri (RdP) sont des outils graphique et mathématique, qui permettent de décrire le comportement dynamique d'un système étudié. Ils se composent d'une structure et d'un marquage, (David & Alla, 2010), (Brauer et al., 1985). Ils permettent de décrire des relations existantes entre des conditions et des événements, de modéliser le comportement et l'évaluation des performances des systèmes dynamiques à événements discrets, en particulier les systèmes concurrents, parallèles, non déterministes, caractérisée par la concurrence, la synchronisation, l'exclusion mutuelle et le conflit des choix multiples. La facilité d'adaptation des réseaux de Petri leur permet de modéliser une grande variété de systèmes dans divers domaines, (Labadi, 2005), tels que :

- L'analyse de la sécurité pour modéliser des systèmes critiques, tels que les systèmes de contrôle de vol ou de production. L'analyse des propriétés de ces réseaux permet d'identifier les risques potentiels de sécurité et de mettre en place des mesures pour les prévenir.
- Analyse de la performance pour modéliser des systèmes dynamiques, tels que les systèmes de traitement des commandes ou de transport. L'analyse des propriétés de ces réseaux permet d'évaluer la performance du système et d'identifier les points d'amélioration possibles.
- Analyse de la fiabilité pour modéliser des systèmes complexes, tels que les systèmes électroniques, biologiques, les protocoles de communication, les chaînes de production et de fabrication, les systèmes informatiques et les architectures d'ordinateurs. L'analyse des propriétés de ces réseaux permet d'évaluer la fiabilité du système et de prédire les pannes potentielles.

Leur aspect mathématique permet d'analyser leur propriétés comportementales et structurelles, essentielles à la validation du système, (Bouarioua, 2013a), (David & Alla, 2010). Selon (David & Alla, 2010), les réseaux de Petri (RdP) permettent de décrire de manière précise, mais non formelle la structure d'un système. Offrant ainsi un support graphique de conception. Les RdP permettent également de décrire un système étape par étape, en décomposant en éléments plus simples les éléments constitutifs initiaux du système. Ils permettent enfin de décrire à l'aide d'un même support de base, à la fois la structure et la dynamique d'un système. Les RdP permettent de passer d'une description graphique d'un système à une description formelle permettant l'analyse mathématique du système (cohérence), il est possible de modéliser les Réseaux de Petri sous une forme matricielle, ce qui permet de développer de nombreuses techniques d'analyse. Cet outil permet l'analyse qualitative et quantitative des systèmes, (Petri, 1962). Leurs principaux utilisateurs se trouvent dans les domaines de l'informatique et de l'industrie. Les travaux sur la modélisation des RdP depuis 1989, sont détaillés dans les travaux suivant : (Valette, 2002), (Moncelet, 1998), (Ziegler, 1996), (Daniel, 1995), (Ladet, 1989), (Murata, 1989).

3.5 Notions de base des réseaux de Petri

Le réseau de Petri (RdP) est un graphe biparti orienté (figure 3.1), comprenant (David & Alla, 2010) :

- Un ensemble de places qui décrivent les états possibles du système ou des conditions. Ces places sont représentées graphiquement par des cercles, telle que P1 et P2 (figure 3.1).
- Un ensemble de transitions qui permettent de modéliser les événements ou les actions qui font basculer le système d'un état à un autre. Ces transitions sont représentées graphiquement par des rectangles, tels que T1 et T2 (figure 3.1).

Chapitre 3 : Théorie des Réseaux de Pétri

- Transition source (figure 3.1) : elle représente un événement initial ou une condition de démarrage qui lance le fonctionnement du réseau. C'est une transition qui n'a pas d'arcs entrants. Elle ne peut être déclenchée par aucun état précédent du système.
- Transition puits (figure 3.1) : elle est utilisée pour modéliser un état terminal ou une condition qui ne peut pas être quittée. C'est une transition qui ne possède pas d'arcs en aval, cela signifie qu'une fois qu'elle est franchie, elle ne peut plus être franchie à nouveau.
- Un ensemble d'arcs, sont représentées graphiquement par des flèches (figure 3.1) qui associent les places d'entrée aux transitions et les transitions aux places de sortie de poids entiers associés aux arcs. Un arc ne peut relier qu'une place à une transition ou une transition à une place, mais ne relie jamais une place directement à une autre place ou bien une transition à une transition.
- Des jetons sont représentés graphiquement par des marques qui sont des éléments ajoutés aux places pour simuler le déroulement du système.
- Un marquage (M), représenté graphiquement par des marques, des points ou des jetons, associe à chaque place un nombre fini positif ou nul. Le marquage est une fonction qui assigne un nombre de jetons à chaque place d'un réseau de Petri. Le marquage définit l'état du système à un instant donné.
- Le franchissement d'une transition dans un RdP consiste à retirer les jetons de chacune des places pré-conditionnelles de la transition et à ajouter les jetons à chacune des places post-conditionnelles de la transition. Le franchissement d'une transition franchissable n'est pas toujours immédiatement.

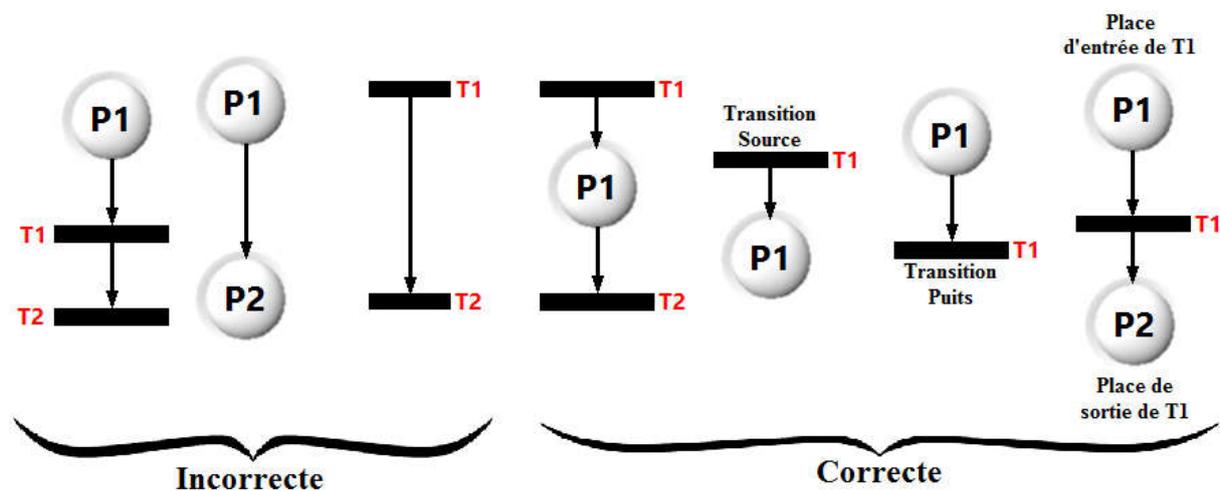


Figure 3. 1. Structure Correcte et incorrecte d'un RdP

En générale, les places d'un RdP sont repérées successivement par P1, P2, ..., Pn, et les transitions par T1, T2, ..., Tm. Le poids d'un arc est indiqué sur le modèle, à côté de l'arc. Si le poids n'est pas indiqué, il est implicitement pondéré à 1.

Par exemple, le modèle RdP (figure 3.2) peut représenter l'assemblage, et le désassemblage successif de deux éléments. Les places sont P1, P2, P3, P4 et P5 et les transitions T1, T2, T3 et T4, où :

- P1 est reliée à T1 par un arc orienté de P1 vers T1 : on dit que T1 est en sortie de P1. De même, on peut dire que P1 est en entrée ou en amont de T1. P1 est en sortie de T2.
- La place P2 est en entrée de T1. T4 est reliée à P2 par un arc orienté de T4 vers P2 : on dit que P2 est en sortie ou en aval de T4.
- P3 est en sortie de T1 et en entrée de T3.

Chapitre 3 : Théorie des Réseaux de Pétri

- P4 est en sortie de T3 et en entrée de T2 et T4 simultanément.
- P5 est en sortie de T4.

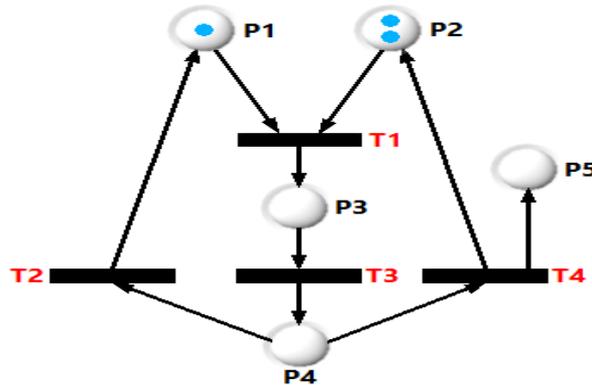


Figure 3. 2. Exemple RdP 1

Chaque place dans un RdP contient un nombre entier positif ou nul de jetons, définissant ainsi l'état du système à un instant donné. Le marquage est un vecteur colonne de dimension égale au nombre de places dans le réseau, où chaque élément du vecteur correspond au nombre de jetons contenus dans la place correspondante. L'ensemble des changements d'états possibles du système est déterminé par l'ensemble des transitions franchissables.

Par exemple à la figure 3.2, la place P1 contient un jeton et la place P2 contient deux jetons. Le marquage initial du réseau est donc $M_0 = [1 \ 2 \ 0 \ 0 \ 0]$. La transition T1 est franchissable, car les places P1 et P2 contiennent chacun au moins un jeton. Lorsque la transition T1 est franchie, un jeton est retiré de chaque place P1 et P2 simultanément et ajouté à la place P3. La transition T2 n'est pas franchissable, car la place P4 ne contient pas au moins un jeton. Le nouveau marquage du réseau est donc $M_1 = [0 \ 1 \ 1 \ 0 \ 0]$.

3.6 Propriétés Mathématiques

Les propriétés mathématiques contenues dans RdP sont selon (Djellal & Lakel, 2017), (Djellal, 2016), (David & Alla, 2010), (Diaz, 2009), (Petri, 1962) :

- 1) La désignation : un RdP est désigné par un quadruplet :

$$N = \{P, T, Pré, Post\} \tag{3. 1}$$

- P est un ensemble fini de places.
- T est un ensemble fini, et disjoint de P, de transitions tel que $P \cap T = \emptyset$ et $P \cup T = N$
- Pré est une application de $(P \times T)$ dans \mathbb{N} dite d'incidence avant. Tel que :

$$(\forall p \in P)T(p) = \{t \in T \mid \frac{T}{Pré(p, T)} > 0\} \tag{3. 2}$$

- Post est une application de $(P \times T)$ dans \mathbb{N} dite d'incidence arrière. Tel que :

$$(\forall t \in T)T(p) = \{p \in P \mid \frac{T}{Post(p, T)} > 0\} \tag{3. 3}$$

- Pré-place d'une transition (T) est une place (P) telle qu'il existe un arc dirigé de P vers T dans F. On note l'ensemble des pré-places de T par :

$$\bullet T \quad \{p \in P \mid (p, t) \in F\} \tag{3. 4}$$

Chapitre 3 : Théorie des Réseaux de Pétri

- Post-place d'une transition (T) est une place (P) telle qu'il existe un arc dirigé de T vers P dans F. On note l'ensemble des post-places de T par :

$$T \bullet = \{p \mid p \in P \wedge (t, p) \in F\} \quad (3.5)$$

- Pré-transitions d'une place (P) :

$$\bullet P = \{t \mid t \in T \wedge (t, p) \in F\} \quad (3.6)$$

- Post-transitions d'une place (P) :

$$P \bullet = \{t \mid t \in T \wedge (p, t) \in F\} \quad (3.7)$$

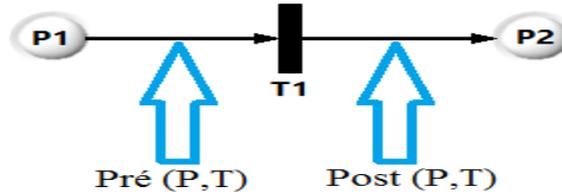


Figure 3. 3. RdP quadruplet $\{P, T, Pré, Post\}$

- 2) Le marquage : Un RdP marqué est un couple $\{R, M\}$ où R représente un RdP et est désigné par un quadruplet :

$$R = \{P, T, f, M_0\} \quad (3.8)$$

- L'ensemble des arcs orientés, pondérés par des valeurs entières non négatives, définit par :

$$F: (P \times T) \cup (T \times P) \rightarrow N_0 \quad (3.9)$$

- Le marquage initial (M_0), notation $M_i[T_j \geq M_k$. M est une application de P dans \mathbb{N} (\mathbb{N} : Ensemble des entiers naturels). $M(p)$ est égal au nombre de marques dans une place $p \in P$.

- 3) La matrice d'incidence : La matrice d'incidence (W) du réseau de Petri représente le coût global du franchissement d'une transition pour chaque place. Cette matrice associe à chaque ligne une place, et à chaque colonne une transition. Sachant que :

$$W = W^+ - W^- \text{ ou } W = Post - Pré \quad (3.10)$$

L'élément (i, j) de la matrice W donne le bilan, pour la place i, du franchissement de la transition j.

- Matrice d'incidence avant : Les arcs de liaison des Transitions vers les places $Pré (P_j, T_j)$ peuvent être représentés dans une matrice avec :

$$W^+(i, j) = Pré (P_i, T_j) \quad (3.11)$$

L'indice des lignes correspond aux places et l'indice des colonnes correspond aux transitions. Les valeurs de la matrice sont :

$$w^+(i, j) = \begin{cases} 1 & \text{s'il existe un arc de } T_j \text{ vers } P_i \\ 0 & \text{sin on} \end{cases}$$

- Matrice d'incidence arrière : Les arcs de liaison de places vers les transitions $Post (P_i, T_j)$ peuvent être représentés dans une matrice avec :

$$W^-(i, j) = Post (P_i, T_j) \quad (3.12)$$

Chapitre 3 : Théorie des Réseaux de Pétri

L'indice des lignes correspond aux place et l'indice des colonnes correspond aux transitions. Les valeurs de la matrice sont :

$$w^-(i, j) = \begin{cases} 1 & \text{s'il existe un arc de } P_i \text{ vers } T_j \\ 0 & \text{sin on} \end{cases}$$

- 4) Le vecteur de marquage du RdP est construit à partir du vecteur caractéristique \underline{S} , qui représente le nombre d'occurrences de chaque transition dans une séquence de franchissement. Soit S une séquence de transitions franchissables, l'état obtenu après avoir franchi les transitions de S est donné par la formule suivante :

$$M_K = M_0 + (W \times \underline{S}) \quad (3.13)$$

3.7 Propriétés Génériques

Les propriétés génériques contenues dans un RdP sont selon (Ghenai, 2017), (Bouarioua, 2013b) :

3.7.1 RdP borné

Pour définir la notion de Réseau de Petri borné, on peut se référer aux concepts suivants :

- Place bornée : Une place P_i est bornée pour un marquage initial (M_0), si pour tout marquage accessible à partir de M_0 , le nombre de jetons dans P_i est fini.
- RdP borné : Un RdP est borné pour un marquage initial (M_0), si toutes ses places sont bornées pour M_0 .
- Place K-bornée : Si pour tout marquage (M) appartenant à l'ensemble des marquages accessibles à partir de M_0 (noté $*M_0$), on a $M(P_i) \leq K$ où K est un nombre entier naturel, on dit que P_i est K-bornée.
- RdP K-borné : Si la propriété de K-bornéité est vraie pour toutes les places du RdP, on dit que le RdP est K-borné.

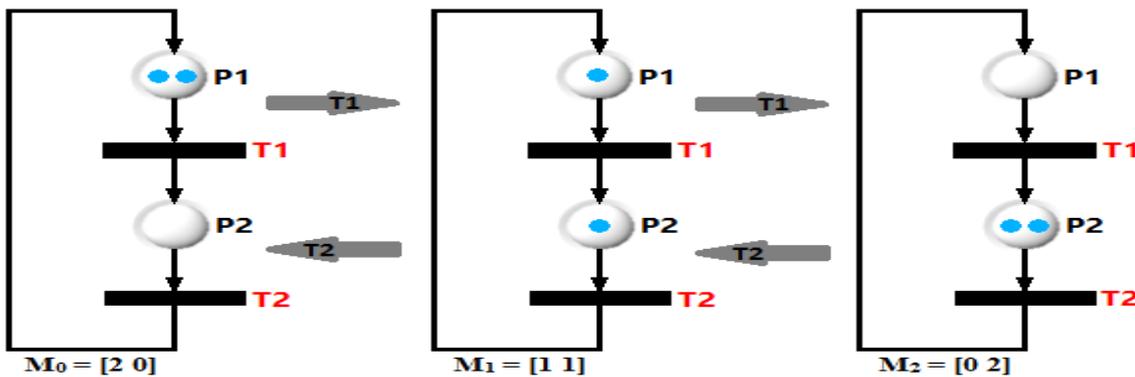


Figure 3. 4. Exemple RdP 2

La figure 3.4 illustre les propriétés génériques d'un RdP borné. Le nombre de marquages de ce réseau est borné et fini.

Chapitre 3 : Théorie des Réseaux de Pétri

3.7.2 RdP sauf

Un réseau de Petri sauf (ou binaire), est un RdP 1-borné. C'est-à-dire un RdP dans lequel chaque place peut contenir au plus un jeton.

3.7.3 Vivacité

- Transition vivante : Une transition T_j est vivante pour un marquage initial M_0 , si pour tout marquage accessible $M_i \in *M_0$, il existe une séquence de franchissement S qui contient T_j à partir de M_i .
- RdP vivant : Un RdP est vivant pour un marquage initial M_0 si toutes ses transitions sont vivantes pour M_0 .

A la figure 3.4 ci-dessus, toutes les transitions sont vivantes pour M_0 , le RdP est vivant pour M_0 .

3.7.4 Blocage

- Blocage (ou état puits) : un blocage est un marquage dans lequel aucune transition n'est validée.
- RdP sans blocage : Un RdP est considéré comme sans blocage pour un marquage initial M_0 , si pour tout marquage accessible $M_i \in *M_0$, il est sans blocage.

3.7.5 Réinitialisabilité

Un réseau de Petri est réinitialisable si et seulement si, pour tout marquage (M), il existe une séquence de transitions qui permet de revenir au marquage initial M_0 . Cette propriété est importante pour les systèmes interactifs qui doivent pouvoir être réinitialisés après chaque utilisation et elle renseigne sur le fonctionnement répétitif.

A la figure 3.4, le réseau est réinitialisable car il existe un marquage initial (M_0) à partir duquel tous les autres marquages peuvent être atteints.

3.7.6 Conflit

Un conflit dans un RdP se produit lorsque deux ou plusieurs transitions peuvent être déclenchées à partir du même marquage, mais que le système ne peut en choisir qu'une seule. Les types de conflits :

- Un conflit structurel : correspond au moins deux transitions : T_1 et T_2 sont en conflit structurel si elles partagent une place d'entrée commune. Ce conflit est noté comme suit :

$$K = \langle p, \{T_1, T_2, \dots, T_n\} \rangle \quad (3.14)$$

- Un conflit effectif : C'est l'existence d'un conflit structurel (k), et d'un marquage (M), tel que le nombre de jetons dans p est inférieur au nombre de transitions de sortie de p qui sont validés par M .

3.7.7 Accessibilité

- L'accessibilité dans un RdP se réfère à la possibilité d'atteindre un marquage final M à partir d'un marquage initial M_0 en suivant une séquence de transitions valides.
- Le graphe d'accessibilité d'un RdP est un outil puissant pour déterminer l'accessibilité des situations. Il représente tous les marquages accessibles à partir du marquage initial et les transitions qui les relient.

3.7.8 Sûreté

La sûreté d'un RdP réside dans sa capacité à fonctionner correctement et à prévenir les situations dangereuses. Les propriétés de sûreté qui peuvent être garanties pour un RdP incluent : absence de blocage, absence de conflit, atteinte des états desejables, évitement des états non desejables.

3.8 Analyse des réseaux de Petri

Les méthodes d'analyse des réseaux de Petri sont des outils puissants pour étudier les propriétés des systèmes modélisés.

3.8.1 Analyse par Graphes des Marquages

L'analyse par graphes des marquages est la méthode simple et puissante pour analyser les propriétés d'un RdP. Elle consiste à construire un graphe orienté où chaque sommet représente un marquage accessible et chaque arc symbolise le franchissement d'une transition permettant de passer d'un marquage à l'autre. Le graphe des marquages accessibles d'un RdP de \mathbb{N} à partir d'un marquage initial (M_0), noté $G(N, M_0)$, est un graphe orienté dont les sommets sont les marquages accessibles à partir de M_0 et les arcs représentent les transitions du RdP.

Propriétés du graphe des marquages :

- Sommets : Les sommets du graphe correspondent aux marquages atteignables à partir de M_0 en déclenchant une séquence de transitions.
- Arcs : Un arc relie deux sommets M_1 et M_2 si et seulement s'il existe une transition (T) dans le RdP telle que $M_1 [T > M_2$
- Propriétés : Le graphe des marquages permet déterminer plusieurs propriétés importantes du RdP, telles que la vivacité, l'accessibilité et la sûreté.

Deux situations se distinguent (Bouarioua, 2013a):

- Graphe des marquages fini : toutes les propriétés du réseau peuvent être déduites en l'analysant ce graphe. C'est la situation la plus favorable, car elle permet d'analyser le réseau dans son intégralité.
- Graphe des marquages infini : il est nécessaire de construire le graphe de couverture, qui est un sous-graphe contenant tous les marquages accessibles. Le graphe de couverture permet de déduire certaines propriétés du réseau, telles que sa bornitude et sa vivacité.

Le graphe des marquages du RdP correspondant à la figure 3.4 est :



3.8.2 Analyse par Arbre de Couverture et Graphe de Couverture

L'arbre de couverture et le graphe de couverture sont deux outils puissants pour l'analyse des réseaux de Petri (RdP). Ils permettent de (Guillou, 2010), (Karp & Miller, 1969) :

- Déterminer si un marquage accessible (M') couvre un marquage donné (M), tel que

$$\forall p \in P, M(p) \leq M'(p) \quad (3.15)$$

- Déterminer si un réseau est borné, c'est-à-dire si le nombre de jetons dans chaque place est toujours fini.

Chapitre 3 : Théorie des Réseaux de Pétri

- Déterminer si une place du réseau est bornée, c'est-à-dire si le nombre de jetons dans cette place est toujours fini.

Le principe de construction de l'arbre de couverture et du graphe de couverture dans les RdP consiste à explorer exhaustivement par chaînage avant les marquages accessibles à partir du marquage initial, la repérant les séquences répétitives croissantes selon un ordre défini sur l'ensemble des marquages.

Le graphe de couverture est une version finie du graphe des marquages, obtenue en fusionnant les marquages équivalents. Il permet de représenter un ensemble infini de marquages par un graphe fini, mais il y a une perte d'information en contrepartie.

Le graphe de couverture du RdP 3, figure 3.5 est :

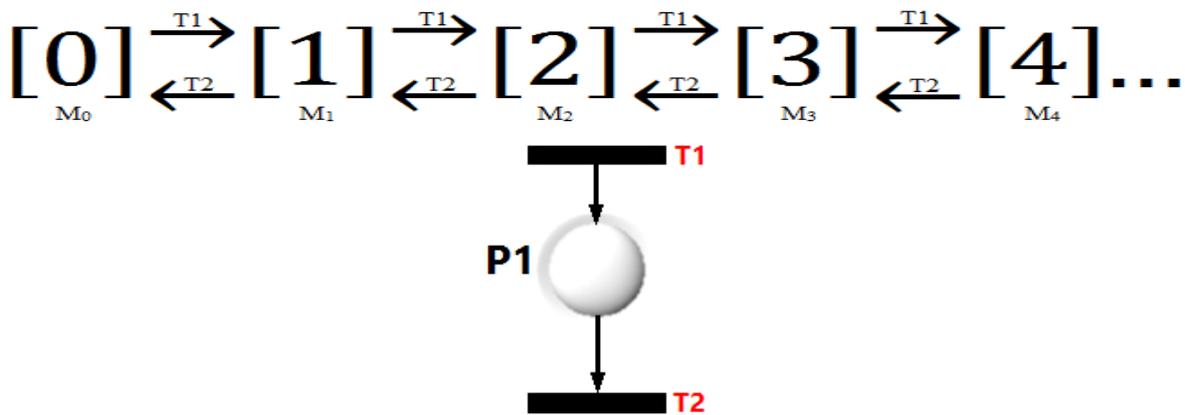


Figure 3. 5. Exemple RdP 3

3.8.3 Analyse par Algèbre Linéaire

L'analyse par algèbre linéaire permet d'étudier les propriétés structurelles d'un réseau de Petri, indépendamment du marquage initial. Elle consiste à représenter le RdP sous la forme d'une matrice d'incidence, qui décrit les relations entre les places et les transitions. En utilisant l'algèbre linéaire, il est possible de démontrer que certaines propriétés du réseau, telles que sa bornitude ou sa vivacité sont indépendantes du marquage initial. Par exemple, un réseau est structurellement borné s'il est borné pour tout marquage initial fini, et s'il est vivant pour tout marquage initial, alors le réseau est structurellement vivant, (F. Girault, 1997), (Toudic, 1982).

3.8.4 Analyse par Réduction

L'analyse par réduction de RdP marqués est une technique puissante pour simplifier l'analyse de systèmes complexes. En réduisant le nombre d'états et de transitions, elle permet d'obtenir un modèle plus facile à manipuler tout en conservant les propriétés importantes du système. Il est important de garder à l'esprit que le RdP simplifié n'est pas une représentation exacte du système réel, mais plutôt un outil d'analyse permettant de mieux comprendre son comportement, (Labadi, 2005).

3.9 Structures particulières des RdP

Selon (David & Alla, 1992b), il existe différents RdP qui présentent des structures particulières, c'est-à-dire qu'ils possèdent des caractéristiques et des propriétés qui ne sont pas présentes dans les réseaux dans le cas le plus général : graphe d'états, graphe d'événement, RdP sans conflit, RdP à choix libre, RdP simple, RdP pur, RdP sans boucle. RdP généralisé, RdP à capacité, RdP autonome, RdP non autonome.

3.9.1 Graphe d'États

Un RdP graphe d'états, est un RdP dans lequel les états représentent les situations possibles du système et les transitions représentent les actions possibles, si et seulement si toute transition a exactement une place d'entrée et une place de sortie, ce qui signifie qu'il n'y a pas de conflit entre les transitions. RdP est un graphe d'états si et seulement si le graphe de transitions est un graphe bidirectionnel. Les RdP graphe d'états sont utilisés pour modéliser des systèmes qui ont des états discrets et des transitions déterministes entre ces états, tels que les systèmes de contrôle de processus.

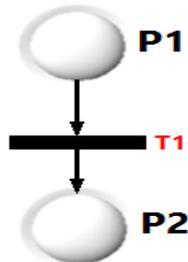


Figure 3. 6. Graphe d'états

3.9.2 Graphe d'Événement

Un RdP graphe d'événement, est un RdP dans lequel chaque place représente un événement et chaque transition représente un changement d'état possible. Si et seulement si chaque place a exactement une transition d'entrée et une transition de sortie, ce qui signifie qu'aucune transition ne peut être activée à partir d'un même état au même moment. Un graphe d'événement est donc dual d'un graphe d'états, et il est nécessairement sans conflit.

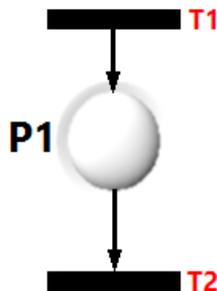


Figure 3. 7. Graphe d'événement

3.9.3 RdP sans Conflit

Un RdP sans conflit (K), est un RdP dans lequel aucune place n'a plus d'un jeton à la fois, si et seulement si toute place a au plus une transition de sortie. Il n'existe aucune transition qui peut être activée à partir d'un même état au même moment. Un conflit correspond à l'existence d'une place P1 qui a au moins deux transitions de sortie T1, T2, ... On notera $\langle P1, \{T1, T2, \dots\} \rangle$.

3.9.4 RdP Conflit Structurel

Un conflit structurel dans un réseau de Petri (RdP) survient lorsque deux ou plusieurs transitions peuvent se déclencher simultanément, mais que les ressources disponibles ne le permettent pas. En d'autres termes, il y a un conflit lorsqu'une place en amont de plusieurs transitions ne possède pas assez de jetons pour les activer toutes en même temps, $K = \langle P_i, \{T_1, T_2, \dots\} \rangle$. On distingue deux types de conflits structurels : Conflit simple (Deux transitions sont en conflit simple si

Chapitre 3 : Théorie des Réseaux de Pétri

elles partagent une seule place en amont.), Conflit multiple (Deux ou plusieurs transitions sont en conflit multiple si elles partagent plusieurs places en amont.).

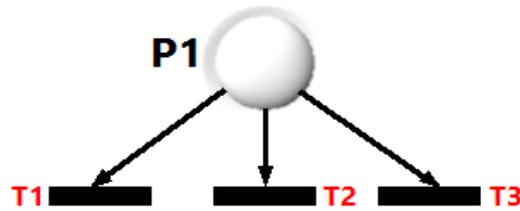


Figure 3. 8. Conflit structurel

3.9.5 RdP à Choix Libre

Un RdP à choix libre, est un RdP dans lequel, pour tout conflit, toutes les transitions de sortie de ce conflit ont la même place d'entrée. RdP est à choix libre si et seulement si chaque transition peut être franchie indépendamment des autres, et lorsqu'une transition est activée à partir d'un état, une seule transition peut être sélectionnée.

3.9.6 RdP Simple

Un RdP simple, est un RdP dans lequel chaque transition ne peut être impliquée que dans un conflit au plus. S'il existe une transition T1 est impliquée dans deux conflits, par exemple $\langle P1, \{T1, T2, \dots\} \rangle$ et $\langle P2, \{T1, T3, \dots\} \rangle$, alors le RdP n'est pas simple. RdP est simple si et seulement si chaque place a au plus une transition d'entrée et une transition de sortie. Il n'y a pas de boucles dans le graphe d'états.

3.9.7 RdP Pur

Un RdP pur, est un RdP dans lequel aucune transition n'a une place d'entrée qui est également une place de sortie de cette transition. RdP est pur si et seulement si chaque place a exactement une transition d'entrée et une transition de sortie, il n'y a pas de transitions sans condition.

3.9.8 RdP sans Boucle

Un RdP sans boucle, est un RdP qu'il ne contient pas de cycle, tel qu'il existe une transition Tj, et une place Pi qui est à la fois place d'entrée et place de sortie de Tj, donc Tj doit avoir au moins une autre place d'entrée. Dans un RdP sans boucle, chaque transition doit avoir au moins une autre place d'entrée. Cela garantit qu'il n'y a pas de boucle de transition directe entre deux places. Les RdP sans boucle sont utilisés pour modéliser des systèmes sans rétroaction directe entre les places, ce qui peut simplifier l'analyse du système.

3.9.9 RdP Généralisé

Un RdP généralisé, est un RdP dans lequel des poids sont pondérés par des nombres entiers strictement positifs et peuvent être étiquetés avec des expressions algébriques, sont associés aux arcs. Les transitions peuvent être activées simultanément à partir d'un même état au même moment. Les RdP généralisés peuvent être transformés en RdP ordinaires, mais des contraintes peuvent être placées au niveau des transitions pour les RdP généralisés.

3.9.10 RdP à Capacité

Un RdP à capacité, est un RdP dans lequel des capacités (nombres entiers strictement positifs) sont associées aux places. Le franchissement d'une transition d'entrée d'une place, dont la

Chapitre 3 : Théorie des Réseaux de Pétri

capacité est $\text{cap}(P_i)$, n'est possible que si le franchissement ne conduit pas à un nombre de jetons dans P_i qui dépasse cette capacité. Le nombre d'occurrences d'une transition est limité par la capacité des places concernées. Les RdP à capacité sont une extension des RdP classiques et peuvent être transformés en RdP ordinaires.

Dans l'exemple de la figure 3.9, le franchissement de T_1 conduit à 4 jetons. Après le franchissement de la transition T_1 , il y a 2 jetons dans la place P_2 , ce qui empêche T_1 d'être franchie à nouveau, parce que la capacité de la place P_2 est $\text{cap}(P_2) = 2$.

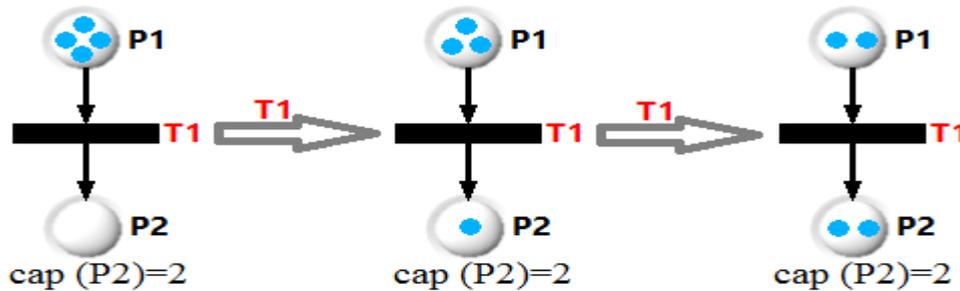


Figure 3. 9. RdP à capacité

3.9.11 RdP Autonome

Un RdP autonome, est un RdP décrit le fonctionnement d'un système qui évolue de façon autonome, c.-à-d. dont les instants de franchissement des transitions ne sont pas connus ou indiqués. Ce type de RdP peut fonctionner sans intervention externe ou du temps, et fonctionne selon des règles prédéfinies.

3.9.12 RdP non Autonome

Un RdP non autonome, est un RdP décrit le fonctionnement d'un système dont l'évolution est conditionnée par des événements externes ou par le temps. Il est nécessité une intervention externe pour fonctionner. Les RdP non autonomes peuvent être de différents types, tels que les RdP synchronisés, les RdP temporisés et les RdP P-temporels.

3.9.13 RdP à Priorité

Un RdP à priorité, est un RdP dans lequel les transitions ont des priorités. Les transitions avec la priorité la plus élevée sont activées en premier, suivies des transitions avec la priorité suivante, et ainsi de suite. Les RdP à priorité sont une extension des réseaux de Petri ordinaires et font partie des différentes variantes de réseaux de Petri, telles que les réseaux de Petri à arcs inhibiteurs, les réseaux de Petri non autonomes et les réseaux de Petri continus.

3.9.14 RdP à Arcs Inhibiteurs

Dans les réseaux de Petri classiques, le déclenchement d'une transition dépend uniquement de la présence de jetons dans les places d'entrée. Si une place ne contient aucun jeton, la transition associée ne peut pas se déclencher. Cette logique de base peut s'avérer insuffisante pour modéliser certains scénarios où le comportement du système dépend de l'absence de jetons dans une place. Les arcs inhibiteurs introduisent une nouvelle logique de déclenchement dans les réseaux de Petri. Ils permettent de conditionner le déclenchement d'une transition non pas seulement par la présence de jetons, mais également par l'absence de jetons dans une place spécifique. Graphiquement, les arcs inhibiteurs sont représentés par une flèche hachurée reliant

Chapitre 3 : Théorie des Réseaux de Pétri

une place à une transition. Ils sont représentés graphiquement dans l'exemple suivant (Figure 3.10) :

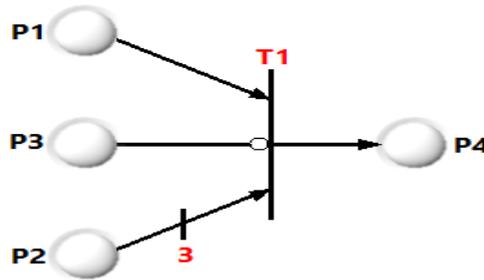


Figure 3. 10. Exemple de RdP avec arc inhibiteur

La figure 3.10, illustre un exemple de réseaux de Petri avec arc inhibiteur. Le déclenchement de la transition T1 requiert la présence d'un jeton dans la place P1, et de trois jetons dans la place P2. De plus, la place P3 doit être vide.

3.10 Avantages et inconvénients des réseaux de Petri généralisés et particuliers

3.10.1 Avantages

Le formalisme des réseaux de Petri présente plusieurs avantages :

- **Définition formelle** : Chaque modèle possède une sémantique bien définie, élimine ainsi toute ambiguïté dans la spécification.
- **Les réseaux de Petri sont exécutables** : Ce qui signifie qu'il existe des programmes construits sur la définition formelle de la notation qui permettent d'interpréter les modèles en RdP, et de simuler le fonctionnement du système en cours de spécification, offrant ainsi une vision dynamique du système.
- **Expression puissante** : Les RdP sont adaptés pour décrire des comportements complexes, réactifs ou concurrents.
- **Support de vérification** : Les RdP disposent de nombreuses techniques de vérification automatique, des propriétés génériques du système.
- **Représentation graphique** : Cette qualité facilite l'interprétation et la compréhension des modèles.
- **Flexibilité** : Les RdP peuvent être adaptés à une grande variété de besoins. Ils peuvent être utilisés pour modéliser des systèmes de différentes tailles et complexités.
- **Outils** : Il existe de nombreux outils disponibles pour la modélisation, la simulation et la vérification des réseaux de Petri. Cela facilite l'utilisation de cette technique.

3.10.2 Inconvénients

Les réseaux de Petri souffrent de certains inconvénients :

- **Taille de structuration limitée** : Les RdP peuvent devenir complexes et difficiles à gérer, en particulier pour les systèmes de grande taille. Cela peut rendre difficile la compréhension du modèle et la détection de problèmes.
- **Structure de données** : Les RdP places/transitions ne permet pas de décrire la structure des données manipulées par le système. Cela peut être un obstacle pour la modélisation de systèmes complexes.
- **Complexité de la vérification** : La vérification des propriétés des réseaux de Petri peut être complexe, en particulier pour les systèmes de grande taille.

Chapitre 3 : Théorie des Réseaux de Pétri

- **Performance** : La simulation des RdP peut être coûteuse en termes de temps et de ressources. Cela peut être un obstacle pour la modélisation de systèmes de grande taille.
- **Escalade** : La modélisation de systèmes complexes avec des RdP peut être difficile à réaliser et à maintenir.

Les inconvénients des RdP ont été étudiés et commencent à être résolus graduellement. Par exemple, des techniques de structuration des réseaux de Petri ont été développées pour faciliter leur gestion. De plus, des extensions des RdP (RdP de haut-niveau) ont été proposées pour permettre la description de la structure des données.

3.11 Réseaux de Petri de Haut Niveau

Selon (David & Alla, 1992b), il existe aussi différents types de réseaux de Petri de haut niveau: synchronisé, temporisés, stochastiques, colorés, interprétés, algébriques, flous, hiérarchiques, continus et hybrides.

3.11.1 Réseaux de Petri Synchronisés (RdPS)

Les réseaux de Petri synchronisés (RdPS) introduits par Diaz et Alla (Diaz & Alla, 1985), sont une extension des réseaux de Petri classiques qui permettent de modéliser des systèmes à événements synchrones. Dans un RdPS, chaque transition est associée à un événement spécifique. La transition est validée lorsque les conditions sur son marquage d'entrée sont satisfaites, mais elle ne devient franchissable que lorsque l'événement associé se produit. Cette distinction entre validation et franchissement permet de modéliser plus précisément les interactions synchrones entre les différentes composantes du système. Dans un RdP synchronisé, une transition validée n'est pas forcément franchissable immédiatement. Elle est validée lorsque les conditions sur les marquages sont satisfaites, mais elle devient franchissable uniquement lorsque l'événement externe associé à la transition se produit, entraînant son franchissement immédiat. Si plusieurs franchissements sont possibles en fonction des marquages des places d'entrée, seul celui associé à l'événement se produisant en premier se réalisera, (Berry et al., 1987). Les RdPS sont conçus pour être en phase avec l'évolution d'événements extérieurs, comme par exemple le démarrage d'un moteur sur le front montant (passage de 0 à 1) d'un bouton poussoir.

3.11.2 Réseaux de Petri Temporisés (RdPT)

Selon (Ghenai, 2017), (Medjoudj, 2006) et (Labadi, 2005), les réseaux de Petri temporisés (RdPT) introduisent la notion de temps dans les RdP classiques. Ils permettent de modéliser des systèmes à événements discrets dont le comportement dépend du temps (J. Wang, 1998). Il existe deux principales familles de RdPT : les réseaux de Petri Temporisés et les réseaux de Petri temporels.

3.11.2.1 Réseaux de Petri Temporisés

Les réseaux de Petri Temporisés ont été introduits par Ramchandani (Ramchandani, 1974). Les temporisations sont associées aux transitions (T-temporisés), puis aux places (P-temporisés). Ces temporisations représentent la durée minimale de franchissement ou le temps de séjour minimum d'un jeton dans une place. Les RdPs T-temporisés et P-temporisés sont équivalents.

3.11.2.1.1 Réseaux de Petri T-temporisés (RdPTt)

Un Réseau de Petri T-temporisé (RdPTt) est un couple $Ntl = \langle N, D \rangle$ où :

Chapitre 3 : Théorie des Réseaux de Pétri

- N est un réseau de Petri marqué $N = \langle P, T, M \rangle$.
- D est une fonction qui associe à chaque transition t_i du réseau une durée de temps dt_i .

Dans le triplet $R = \langle P, T, M \rangle$ où :

- P est l'ensemble des places du RdPTt.
- T est l'ensemble des transitions.
- M est le marquage.

Les réseaux de Petri T-temporisés modélisent la durée de temps d'attente associé à chaque transition. Le franchissement d'une transition t_i prend dt_i unités de temps, durant lesquelles les jetons consommés sont réservés et ne peuvent être utilisés par d'autres transitions.

3.11.2.1.2 Réseaux de Petri P-temporisés (RdPPT)

Un Réseau de Petri P-temporisé (RdPPT) est un couple $Ntl = \langle N, D \rangle$ où :

- N est un réseau de Petri marqué $N = \langle P, T, M \rangle$.
- D est une fonction qui associe à chaque place p_i du réseau une durée de temps dp_i .

Les réseaux de Petri P-temporisés modélisent la durée de temps d'attente associée à chaque place. Un jeton doit rester au moins dp_i unités de temps dans une place p_i avant d'être consommé par un franchissement de transition.

3.11.2.2 Réseaux de Petri Temporels

Les réseaux de Petri temporels ont été introduits par Merlin (Merlin, 1974), les temporisations sont associées aux transitions (T-temporel) ou aux places (P-temporel). Elles définissent un intervalle de temps pendant lequel la transition peut se franchiser (pour les transitions) ou pendant lequel un jeton peut séjourner dans une place (pour les places). Les RdPs T-temporels et P-temporels n'est pas équivalente.

3.11.2.2.1 Réseaux de Petri T-temporels

Un réseau de Petri T-temporel est un couple $Ntl = \langle N, D \rangle$ où :

- N est un réseau de Petri marqué $N = \langle P, T, M \rangle$.
- D est une fonction, associée à chaque transition t_i du réseau un intervalle statique de temps

$$d(t_i) = [d_{imin}(t_i), d_{imax}(t_i)] \quad (3. 16)$$

Les réseaux de Petri T-temporels modélisent le temps en associant des intervalles de temps à chaque transition. Chaque transition t_i doit rester sensibilisée pendant au moins $d_{imin}(t_i)$ unités de temps, et au plus $d_{imax}(t_i)$ unités de temps avant d'être franchie.

3.11.2.2.2 Réseaux de Petri P-temporels

Un réseau de Petri P-temporel est un couple $Ntl = \langle N, D \rangle$ où :

- N est un réseau de Petri marqué $N = \langle P, T, M \rangle$.
- D est une fonction qui associe à chaque place p_i un intervalle statique de temps

$$d(p_i) = [d_{imin}(p_i), d_{imax}(p_i)] \quad (3. 17)$$

Les réseaux de Petri P-temporels modélisent des intervalles de temps associés aux places. Chaque jeton doit rester dans la place p_i pendant au moins $d_{imin}(p_i)$ unités de temps, et au plus tard $d_{imax}(p_i)$ unités de temps de pouvoir être consommé par une transition. La durée pendant

laquelle un jeton reste dans la place p_i peut varier dans l'intervalle temporel $[d_{imin}(p_i), d_{imax}(p_i)]$.

Les réseaux de Petri temporels sont plus généraux que les réseaux de Petri temporisés. Ils permettent de décrire des mécanismes impossibles avec les réseaux de Petri temporisés comme les chiens de garde (watchdogs). Pour convertir un réseau de Petri temporisé en un réseau de Petri temporel, chaque transition t du réseau de Petri temporisé est remplacée par une séquence $[t_1, t_2]$ avec les caractéristiques suivantes :

$$d_{1min}(t_1) = d_{1max}(t_1) = 0 \quad (3.18)$$

$$d_{2min}(t_2) = d_{2max}(t_2) = dt_i \quad (3.19)$$

De manière similaire, la conversion d'un réseau de Petri p-temporisé en un réseau de Petri p-temporel suit le même principe.

3.11.2.2.3 Réseaux de Petri arc-pt-temporels

Un réseau de Petri arc-pt-temporel est un couple $Ntl = \langle N, D \rangle$ où :

- N est un réseau de Petri marqué $N = \langle P, T, M \rangle$.
- D est une fonction qui associe à chaque arc (p_i, t_i) reliant la place p_i à la transition t_i du réseau un intervalle statique de temps

$$d(p_i, t_i) = [d_{imin}(p_i, t_i), d_{imax}(p_i, t_i)] \quad (3.20)$$

Les réseaux de Petri arc-pt-temporel modélisent des intervalles de temps associé aux arcs. Chaque jeton doit rester dans l'arc reliant une place p_i à une transition t_i pendant au moins $d_{imin}(p_i, t_i)$ unités de temps après sa consommation de la place p_i , et il doit être consommé par une transition avant $d_{imax}(p_i, t_i)$. La durée pendant laquelle un jeton reste dans l'arc (p_i, t_i) peut varier l'intervalle temporel $[d_{imin}(p_i, t_i), d_{imax}(p_i, t_i)]$.

3.11.3 Réseaux de Petri Stochastiques (SPN)

Les réseaux de Petri stochastiques (SPN) introduits par Florin en 1978, sont obtenus à partir des réseaux de Petri classiques en attribuant des durées de franchissement aléatoires aux transitions, généralement distribuée selon une loi exponentielle. Cette distribution exponentielle permet d'exploiter les propriétés mathématiques d'un graphe de Markov, en prenant en compte l'occurrence des défaillances et leur impact sur le système. Les SPN permettent de modéliser le comportement des systèmes en décrivant de manière indépendante les états des différents composants du système et leurs interactions, tout en tenant compte du parallélisme. Ils sont utilisés pour évaluer la sûreté de fonctionnement des systèmes informatiques, en traitant des problèmes impliquant des phénomènes aléatoires. Dès le début des années 80, ce concept a connu un développement important (Chiola et al., 1993), (Florin, 1985), (Molloy, 1982), (Natkin, 1980), pour répondre aux exigences de la modélisation de systèmes de plus en plus complexes, comme les systèmes de production. Les notions de base et les principales propriétés des SPN sont présentées dans de nombreux ouvrages (Haas, 2006), (Diaz, 2001), (Marsan et al., 1998), (Lindemann, 1998).

Les réseaux de Petri stochastiques sont utilisés pour l'analyse des performances des systèmes de production, et différentes classes de SPN ont été proposées à cet effet. Ces classes se distinguent principalement par la nature des transitions utilisées. Initialement, les SPN associent à chaque transition un temps de franchissement aléatoire. Cependant, on retrouve également d'autres types de transitions.

3.11.3.1 Réseaux de Petri Stochastiques Généralisés (GSPN)

Cette classe de réseaux de Petri stochastiques (SPN) a été introduite par Ajmone Marsan (Marsan et al., 1998), (Marsan et al., 1984), se compose de transitions à temporisation nulle, dites transitions immédiates, et de transitions avec une temporisation aléatoire distribuée exponentiellement, dites transitions stochastiques. Elle se caractérise par la présence de deux types de transitions : Transitions immédiates : Ces transitions n'ont pas de temporisation et s'exécutent instantanément dès que leurs conditions d'activation sont remplies. Et transitions stochastiques : Ces transitions ont une temporisation aléatoire distribuée selon une loi exponentielle, s'exécutant après un délai aléatoire dont la loi de probabilité est connue. Le processus stochastique associé au graphe des marquages d'un GSPN est un processus de Markov à temps continu.

3.11.3.2 Réseaux de Petri Stochastiques et Déterministes (DSPN)

Cette classe de DSPN, également introduite par Ajmone Marsan (Marsan & Chiola, 1987), (Marsan & Chiola, 1986), étend les GSPN en incluant un troisième type de transition : des transitions immédiates, des transitions à temporisations déterministes et des transitions à temporisations stochastiques distribuées selon des lois exponentielles. Les DSPN ont été étudiés et analysés par de nombreux auteurs, notamment (Lindemann, 1998), (Ciardo & Lindemann, 1993), (Choi et al., 1993a), (Choi et al., 1993b), (Haas & Shedler, 1989a), (Haas & Shedler, 1989b).

3.11.3.3 Réseaux de Petri Stochastiques Étendus (ESPN)

Les réseaux ESPN se composent uniquement de transitions temporisées aléatoires, mais la loi de probabilité associée peut être quelconque, et non nécessairement exponentielle comme dans les SPN classiques. Le processus stochastique sous-jacent au graphe des marquages d'un ESPN est avec quelques restrictions, un processus semi-markovien. Si ces restrictions ne sont pas remplies, la résolution du réseau ESPN nécessite alors l'utilisation de technique de simulation à événements discrets, (Dugan, 1984).

3.11.4 Réseaux de Petri Colorés (RCP)

Les réseaux de Petri colorés (RCP) étendent les réseaux de Petri classiques en associant des couleurs ou des identificateurs aux jetons et des ensembles de couleurs aux transitions. De plus, des fonctions sont associées aux arcs pour définir les transformations appliquées au marquage, c'est-à-dire aux identificateurs, lorsqu'une transition est franchie, (Ghenai, 2017), (Moraru, 2005). Les RCP permettant de distinguer les jetons les uns des autres en leur attribuant des informations spécifiques, (S. Haddad, 1987), (Jensen, 1986). Cette caractéristique ne confère pas de capacité supplémentaire en termes de description par rapport aux réseaux de Petri classiques, mais permet une meilleure organisation de l'information. Chaque réseau de Petri coloré marqué correspond à un réseau de Petri isomorphe. Des extensions stochastiques de ce concept ont été développées par nombreux auteurs, telles que les RdPs Stochastiques Colorés (Chincholkar & Krishnaiah Chetty, 1996) et les RdPs stochastiques généralisés colorés (Dutheillet & Haddad, 1989). Ces réseaux de Petri colorés sont principalement utilisés comme des outils de simulation.

3.11.5 Réseaux de Petri Interprétés (RdPI)

Les réseaux de Petri interprétés (RdPI) ont été introduits par Petri et Reising (Petri & Reising, 1985), en tant qu'extension des réseaux de Petri classiques afin d'améliorer leurs capacités de

Chapitre 3 : Théorie des Réseaux de Pétri

modélisation, en particulier pour la représentation et l'analyse des systèmes concurrents. Les RdPI introduisent une fonction d'interprétation qui attribue des significations aux éléments du réseau, permettant une représentation plus précise et formelle du comportement du système.

Dans les réseaux de Petri colorés, les données du problème étudié sont modélisées à la fois par les places et par les domaines de couleurs. Cette approche permet de modéliser le contrôle et les données avec un formalisme unique, favorisant une analyse plus riche. Cependant, elle peut conduire à la construction de modèles complexes. Dans un réseau interprété, on sépare les données du contrôle. Le contrôle est modélisé par un RdP classique, tandis que les données sont représentées par un environnement, constitué d'ensembles finis de variables, associé à l'état du réseau. Chaque transition est associée à un prédicat portant sur l'environnement et à une action qui peut modifier cet environnement, (Reisig, 1991), (C. Girault & Pradat-Peyre, 2001).

3.11.6 Réseaux de Petri Algébriques (APN)

Les réseaux de Petri algébriques (APN) ont été introduits par De Micheli et Sangiovanni-Vincentelli (De Micheli & Sangiovanni-Vincentelli, 1981). Il s'agit d'un cadre de modélisation formelle permettant de représenter et d'analyser des systèmes concurrents. Les APN étendent le formalisme des réseaux de Petri classiques incluant les places, transitions, pré et post conditions, en introduisant une structure algébrique pour définir les données transportées par les jetons et les opérations qui les manipulent. Cette spécification algébrique donne une interprétation aux jetons et à la dynamique du modèle, en précisant quels jetons consommés ou produits lors du franchissement des transitions. Cette amélioration permet une représentation plus expressive et rigoureuse des systèmes complexes, en particulier ceux impliquant le flux de données et la synchronisation, (Reisig, 1991), (C. Girault & Pradat-Peyre, 2001).

Une spécification algébrique repose sur les notions de signature, terme, axiome et affectation. La signature constitue la base de la spécification algébrique. Elle définit les éléments de base du langage algébrique utilisé pour décrire les données et les opérations dans l'APN. Elle comprend deux composantes principales : (ensemble de types S et ensemble d'opérations OP. Les termes représentent des expressions algébriques construites à partir des types et des opérations définis dans la signature. Ils permettent de décrire des valeurs complexes en combinant des opérations élémentaires, comprend (Termes de base et Termes construits). Les axiomes servent à définir les propriétés des opérations et à garantir la cohérence du langage algébrique. Ils sont exprimés sous forme d'équations entre termes et permettent de spécifier des relations entre les différentes opérations. Par exemple, un axiome pourrait définir la commutativité de l'addition : $a + b = b + a$ pour tous les termes "a" et "b" du type "entier". Les affectations associent des expressions algébriques (termes) aux arcs du réseau de Petri. Elles définissent les valeurs des jetons qui sont produits ou consommés lors du franchissement des transitions. Les expressions algébriques peuvent utiliser les variables des places adjacentes à la transition pour représenter les valeurs des jetons consommés, (Reisig, 1991), (C. Girault & Pradat-Peyre, 2001).

3.11.7 Réseaux de Petri Flous (RdPF)

Les réseaux de Petri flous (RdPF) introduits par Cardoso en 1996, sont obtenus à partir des réseaux de Petri classiques qui permet de modéliser des systèmes avec des connaissances incertaines ou floues, (Ghenai, 2017). Deux catégories principales de RdPF existent :

Chapitre 3 : Théorie des Réseaux de Pétri

3.11.7.1 RdPF basés sur des règles floues

Cette catégorie modélise des systèmes à événements discrets en utilisant des règles floues pour décrire leur comportement. Les règles floues développées par Valette (Valette & Atabakhche, 1987) et (F. Girault et al., 1995), sont mises en correspondance avec les transitions du réseau. L'ensemble des règles représente un système expert flou, où les transitions correspondant aux règles simultanément vraies seront franchies en parallèle. Ce type de RdPF est similaire à un système expert flou simplifié.

3.11.7.2 RdPF basés sur la logique linéaire floue

Cette catégorie modélise des systèmes physiques dynamiques avec des connaissances incertaines sur les états des ressources du système. La logique linéaire floue est appliquée aux transformations de l'état des ressources, tout en conservant l'incertitude sur leur présence ou leur disponibilité. Les composantes du RdPF sont adaptées pour représenter les connaissances incertaines par des variables floues associées aux places, transitions, jetons, marquage ou arcs.

Dans toutes les approches RdPF, la présence du jeton reste binaire, même s'il peut porter des valeurs floues lorsqu'il se trouve dans une place, (Sallak, 2007), (Ghenai, 2017). Les RdPF offrent un cadre flexible pour modéliser des systèmes complexes avec des connaissances incertaines ou floues. De nombreuses applications des RdPF existent dans divers domaines, tels que le contrôle flou, la robotique, la gestion de la production et l'analyse de systèmes, (Sallak, 2007).

3.11.8 Réseaux de Petri Hiérarchiques (HPN)

Les réseaux de Petri hiérarchiques (HPN) offrent un formalisme puissant pour modéliser des systèmes complexes de manière modulaire et hiérarchique, s'inspirant des langages de programmation. Un HPN se compose de plusieurs modules, chacun représentant un sous-réseau de Petri. Un module principal référence tous les autres modules secondaires. Un module principal, appelé module racine, référence tous les autres modules secondaires.

Le référencement d'un module s'effectue via une transition de substitution portant le nom du module en question. L'utilisation des HPN est préconisée pour la modélisation des problèmes de grande taille, car cette approche modulaire facilite la gestion de la complexité, (Ghenai, 2017), (Moraru, 2005).

3.11.9 Réseaux de Petri Continus (RdPC)

Les réseaux de Petri continus (RdPC) ont été introduits par David et Alla (David & Alla, 1990) à partir des réseaux de Petri discrets. Un RdPC est constitué de places continues et transitions continues. Son fonctionnement repose sur la notion de vitesses associées aux transitions, qui peuvent être constantes ou variables.

Le nombre de jetons dans un RdPC est un réel positif. Le franchissement des transitions s'effectue de manière continue, en introduisant la notion de vitesse de transition, qui représente le nombre de marques franchies par unité de temps, (Bouarioua, 2013a).

Dans le cas des systèmes où le nombre de ressources est important et/ou que les durées opératoires varient dans le temps, une approximation continue des phénomènes et des variables devient pertinente. Une modélisation continue présente dans les RdPC, considère les transferts d'entités comme des flux, remplaçant ainsi les variables discrètes par des variables continues. Cette classe de réseaux de Petri a conduit à une autre classe dite les réseaux de Petri Hybrides, (Labadi, 2005).

3.11.10 Réseaux de Petri Hybrides (RdPH)

Les réseaux de Petri hybrides (RdPH) introduits par David et Alla en 1989, offrent un cadre formel puissant pour représenter et analyser des systèmes dynamiques qui présentent à la fois des aspects discrets et continus. Ils permettent de modéliser de manière unifiée la dynamique discrète et la dynamique continue en utilisant des places et des transitions spécifiques, (Zaytoon, 2001), (Quenec'hdu & Zaytoon, 2001). La dynamique discrète dans les RdPH est modélisée par des places et des transitions ordinaires, tandis que la dynamique continue est représentée par des places dites continues, où le marquage est un nombre réel positif ou nul, et des transitions continues qui symbolisent des écoulements continus. Le franchissement d'une transition continue t_i est associé à une vitesse $V(t)$, qui détermine la quantité de jetons transférés par unité de temps. Entre deux instants t et $t + dt$, une quantité $V(t) dt$ de jetons est retirée de sa place amont et la transition et ajoutée à la place aval. Cette approche permet de modéliser des variables qui évoluent de manière continue et linéaire en fonction du temps, (Medjoudj, 2006).

Les réseaux de Petri Lots (Demongodin, 2001) (Demongodin, 1994), étendent les RdPH (David & Alla, 2001), en introduisant un nouveau type de nœud appelé "nœud lot". Ces nœuds représentent des éléments de transfert avec possibilité d'accumulation d'entités, tels que les convoyeurs dans les systèmes de production à haute cadence. Cette extension permet de modéliser plus précisément le comportement de ces systèmes, en tenant compte de la dynamique des lots d'entités, (Labadi, 2005).

3.12 Conclusion

Les réseaux de Petri offrent un cadre puissant et flexible pour la modélisation et l'analyse des systèmes complexes. Ils sont largement utilisés pour leur formalisme mathématique rigoureux, leur représentation graphique intuitive et leur large éventail d'outils d'analyse.

Dans ce chapitre, nous avons passé en revue l'historique des RdP, leurs notions de base, ainsi que leurs propriétés mathématiques et génériques, leurs structures générales et particulières, leurs avantages et inconvénients. Enfin, les différents types de réseaux de Petri de haut niveau ont été décrits.

Chapitre 4 : Applications – Usinage robotisé en parachèvement

4.1	Introduction	60
4.2	Problématique	60
4.3	Transfert AdF - RdP	62
4.4	Étude de cas 1 - RdP	63
4.4.1	Analyse par Arbre des Fautes	63
4.4.2	Implémentation RdP - LabView	64
4.4.3	Simulations et résultats	66
4.5	Étude de cas 2 - RdP λ	67
4.5.1	Formalisation RdP λ	67
4.5.2	Paramétrage de la cellule d'usinage de parachèvement.....	67
4.5.3	Analyse par AdF	69
4.5.3.1	Analyse par RdP λ	70
4.5.4	Implémentation LabView	73
4.5.4.1	Implémentation AdF – LabView	73
4.5.4.2	Implémentation RdP λ – LabView	74
4.5.5	Simulations et résultats	76
4.5.5.1	Taux de défaillance	76
4.5.5.2	Simulation LabView	76
4.5.5.3	Marquage	78
4.5.5.4	Propriétés génériques du RdP λ	79
4.5.5.5	Matrice d'incidence Avant du RdP λ	79
4.5.5.6	Matrice d'incidence Arrière du RdP λ	79
4.5.5.7	Matrice d'incidence du RdP λ	79
4.5.5.8	Vecteur de marquage du RdP λ	80
4.6	Conclusion	81

4.1 Introduction

Les multiples reconfigurations et la complexité des systèmes de production modernes, telles les cellules robotisées d'usinage, conduisent à concevoir des systèmes d'aide à la surveillance de plus en plus performants. Dans le cadre de la gestion d'une cellule robotisée d'usinage en parachèvement, nous nous sommes intéressés à la modélisation des changements dans la dynamique du système lorsqu'une ou plusieurs défaillances surviennent.

Nous avons commencé notre travail par un examen approfondi des composants de la surveillance (figure 4.1). Plus précisément, nous nous sommes concentrés sur le système de diagnostic avec des entrées externes qualitatives (Arbre des Fautes), sur le système de diagnostic de la cellule robotisée d'usinage. En sortie, nous obtenons les différentes causes possibles associées à une localisation des fautes et un degré de crédibilité et un degré de criticité et gravité pour chacune d'elles. Ces degrés aideront les responsables à évaluer et à planifier les actions de maintenance. Dans la classification globale des méthodes et modèles de surveillance, présentée à la section 2.4, nous nous sommes concentrés sur les méthodes de surveillance par modélisation fonctionnelle et matérielle (AdF et RdP).

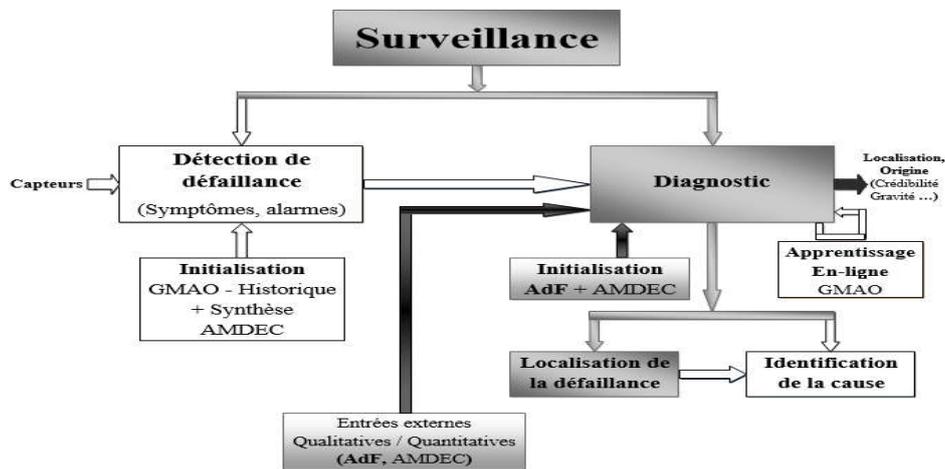


Figure 4. 1. Schéma fonctionnel du système de surveillance

4.2 Problématique

Deux études de cas ont été réalisées sur la cellule robotisée d'usinage avancée du Laboratoire Génie de Production (LGP) de l'École Nationale d'Ingénieurs de Tarbes (ENIT). La première étude, (Mehar et al., 2022), est basée sur une analyse qualitative pour le parachèvement de pièces parallélépipédiques, figure 4.2. Tandis que la seconde étude, (Mehar et al., 2023) est basée sur une analyse qualitative et quantitative pour le parachèvement de pièces cylindriques, figure 4.3.



Figure 4. 2. Étude de Cas 1 : Cellule robotisée d'usinage de parachèvement de pièces parallélépipédiques

i) Robot KUKA KR120 ; ii) Meuleuse pneumatique ; (iii) Outil (fraise) adapté à la tâche à réaliser ; (iv) Pièce (parallélépipédique) bridée dans un étau disposé sur une table rainurée ; (v) Caméra du système de vision (logiciels de traitement d'images de National Instruments).



Figure 4. 3. Étude de Cas 2 : Cellule robotisée d'usinage de parachèvement de pièces cylindriques

i) Robot KUKA KR120 ; ii) Électro-broche ; (iii) Outil (conique) adapté à la tâche à réaliser ; (iv) Pièce (cylindrique) bridée dans un étau disposé sur une table rainurée ; (v) Système de lubrification (pompe).

Chapitre 4 : Application – Usinage robotisé en parachèvement

Les défaillances sont supposées être permanentes, c'est-à-dire que lorsqu'une défaillance se produit, le système passe en mode dégradé et ne peut plus revenir à un mode nominal sans réparation. Il peut ainsi finir dans un mode de défaillance, dans lequel le système n'est plus opérationnel. Dans le cas de défaillances, la minimisation du temps de remise en route est primordiale afin de ne pas pénaliser la productivité de la cellule concernée.

C'est dans ce contexte que nous proposons un système de diagnostic basé sur l'utilisation d'un réseau de Petri (RdP), étude de cas 1, implémenté à partir de l'arbre des fautes correspondant. L'étude de cas 2 est une extension du RdP à un RdP λ (Réseau de Petri Lambda), implémenté à partir de l'arbre des fautes (AdF) correspondant.

Pour cela, notre méthodologie a été établie, dans les deux cas d'étude, suivant trois étapes : Dans la première étape l'analyse par AdF de la cellule robotisée d'usinage a été réalisée. Dans la seconde étape, le transfert des portes logiques de l'AdF vers le RdP a été effectué, s'en est suivie la modélisation du système en RdP. La troisième étape a consisté à piloter le RdP dans l'environnement LabView. En introduisant les taux de défaillance respectifs pour le cas 2 (pilotage du RdP vers le RdP λ), selon une expertise hiérarchique de robustesse.

Dans ce dernier cas, les données de défaillance de chaque composant ont ensuite été utilisées pour calculer le RdP λ du système. À ce stade, des propriétés génériques normatives de vérification ont été appliquées. Ainsi, des propriétés mathématiques incluent notamment le marquage, les matrices d'incidences et le vecteur de marquage. Ces analyses qualitatives et quantitatives du système ont été primordiales pour la validation du système.

4.3 Transfert AdF - RdP

Selon (W. S. Lee et al., 1985) et (Hura & Atwood, 1989), l'arbre de fautes (AdF) en général, se divise en deux catégories : cohérent et incohérent. Un AdF cohérent est constitué uniquement d'opérations logiques "ET" et "OU", tandis qu'un arbre non cohérent contient également d'autres opérations logiques. Dans notre étude, nous nous sommes intéressés uniquement à l'application de la modélisation des réseaux de Petri à l'analyse cohérente des arbres de fautes. L'utilisation des connaissances AdF avec le principe RdP, à travers les raisonnements, doit se faire de façon qualitative (étude de cas 1), mais aussi quantitative (étude de cas 2) pour fournir une analyse efficace.

Il existe une relation logique entre les réseaux de Petri et les arbres de fautes. Le formalisme utilisé dans notre travail permet d'implémenter le RdP à partir de l'AdF. Nous utilisons donc un formalisme qui associe chaque porte de l'AdF à une symbolisation du RdP.

1) La porte logique « ET » se déclinera de la façon suivante, figure 4.4 :

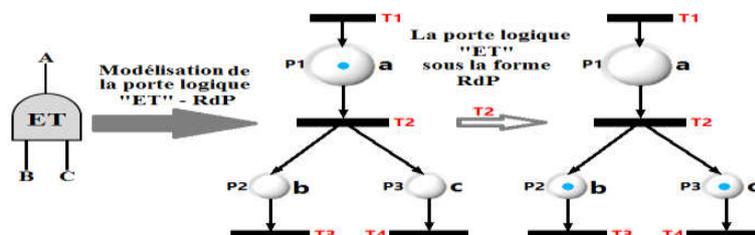


Figure 4. 4. Transformation de la porte logique « ET » de l'AdF en RdP

2) La porte logique « OU » se déclinera de la façon suivante, figure 4.5 :

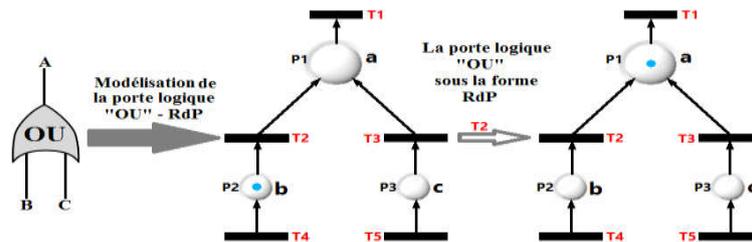


Figure 4. 5. Transformation de la porte logique « OU » de l'AdF en RdP

4.4 Étude de cas 1 - RdP

4.4.1 Analyse par Arbre des Fautes

Le système de diagnostic que nous développons s'appuie sur la base d'apprentissage initiée à partir de l'Arbre des Fautes (AdF) de la cellule robotisée considérée. Cet AdF sera la pièce centrale de notre stratégie basée sur les RdP et est présenté en figure 4.6. L'analyse de l'événement redouté dans notre AdF met en évidence la non-conformité de la pièce usinée (a) dans la cellule robotisée.

Nous avons utilisé le logiciel CABTREE pour construire et traiter notre arbre de fautes. Nous avons limité notre étude à deux niveaux qui font apparaître les premiers événements élémentaires, comme il est présenté en figure 4.6 :

- Premier niveau : Robot KUKA KR120 défaillant, (b) ; Meuleuse défaillante, (c) ; Outil défaillant, (d) ; Pièce défaillante, (e) ; Contrôle système de vision défaillant (Caméra), (f).
- Deuxième niveau : Armoire de commande de KUKA KR 120, (b1) ; Système Mécanique Articulé, (b2) ; Couple erroné, (c1) ; Défaillance totale, (c2) ; Bris sur l'outil, (d1) ; Mauvais affutage, (d2) ; Mauvais positionnement de la pièce, (e1) ; Caractéristiques non conformes de la pièce brute, (e2) ; Vision floue (Faux résultats), (f1) ; Mauvais réglage caméra - traitement erroné, (f2).

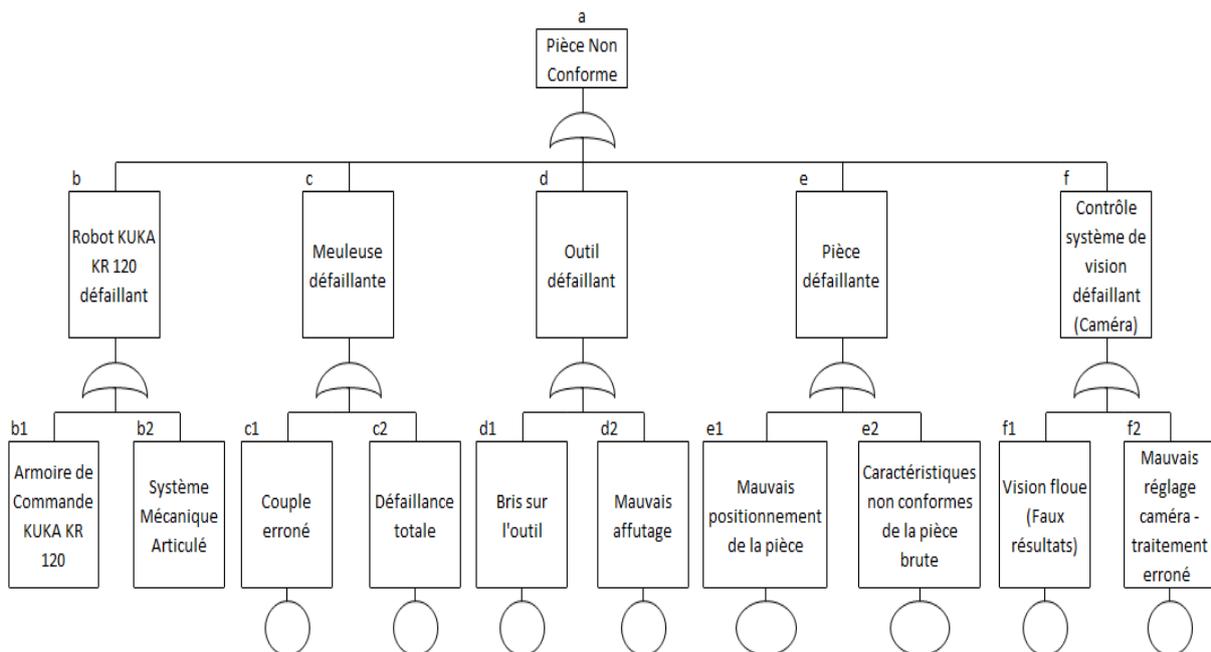


Figure 4. 6. Arbre des Fautes de la cellule robotisée

4.4.2 Implémentation RdP - LabView

Nous avons appliqué notre technique de transformation de l'AdF en un RdP, nous avons utilisé le logiciel LabView comme plateforme pour la modélisation et la simulation. La structure obtenue est illustrée aux figures 4.7 et 4.10. Ces figures représentent la face-avant, à savoir l'interface utilisateur des instruments virtuels de notre RdP qui contient 16 places qui sont des LEDs (Diode Électroluminescente circulaire) et 26 transitions. Chaque place représente un évènement de notre AdF et contient aussi leur état : des entrées (les commandes) et des sorties (les indicateurs) du programme. Les commandes et indicateurs sont matérialisés par des afficheurs numériques pour 0 ou 1.

Un graphe numérique affiche les données en tant qu'impulsions, agissant comme un générateur fait référence à un outil ou un processus qui produit une suite de valeurs binaires (0 ou 1). Ces valeurs binaires sont ensuite utilisées pour simuler le test de marquage et représenter l'état de chaque "place" dans un réseau de Petri. Si une place contient une marque, le signal affiche l'état 1 ; sinon, il affiche l'état 0. L'axe des abscisses (X) représente l'échelle des États (places), tandis que l'axe des ordonnées (Y) correspond à l'échelle des Marquages.

Notre RdP est borné à 1, vivant, ni réinitialisable, ni répétitif et graphe des marquages infini. Le marquage initial, correspondant à la figure 4.7, est $M_0 = [0000000000000000]$.

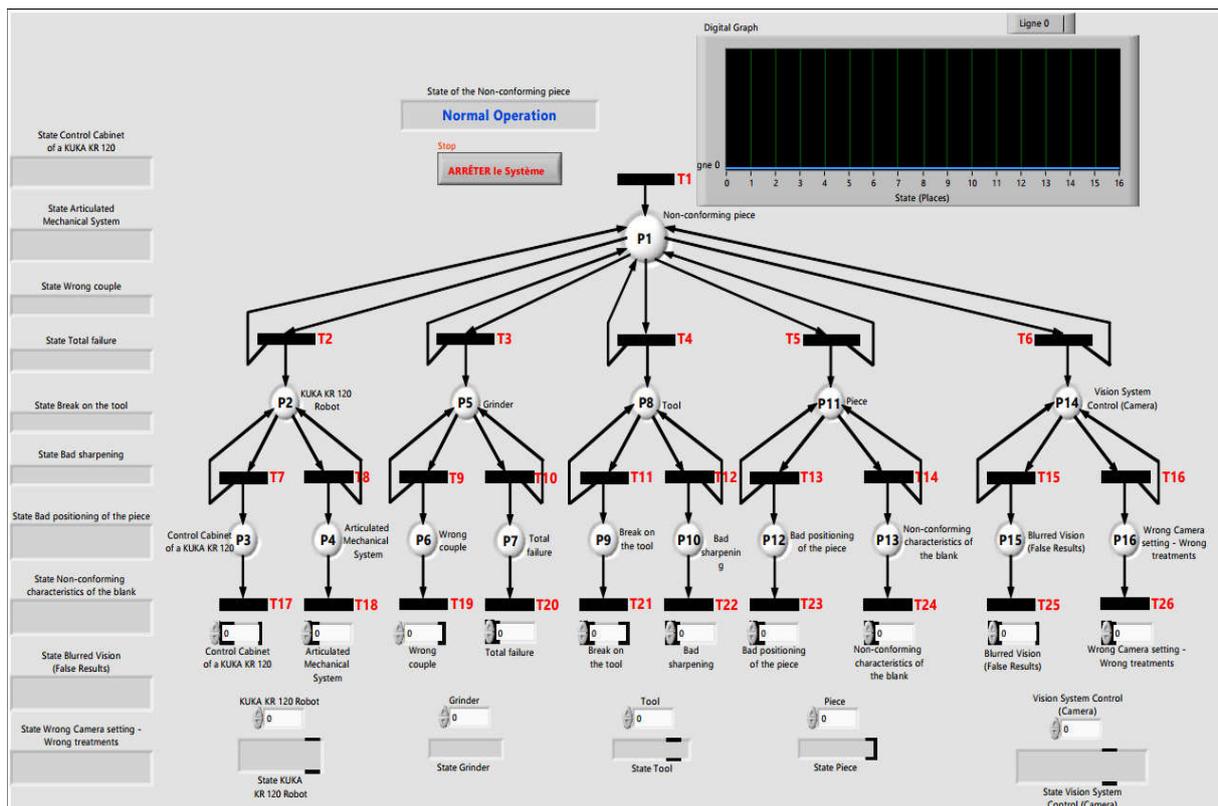


Figure 4. 7. Block Face avant du LabView du système RdP- Modélisation sans faute

Le block diagramme, figure 4.8, représente une partie du programme de l'application développé sous la forme d'un diagramme de flux de données.

Chapitre 4 : Application – Usinage robotisé en parachèvement

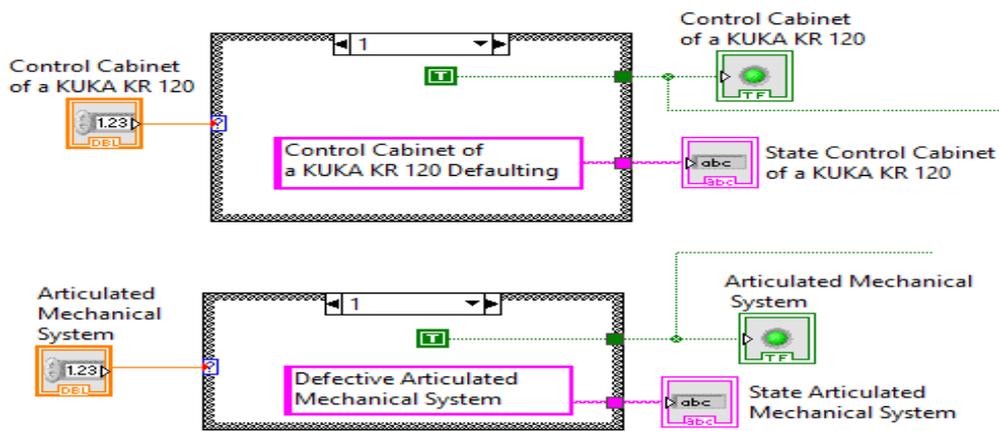


Figure 4. 8. Block diagramme du LabView - une partie du système RdP

La figure 4.9 illustre comment matérialiser la commande et l'indicateur par un afficheur numérique pour l'état 0 ou 1, dans le block diagramme du LabView.

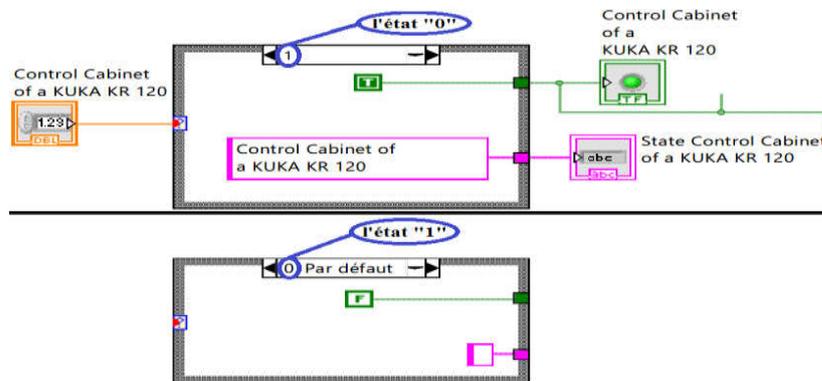


Figure 4. 9. Block diagramme du LabView - exemple d'une commande

Les transitions (T1, T2,, T26) de notre système réseau de pétéri sont matérialisées par des capteurs. Le tableau 4.1 décrit la signification de chaque place.

Tableau 4. 1. Signification des places

Les places	La signification de chaque place
P1	Pièce non Conforme
P2	Robot KUKA KR 120
P3	Armoire de Commande Armoire de KUKA KR 120
P4	Système Mécanique Articulé
P5	Meuleuse
P6	Couple erroné
P7	Défaillance totale
P8	Outil
P9	Bris sur l'outil
P10	Mauvais affutage
P11	Pièce
P12	Mauvais positionnement de la pièce
P13	Caractéristiques non conformes de la pièce brute
P14	Contrôle Système de Vision (Caméra)
P15	Vision floue (Faux résultats)
P16	Mauvais réglage caméra - traitement erroné

Chapitre 4 : Application – Usinage robotisé en parachèvement

4.4.3 Simulations et résultats

Les simulations et résultats obtenus sont illustrés à la figure 4.10. Le marquage correspondant est $M_1 = [1100000000110000]$, l'état de la pièce non conforme est en fonctionnement avec faute, le Robot KUKA KR 120 est défaillant dans le deuxième niveau de l'AdF ; ainsi que le mauvais positionnement de la pièce dans le troisième niveau.

- Les indicateurs numériques du Robot KUKA KR 120 et du mauvais positionnement de la pièce sont en 1 : dans « État Robot KUKA KR 120 » marqué « Robot KUKA KR 120 Défaillant » ; dans « État Mauvais positionnement de la pièce » marquer « Mauvais positionnement de la pièce ».
- Ainsi, respectivement, la place (P2) et la place (P12) sont colorées en gris, et leur signal est dans l'état 1.
- Si la place (P2) est défaillante alors (P1) est défaillante ; Si la place (P12) est défaillante alors (P11) est d'abord défaillante et ensuite (P1) est aussi défaillante : S'il y a une place défaillante, le processus de diagnostic déclenché rend le système défaillant.

Pour faire une analyse déductive sur notre RdP, nous procédons par franchissement de la transition (T1), alors la place (P1) comporte un jeton. S'il y a un jeton dans la place (P1) alors il y a lieu d'effectuer le franchissement directement des transitions (T2) et (T5). Si (T2) est franchi, alors les places (P2) et (P1) comportent un jeton chacune. Ainsi, en franchissant (T5), les places (P11) et (P1) comportent un jeton chacune. Si la place (P11) comporte un jeton alors (P12) comporte également un jeton après le franchissement de la transition (T13) : Ce processus de diagnostic s'obtient grâce aux arcs de retours constituant notre réseau de Petri.

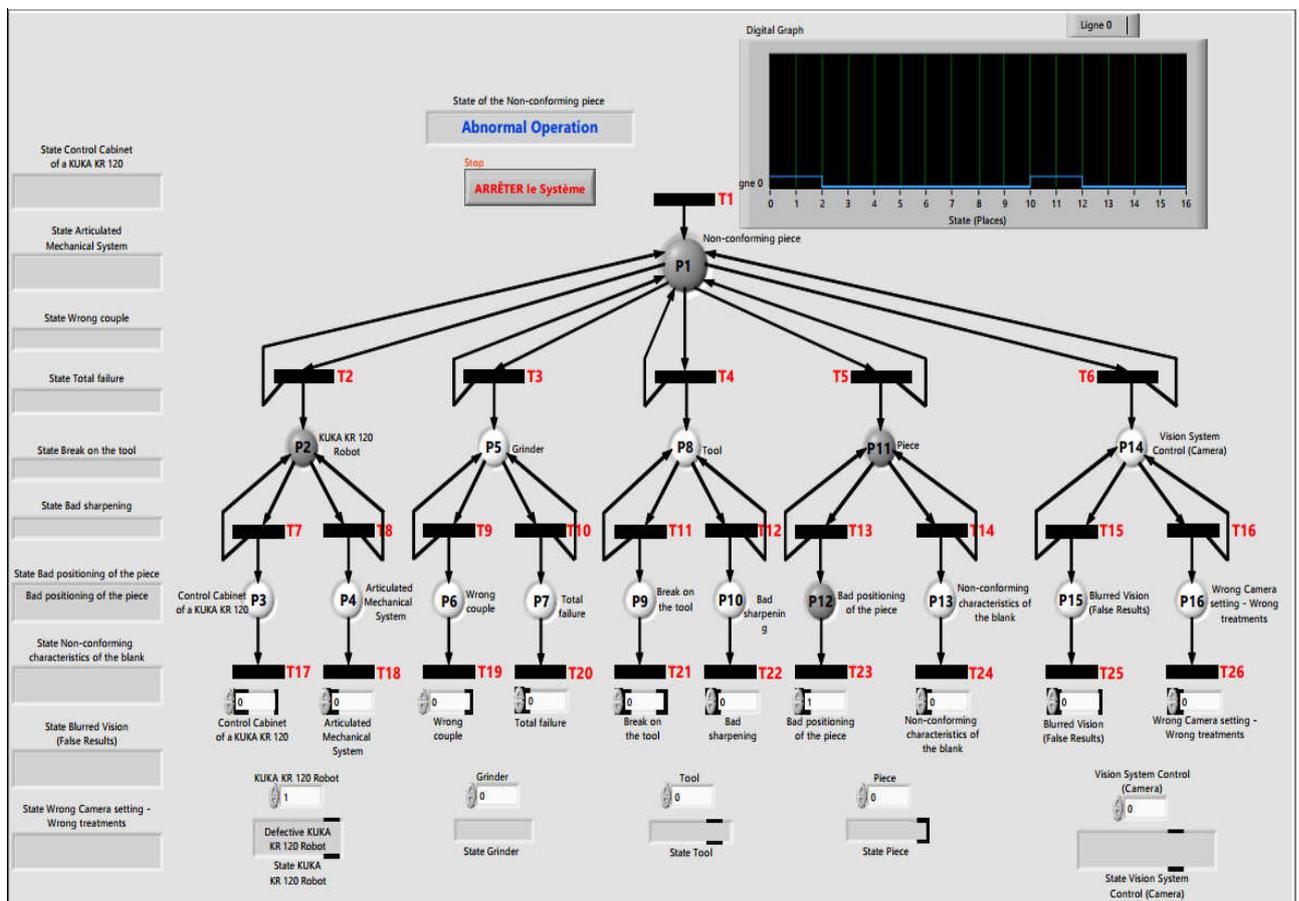


Figure 4.10. Block Face avant du LabView du système RdP- Modélisation avec faute

4.5 Étude de cas 2 - RdP λ

Nous proposons une extension Lambda des réseaux de Petri (RdP λ), spécifique à la modélisation et à l'analyse des activités de surveillance des systèmes. Les connaissances incertaines associées à ces activités nécessitent des méthodes de raisonnement et de modélisation spécifiques, adaptées aux différents taux de défaillances.

En coopération avec le modèle réseau de Petri ordinaires et RdP capacitaires qui modélisent le système à surveiller, ce nouvel outil permet de réaliser un diagnostic complet des localisations des défauts et des dégradations du système. L'approche de RdP λ fournit des informations plus détaillées sur l'état de fonctionnement du système surveillé.

4.5.1 Formalisation RdP λ

Ce nouvel outil, RdP λ décrit le fonctionnement du système non autonome. Son fonctionnement est conditionné par des taux de défaillances. Un RdP λ est composé de deux parties : une partie statique et une partie dynamique. La partie statique définit la structure du RdP λ où sont stockées les données et comment ces données interagissent les unes avec les autres. La partie dynamique définit l'état initial du système. En effet, un même RdP λ n'aura pas le même comportement selon son état initial, il est donc important de séparer les deux concepts.

La modélisation de notre système de diagnostic est réalisée par l'association des réseaux de Petri généralisés, particuliers et de haut niveau, en supposant que les défaillances possibles sont connues a priori et modélisées par des mécanismes spécifiques. Notre approche prend en considération la modélisation par RdP λ au niveau des transitions.

Pour modéliser cette fonction de surveillance, nous utilisons une extension du RdP qui intègre, à travers l'aspect λ , le taux de défaillance dans le système surveillé. Le RdP λ est orienté vers la modélisation de la fiabilité. Plus précisément, il sert à l'orientation du RdP λ pour fournir une représentation quantitative et dynamique de la fiabilité d'un système complexe et ce, à partir de l'information qualitative contenue dans un AdF. Cette approche permet d'extraire des données à partir de l'expression logique de l'AdF du système surveillé. L'outil RdP λ modélise l'ensemble du raisonnement logique de l'AdF en respectant les concepts spécifiques de l'expression logique. L'aspect analyse offre une information affinée au niveau de chaque défaut par le transfert des signaux des défauts. Enfin, l'utilisation des propriétés génériques et mathématiques assure la vérification des résultats pour valider le système.

4.5.2 Paramétrage de la cellule d'usinage de parachèvement

De nombreux auteurs se sont intéressés au domaine de la fiabilité des outils coupants et à la modélisation de la rugosité de surface des pièces usinées.

(Abdoulaye et al., 2019) ont étudié et modélisé une nouvelle fonction de risque de base dénommée fonction de hasard proportionnel GEV. Cette fonction généralisée aux valeurs extrêmes est obtenue grâce à la fonction de Gumbel et à la propriété d'être non monotone, fonction croissante puis décroissante, ainsi capable de prendre en compte le comportement de l'outil de coupe avec des différents mécanismes d'auto-réparation et d'auto-récupération après un petit bris. Les auteurs introduisent des paramètres prenant en compte les conditions de fonctionnement et les conditions environnementales, notamment les signaux de vibration, la dureté du matériau et la lubrification/refroidissement. Leurs résultats montrent l'impact de toutes ces variables sur la rugosité de surface des pièces usinées

Chapitre 4 : Application – Usinage robotisé en parachèvement

- 1) Signal de vibration : Deux types de vibrations se produisent pendant l'usinage qui sont les vibrations soutenues et auto-entretenues. Les vibrations auto-entretenues se caractérisent par l'absence de forces externes périodiques mais dépendent du processus de formation des copeaux. D'autre part, les vibrations obtenues proviennent des forces externes périodiques, telles que celles agissant sur la pièce lors de la coupe intermittente.

(Upadhyay et al., 2013) ont étudié des conditions similaires d'outil de coupe et l'amplitude d'accélération des vibrations de l'outil dans les directions axiale, radiale et tangentielle. Le tableau 4.2 montre l'impact des valeurs expérimentales de l'amplitude d'accélération de la vibration (valeur moyenne quadratique (RMS)) dans les directions axiale (V_x), radiale (V_y) et tangentielle (V_z) sur la rugosité de surface (R_a). Le cas le plus défavorable est la variation de V_y (g).

- 2) Dureté du matériau : Une diminution de la dureté du matériau utilisé et la fiabilité du traitement thermique appliqué entraînent une détérioration de la rugosité de surface.

(Vaxevanidis et al., 2015) ont étudié l'influence des conditions de l'outil de coupe et de la dureté du matériau sur la rugosité de surface. Les résultats obtenus sont donnés dans le tableau 4.2.

- 3) Conditions de lubrification/refroidissement : Une température de coupe élevée pendant l'usinage peut entraîner une qualité de surface inacceptable. Comparées à la coupe à sec, les conditions de lubrification/refroidissement conduisent à une amélioration significative de la rugosité de surface.

(Xavior & Adithan, 2009) montrent tous les impacts des conditions de coupe sur la rugosité de surface et l'usure de l'outil dans le processus de tournage. La contribution du refroidissement/lubrification est d'environ 14 %.

(Abdoulaye et al., 2019) ont étudié l'influence des vibrations, des défauts de dureté du matériau et de la lubrification, les résultats sont présentés dans le tableau 4.2.

On peut déduire que toutes les contributions des variables détériorent la fonction hazard de base.

Tableau 4. 2. Paramètres d'usinage adoptés dans les travaux

Conditions de coupe				
Vitesse de coupe (V_c) (m/min)	Vitesse d'avance (f) (mm/rev)		Profondeur de coupe (d) (mm)	
250	0.1		0.5	
Impact des vibrations sur la rugosité de surface				
	V_x (g)	V_y (g)	V_z (g)	R_a (μm)
Expérience 1	0,37	3,92	1,63	1,51
Expérience 2	0,39	4,84	1,65	1,73
Impact de la dureté du matériau sur la rugosité de surface				
	Dureté (HB)		R_a (μm)	
Expérience 1	130		1,53	
Expérience 2	240		1,35	
Conditions de lubrification/refroidissement				
Signal de vibration (covariable X_1)	Dureté du matériau (covariable X_2)		Lubrification (covariable X_3)	
$\varphi(\beta_1 X_1) = 1,15$	$\varphi(\beta_2 X_2) = 1,13$		$\varphi(\beta_3 X_3) = 0,86$	

4.5.3 Analyse par AdF

L'Arbre des Fautes (AdF) de la cellule robotisée est toujours la pièce centrale de notre stratégie basée sur les RdP et est présentée en figure 4.11. L'analyse et la recherche de l'événement redouté dans notre AdF (désigné aussi événement sommet de l'arbre), mettent en évidence la non-conformité de la pièce cylindrique usinée, (a) figure 4.11, dans la cellule robotisée. Si l'on recherche la cause d'un événement indésirable, il peut être dû à un défaut sur cet élément même ou à un défaut sur tout autre élément du système.

Nous avons utilisé, ici aussi, le logiciel CABTREE, pour construire et traiter nos arbres de fautes. Nous avons limité notre étude à deux niveaux, qui font apparaître les premiers événements élémentaires, comme présentée en figure 4.11 :

- Premier niveau : Robot KUKA KR120 défaillant, (b) ; Electro-broche défaillante, (c) ; Outil défaillant, (d) ; Pièce défaillante, (e) ; Système de lubrification défaillant (Pompe), (f).
- Deuxième niveau : Vitesse d'avance (f) défaillante, (b1) ; Profondeur de coupe (d) défaillante, (b2) ; Vibration dans les directions axiales (Vx), (b3) ; Vibration dans les directions radiales (Vy), (b4) ; Vibration dans les directions tangentielles (Vz), (b5) ; Vitesse de coupe (Vc) défaillante, (c1) ; Casse de l'outil, (d1) ; Mauvais positionnement de la pièce, (e1) ; Dureté défaillante, (e2) ; Faible débit, (f1) ; Mauvais liquide, (f2).

Les portes logiques peuvent modéliser la fonction booléenne F de l'événement redouté de notre arbre de fautes (AdF). Dans notre travail, nous avons utilisé la porte logique "OU". Pour illustrer notre approche, nous considérons l'équation logique F de l'arbre de fautes :

$$F = (b1 \text{ OU } b2 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (c1 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (d1 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (e1 \text{ OU } e2) \text{ OU } (f1 \text{ OU } f2)$$

L'opérateur (+) \Leftrightarrow OR représente l'union des variables logiques {a, b, c, d, e, f, b1, b2, b3, b4, b5, c1, d1, e1, e2, f1, f2}.

$$F = \left(\underbrace{(b1 + b2 + b3 + b4 + b5)}_b + \underbrace{(c1 + c3 + b4 + b5)}_c + \underbrace{(d1 + b3 + b4 + b5)}_d + \underbrace{(e1 + e2)}_e + \underbrace{(f1 + f2)}_f \right) \quad (4.1)$$

Tel que :

$$b = b1 + b2 + b3 + b4 + b5$$

$$c = c1 + b3 + b4 + b5$$

$$d = d1 + b3 + b4 + b5$$

$$e = e1 + e2$$

$$f = f1 + f2$$

$$F = [b + c + d + e + f] \quad (4.2)$$

$$F = a \quad (4.3)$$

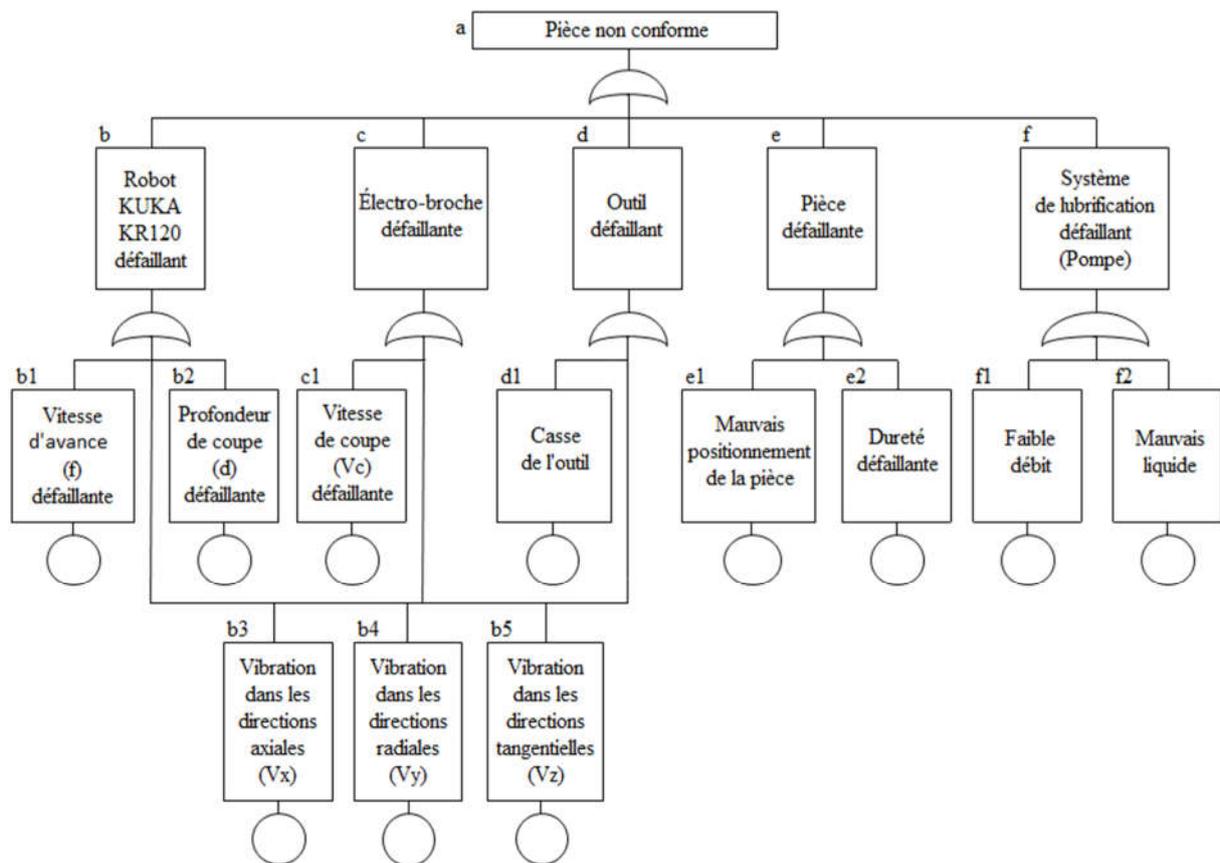


Figure 4.11. AdF de la cellule robotisée, correspondant à F

4.5.3.1 Analyse par RdPλ

La cellule robotisée d'usinage est représentée à la figure 4.13. Son fonctionnement utilisant le RdP est présenté à la figure 4.12. Notre modèle RdP est représenté par 17 places et 28 transitions, dénotées respectivement P1 à P17 et les transitions de T1 à T28. Le nombre de jetons dans la place P_i représente le nombre de défaillances dans la cellule robotisée. Les transitions T_i situées entre les places P_i sont des transitions pour toutes les actions réalisées par la cellule robotisée avec tous les branchements. Ces transitions sont immédiates. Le nombre de jetons dans la place P_i correspond au nombre d'éléments défaillants dans notre cellule robotisée. Les poids des arcs sont indiqués sur le modèle, en regard des arcs. L'absence de notation signifie que l'arc en question est implicitement pondéré à 1.

Notre RdP peut représenter l'assemblage et le désassemblage successifs d'un seul élément (P1). Ainsi, il existe des processus à deux états (Arrêt-Marche) et le passage d'un état à l'autre mobilise une ressource, symbolisée par le jeton contenu dans les arcs, qui a été ajouté entre les places et les transitions. Contenant une place de source (P1) et une transition de source (T1), cette transition est toujours sensibilisée, mais avec un compteur de capacité (Cap (P1) = 5). Il y a 10 transitions puits dans le réseau de Petri, à savoir T18, T19, T20, T21, T22, T23, T24, T25, T26, T27, T28. Ces transitions puits peuvent être franchies si elles sont sensibilisées. Lorsqu'une transition puits est franchie, des jetons sont retirés de ses places d'entrée, mais aucun jeton n'est créé.

Notre modèle de RdP a été enrichi en intégrant la notion de capacité, permettant une gestion plus précise des ressources. Contrairement au RdP généralisé où le nombre de jetons par place

Chapitre 4 : Application – Usinage robotisé en parachèvement

n'est soumis à aucune contrainte. Contrairement, dans le cas RdPs à capacités (RdP particulier) où est associée une capacité à chaque place. Cette capacité est définie comme un nombre entier strictement positif. Le franchissement d'une transition d'entrée d'une place P_i est autorisé uniquement si le nombre de jetons résultant dans P_i ne dépasse pas sa capacité maximale, notée $Cap(P_i)$.

Dans notre exemple donné en figure 4.12, le franchissement de T1 conduit à 5 jetons dans P1 d'où T1 ne peut plus être franchie.

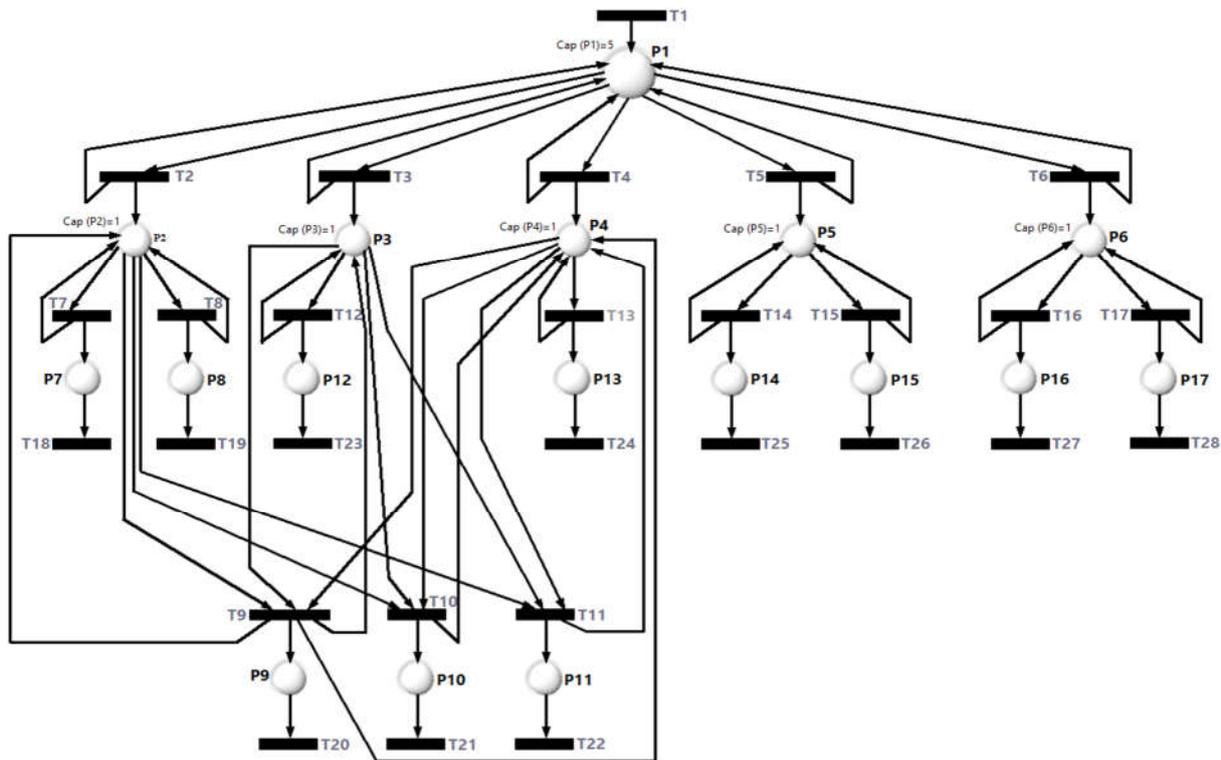


Figure 4. 12. RdP avant le franchissement

Notre objectif est de contrôler et automatiser le système AdF considéré en utilisant son modèle RdP et en utilisant des taux de défaillance λ . Pour ce faire, il est nécessaire de convertir le modèle RdP illustré à la figure 4.14 en son modèle équivalent $RdP\lambda$. La transformation est possible à partir de modèles RdP généralisés et particulières vers les RdP de haut niveau ($RdP\lambda$) et illustrée dans les figures 4.13 et 4.14.

Notre $RdP\lambda$ avant le franchissement est représenté à la figure 4.13.

Chapitre 4 : Application – Usinage robotisé en parachèvement

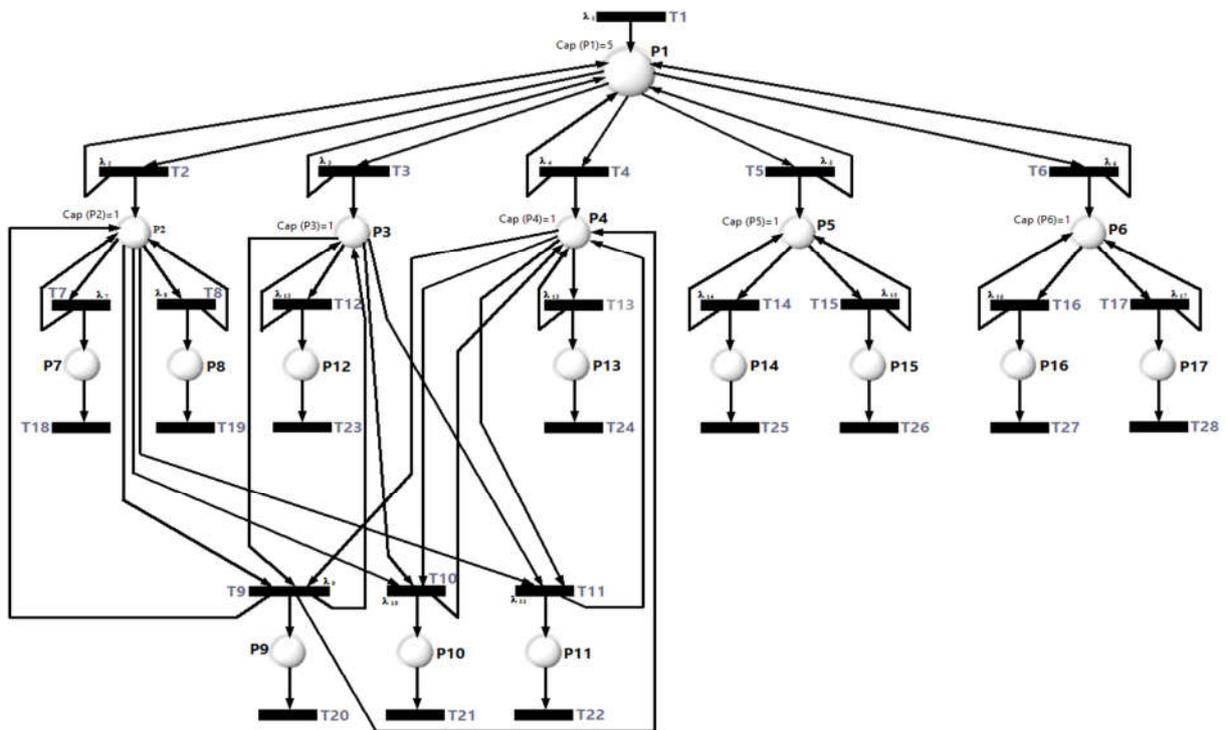


Figure 4. 13. $RdP\lambda$ avant le franchissement

Notre $RdP\lambda$ après le franchissement est représenté à la figure 4.14.

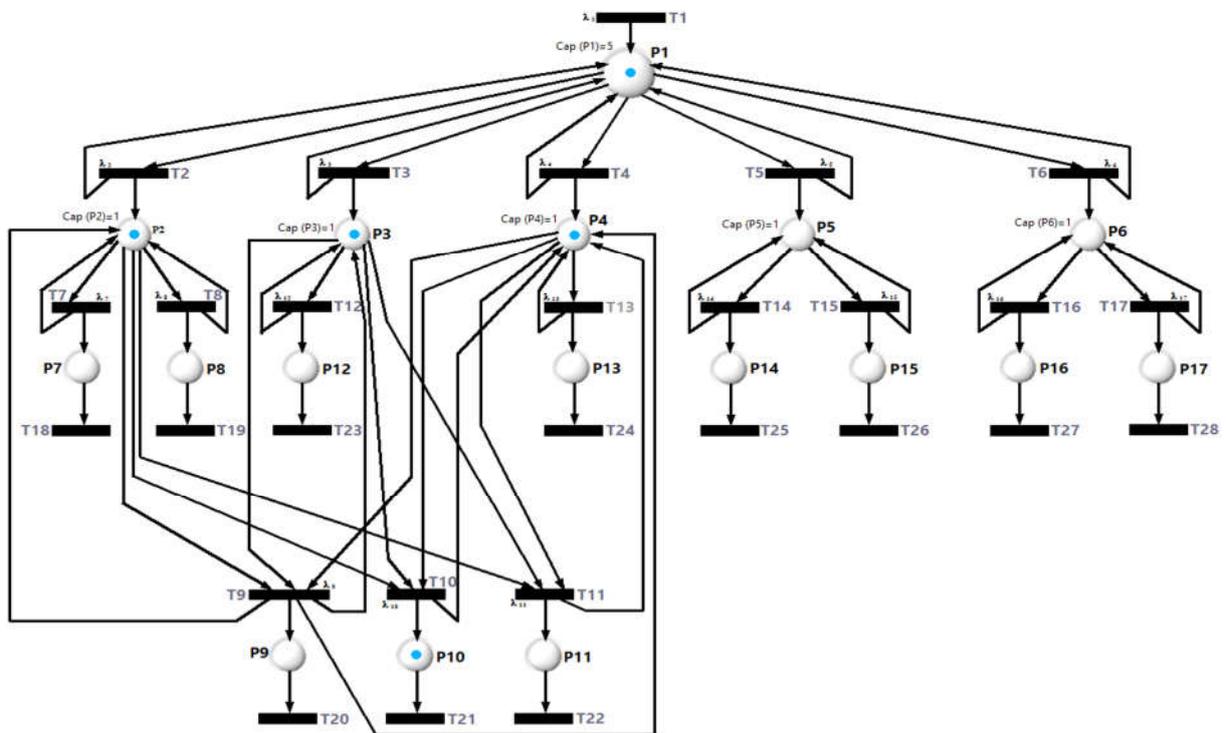


Figure 4. 14. $RdP\lambda$ après le franchissement

Les transitions (T1, T2, ..., T28) de notre système réseau de Petri sont matérialisées par des capteurs. Les messages utilisés sur les transitions sont des taux de défaillance entre 0 et 1, permettant la communication entre les différentes transitions lors de leur franchissement, lorsque toutes les places qui lui sont juxtaposées (ou toutes les places d'entrée de la transition)

Chapitre 4 : Application – Usinage robotisé en parachèvement

contiennent suffisamment de jetons. Ces taux de défaillance sont utilisés pour mettre à jour dynamiquement les probabilités de franchissement des transitions et ainsi simuler l'évolution du système en fonction des défaillances potentielles au sein du réseau de Petri.

Cela permet de modéliser le plus simplement possible le comportement dynamique de tous les états du système. Le tableau 4.3 décrit la signification de chaque place. Les différents taux de défaillance utilisés sur le Réseau de Petri Lambda de notre système sont indiqués dans le tableau 4.4.

Tableau 4. 3. Signification des places

Les places	La signification de chaque place
P1	Pièce non Conforme
P2	Robot KUKA KR 120 défaillant
P3	Électro-broche défaillante
P4	Outil défaillant
P5	Pièce défaillante
P6	Système de lubrification défaillant (Pompe)
P7	Vitesse d'avance (f) défaillante
P8	Profondeur de coupe (d) défaillante
P9	Vibration dans les directions axiales (V_x)
P10	Vibration dans les directions radiales (V_y)
P11	Vibration dans les directions tangentielles (V_z)
P12	Vitesse de coupe (V_c) défaillante
P13	Casse de l'outil
P14	Mauvais positionnement de la pièce
P15	Dureté défaillante
P16	Faible débit
P17	Mauvais liquide

4.5.4 Implémentation LabView

Nous avons appliqué notre technique de transformation d'arbre des fautes (AdF) en un Réseau de Petri Lambda (RdP λ). Nous avons utilisé comme plateforme l'environnement LabView pour la modélisation et la simulation. LabView est basé sur un environnement de développement graphique de « National Instruments », et est utilisé principalement pour le contrôle d'instruments et l'automatisme industriel.

4.5.4.1 Implémentation AdF – LabView

Nous avons proposé l'implémentation dans l'environnement LabView du Réseau de Petri Lambda de l'étude de cas avec des taux de défaillance des différents composants constituant notre AdF. Les figures 4.15 et 4.16 suivantes représentent notre modélisation.

- 1) L'implémentation modélisée ET-RdP λ sous LabView se décline de la façon suivante, figure 4.15 :

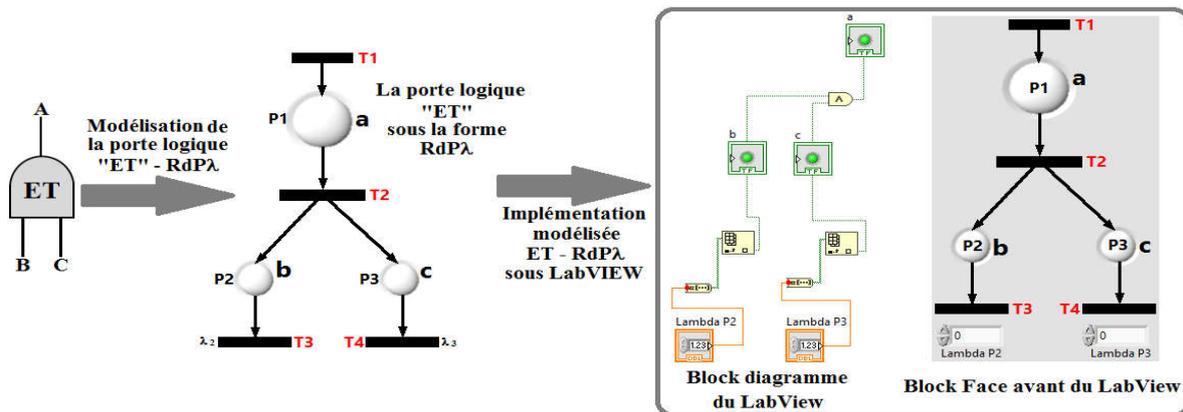


Figure 4.15. Transformation de la porte logique « ET » de l'AdF en RdPλ

- 2) L'implémentation modélisée OU-RdPλ sous LabVIEW se déclinera de la façon suivante, figure 4.16 :

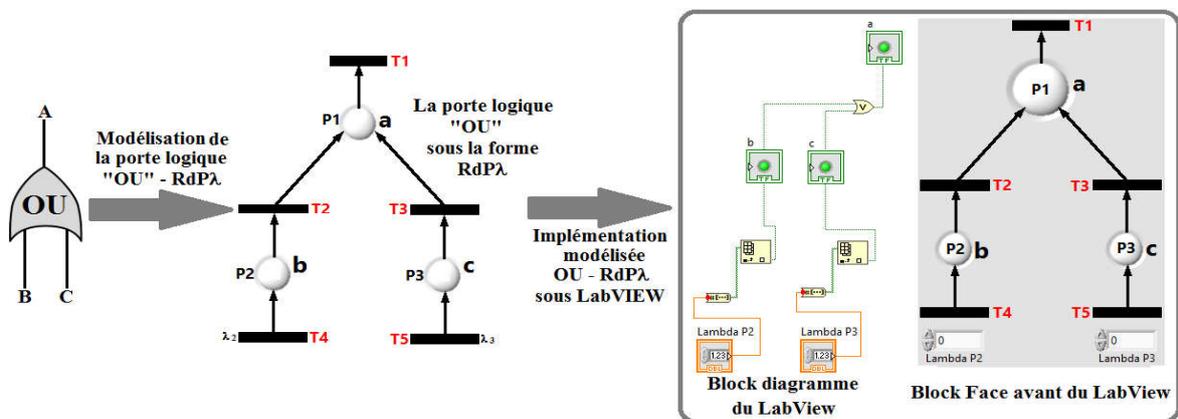


Figure 4.16. Transformation de la porte logique « OU » de l'AdF en RdPλ

4.5.4.2 Implémentation RdPλ – LabVIEW

Nous avons associé notre RdPλ en une machine d'état LabVIEW. La structure obtenue est illustrée dans les figures 4.17 et 4.20. La face-avant est l'interface utilisateur du VIs (instruments virtuels), de notre système. Elle est représentée sur les figures 4.17 et 4.20, et décrit une application d'analyse appelée Mode de Défaillance Dominant. Le RdPλ contient 17 places qui sont des LEDs (Diode Électroluminescente circulaire) contenant 2 phases de lumière, blanc et gris et 28 transitions. Chaque place représente un événement de notre AdF et décrit son état, des entrées (les commandes) et des sorties (les indicateurs) du programme. Ces états sont modélisés par des String Indicator et un autre String Indicator pour afficher l'état de notre système (État de la pièce non conforme). Cet indicateur permet d'afficher ou Fonctionnement Normal ou Fonctionnement Anormal.

Notre application consiste en une Box qui contient un Numeric Indicator pour la somme des différents taux de défaillance de notre AdF. Il comprend également deux autres Numeric Indicators, le premier pour afficher le taux de défaillance le plus élevé du niveau 1 au AdF et le second pour afficher le taux de défaillance le plus élevé du niveau 2 au AdF. 11 Numeric Control et 5 Numeric Indicators sont matérialisés par des taux de défaillance. Les différents taux de défaillance sont des valeurs entre 0 et 1.

Les variables d'entrées sont les taux de défaillance de chaque composant. Elles sont étiquetées

Chapitre 4 : Application – Usinage robotisé en parachèvement

comme "Numeric Control" dans LabVIEW et les résultats sont notés dans "Digital Indicator" sur le block face avant.

Un Digital Waveform Graph affiche les données en tant qu'impulsions, générateur de nombres binaires qui va simuler le test du marquage. S'il y a une marque dans une place, le signal affiche l'état 1 sinon l'état 0. L'axe des X est l'échelle des États (places) et l'axe des Y l'échelle des Marquages. Ce graphique constitue une conversion d'une entrée booléenne en une forme d'onde numérique dans LabVIEW.

Un bouton, STOP the System, met la cellule robotisée à l'arrêt s'il y a une défaillance dans la cellule robotisée.

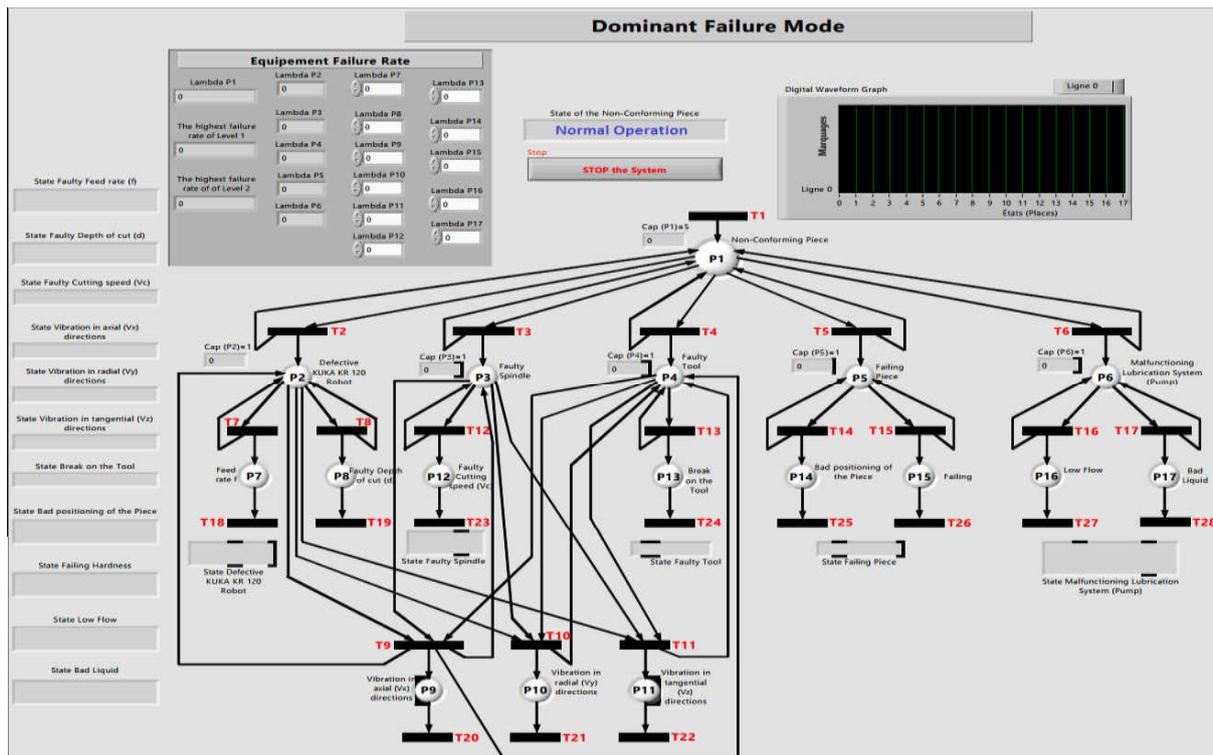


Figure 4.17. Block Face avant du LabVIEW du système RdPλ- Modélisation sans faute

Le block diagramme, figure 4.18, représente le programme de l'application écrit sous la forme d'un diagramme de flux de données. Cette figure illustre comment matérialiser la commande et l'indicateur par un afficheur numérique pour l'état 0 ou 1 dans le block diagramme du LabVIEW.

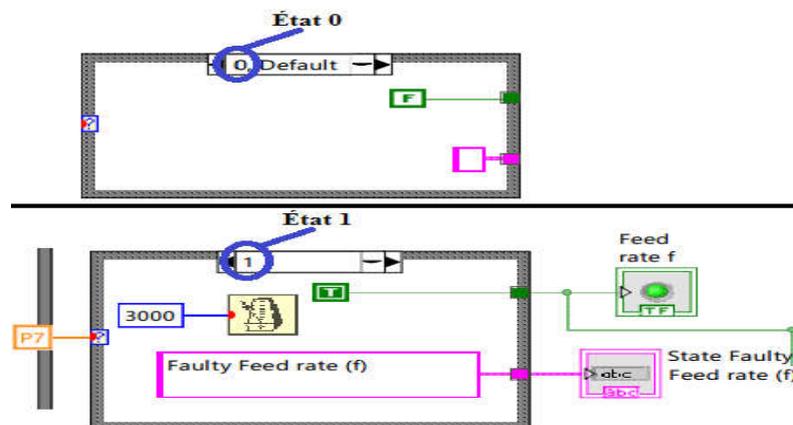


Figure 4.18. Block diagramme du LabVIEW du système RdPλ

4.5.5 Simulations et résultats

La simulation du système de diagnostic proposé, figure 4.19, a été réalisée en trois étapes essentielles. Tout d’abord, nous avons évalué les taux de défaillance de tous les composants du système. Ensuite, nous avons utilisé LabView pour effectuer la simulation permettant d’observer la distribution des états du système. Enfin, dans la troisième étape, nous avons identifié successivement le marquage de la simulation du Réseau de Petri Lambda, obtenu ses propriétés génériques pour une vérification automatique, et enfin déterminé sa matrice d’incidence et obtenu son vecteur de marquage nécessaire pour valider le modèle.

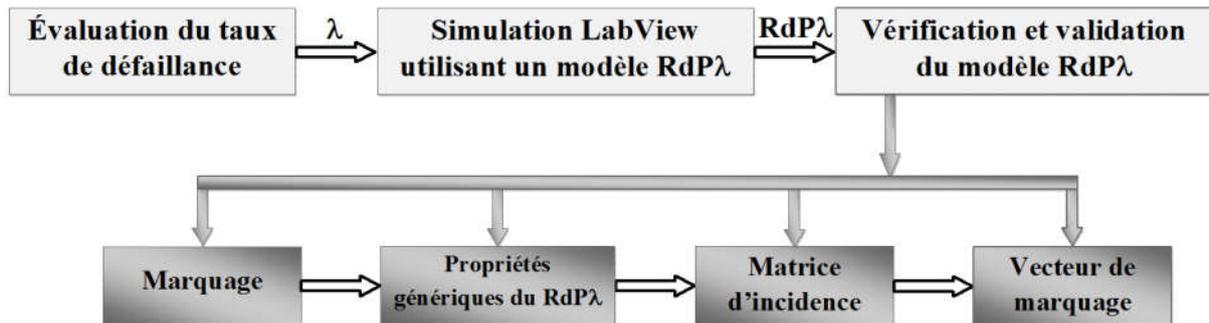


Figure 4. 19. Étapes du système de diagnostic proposé

4.5.5.1 Taux de défaillance

Selon une expertise hiérarchique de robustesse et grâce aux valeurs indiquées dans le tableau 4.2, les données de défaillances de chaque composant sont présentées dans le tableau 4.4. Les vibrations sont les événements les plus critiques. Les vibrations dans les directions radiales présentent le taux de défaillance le plus élevé ($\lambda_{10} = 0.006$).

Tableau 4. 4. Taux de défaillance du niveau 2 – AdF – RdPλ

Niveau 2 - AdF - RdPλ			
λ_7	0.0001	λ_{13}	0.0003
λ_8	0.0001	λ_{14}	0.0003
λ_9	0.0004	λ_{15}	0.0002
λ_{10}	0.0006	λ_{16}	0.0002
λ_{11}	0.0005	λ_{17}	0.0003
λ_{12}	0.0001		

Selon les équations (4.1), (4.2) et (4.3) et le tableau 4.4, nous avons obtenu les taux de défaillance suivants présentés dans le tableau 4.5. L’outil défaillant est l’événement le plus critique avec un taux de défaillance ($\lambda_4 = 0.0018$).

Tableau 4. 5. Taux de défaillance du niveau 1 et de l’événement redouté – AdF – RdPλ

Niveau 2 - AdF - RdPλ				
λ_2	λ_3	λ_4	λ_5	λ_6
0.0017	0.0016	0.0018	0.0005	0.0005
Événement redouté - AdF – RdPλ				
λ_1	0.0061			

4.5.5.2 Simulation LabView

Les simulations et résultats obtenus sont illustrés à la figure 4.20. Nous utilisons des jetons dans les places de graphe pour signaler l’état de chaque ressource à un moment donné. Ces jetons

Chapitre 4 : Application – Usinage robotisé en parachèvement

sont marqués en couleur grise.

Au deuxième niveau AdF et correspondant à son taux de défaillance, le composant « Vibration dans les directions radiales (Vy) » est en mode de défaillance, donc l'état correspondant est activé.

Selon les caractéristiques de diagnostic de l'AdF et des équations (4.1), (4.2), (4.3) l'évènement redouté est $\lambda_1 = 0.0061$. Après la comparaison de tous les taux de défaillance du niveau 1, le taux de défaillance le plus élevé est $\lambda_4 = 0.0018$, donc le taux de défaillance le plus élevé du niveau 2 est $\lambda_{10} = 0.0006$.

- La place (P10) est colorée en gris, et son signal est dans l'état 1.
- Si la place (P10) est défaillante alors (P4) est d'abord défaillante et ensuite (P1) est défaillante aussi, ces deux places sont colorées en gris, et leur signal est dans l'état 1. S'il y a une place défaillante, le processus de diagnostic déclenché rend le système défaillant. Ainsi, la capacité $Cap(P4) = 1$ de la place (P4) est affichée 1 et la capacité finie $Cap(P1) = 5$ de la place (P1) est affichée 1.
- Pour réaliser une analyse déductive dans notre $RdP\lambda$, nous procédons par franchissement de la transition (T1), alors la place (P1) comporte un jeton. S'il y a un jeton dans la place (P1) alors il y a lieu de faire le franchissement directement de la transition (T4). Si (T4) est franchi, alors la place (P4) comporte un jeton. Si la place (P4) a un jeton, alors (P10) comporte aussi un jeton après le franchissement de la transition (T10). Ce processus de diagnostic s'obtient grâce aux arcs de retour constituant notre réseau de Petri lambda.

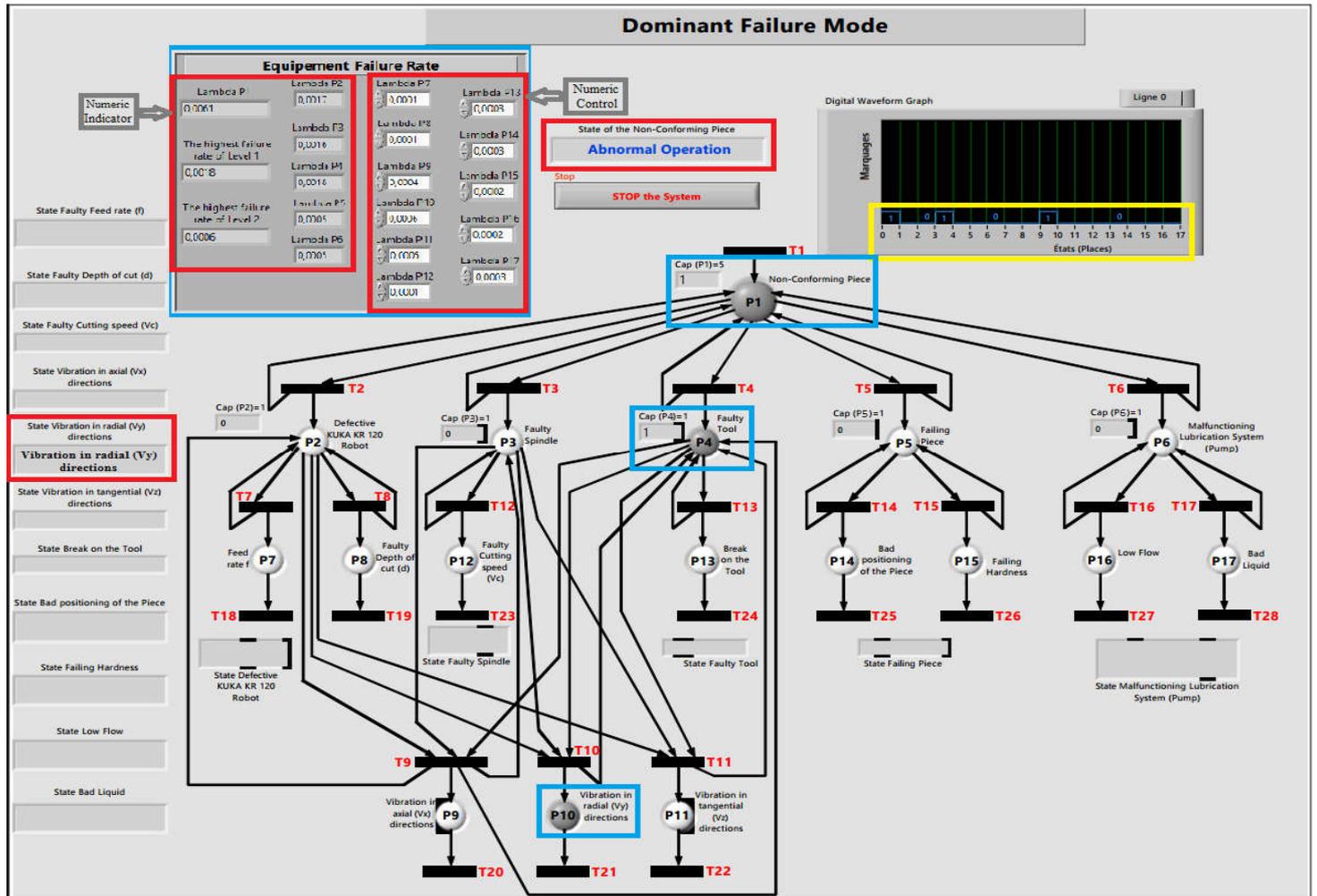


Figure 4. 20. Block Face avant sous LabView du système RdPλ - Modélisation avec faute

4.5.5.3 Marquage

À ce niveau, nous nous intéressons au pilotage d'une cellule robotisée d'usinage présentée figure 4.3 et modélisée par un arbre des fautes figure 4.11. Elle est associée au RdPλ de la figure 4.13. Le résultat d'après le franchissement (après une ou des défaillances) est le modèle représenté à la figure 4.14.

Nous avons également défini le marquage pour interpréter les résultats de la simulation du réseau de Petri lambda. Il nous permet de comprendre le comportement du système, d'identifier les états du système à différents instants et de suivre son évolution en suivant les transitions.

Selon l'équation (3.1) on a :

Le marquage initial de notre RdPλ, correspondant à la figure 4.16, est : $M_0 = [000000000000000000]$.

Le marquage de notre RdPλ après le franchissement, correspondant à la figure 4.18, est : $M_1 = [100100000100000000]$.

$P = \{P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12, P13, P14, P15, P16, P17\}$

$T = \{T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15, T16, T17, T18, T19, T20, T21, T22, T23, T24, T25, T26, T27, T28\}$

Chapitre 4 : Application – Usinage robotisé en parachèvement

4.5.5.4 Propriétés génériques du RdPλ

Nous avons utilisé les propriétés génériques pour vérifier la cohérence du modèle et détecter d'éventuels comportements indésirables. Les propriétés génériques sont des caractéristiques fondamentales des réseaux de Petri qui permettent d'analyser leur comportement. Ces propriétés incluent la bornitude, le sauf, la vivacité, le blocage, la réinitialisabilité, la répétitivité, le conflit, l'accessibilité et la sûreté.

Nous avons obtenu les propriétés génériques suivantes pour la vérification automatique de notre système, notre RdPλ est non borné, mais sauf. Un RdPλ vivant, sans blocage, ni réinitialisable, ni répétitif, sans conflit, accessible et sûreté, avec graphe des marquages infini.

4.5.5.5 Matrice d'incidence Avant du RdPλ

Suivant l'équation (3.11), nous avons obtenu la matrice d'incidence avant (W^+) de notre réseau de Petri Lambda (RdPλ). C'est une matrice de 17 lignes et 28 colonnes représentée par :

$$W^+ = \begin{matrix} & T1 & T2 & T3 & T4 & T5 & T6 & T7 & T8 & T9 & T10 & T11 & T12 & T13 & T14 & T15 & T16 & T17 & T18 & T19 & T20 & T21 & T22 & T23 & T24 & T25 & T26 & T27 & T28 \\ P1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ P2 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ P8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ P9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P13 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P14 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P16 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P17 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

4.5.5.6 Matrice d'incidence Arrière du RdPλ

Suivant l'équation (3.12), nous avons obtenu la matrice d'incidence arrière (W^-) de notre réseau de Petri Lambda (RdPλ). C'est une matrice de 17 lignes et 28 colonnes représentée par:

$$W^- = \begin{matrix} & T1 & T2 & T3 & T4 & T5 & T6 & T7 & T8 & T9 & T10 & T11 & T12 & T13 & T14 & T15 & T16 & T17 & T18 & T19 & T20 & T21 & T22 & T23 & T24 & T25 & T26 & T27 & T28 \\ P1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ P2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P10 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P12 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P13 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ P14 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ P15 & 0 & 1 & 0 & 0 & 0 & 0 \\ P16 & 0 & 1 & 0 & 0 \\ P17 & 0 & 1 & 1 \end{matrix}$$

4.5.5.7 Matrice d'incidence du RdPλ

La matrice d'incidence représente les relations entre les places et les transitions dans un réseau de Petri, permettant ainsi de calculer le marquage du système après une transition.

La matrice d'incidence (W) de notre réseau de Petri Lambda (RdPλ), après l'application de

4.6 Conclusion

La modélisation et la simulation par des RdPs sont des outils puissants pour évaluer les performances des systèmes complexes et ils constituent un formalisme mathématique efficace pour modéliser leurs défaillances.

Dans ce chapitre, nous avons proposé et présenté le développement d'une stratégie de diagnostic pour deux cellules robotisées d'usinage de parachèvement, en utilisant des RdPs générés à partir d'AdFs des cellules étudiées, dans l'environnement LabView. La première étude est basée sur une analyse qualitative pour le parachèvement de pièces parallélépipédiques. La seconde étude est basée sur une analyse qualitative et quantitative pour le parachèvement de pièces cylindriques.

Dans la seconde étude, après avoir introduit dans notre modèle les signaux de vibrations ainsi que les conditions de coupe propres à la cellule, nous avons tout d'abord transformé son AdF en RdP à l'aide des équivalences et ensuite, nous avons déterminé les taux de défaillance à considérer sur les transitions pour obtenir notre RdP λ .

Dans les deux cas d'étude, nous avons effectué des simulations de diagnostic de fautes après implémentation sous environnement LabView. Les résultats obtenus, essentiellement dans la deuxième étude, se révèlent pleinement satisfaisants.

Conclusion Générale

Conclusion Générale

Parmi les nombreuses recherches étudiées visant à améliorer les procédés de parachèvement des pièces par usinage robotisé, notre travail s'inscrit dans une contribution à leur évolution. Plus particulièrement. Nous nous sommes attachés à développer une stratégie de surveillance, dans le cadre de la collaboration entre le Laboratoire Génie de Production et Maintenance Industrielle (LGPMI) de l'Institut de Maintenance et de Sécurité Industrielle et le Laboratoire Génie de Production (LGP) de l'École Nationale d'Ingénieurs de Tarbes (ENIT).

Nous avons en premier lieu passé en revue les différentes notions liées aux robots industriels, aux cellules d'usinage robotisées nécessaires à la compréhension des différents types d'opération de parachèvement ainsi qu'à la maintenance industrielle. Nous avons procédé à un état de l'art et une classification des méthodes et modèles de surveillance industrielle. Nous avons montré que les techniques avec modèles restent des techniques puissantes et très précises bien que leur mise en œuvre s'avère complexe.

L'outil de base choisi pour la modélisation de notre stratégie de surveillance sont les Réseaux de Petri. Nous avons donc détaillé tous les aspects liés à ce formalisme, à savoir l'historique des RdPs, leurs notions de base ainsi que leurs propriétés mathématiques et génériques, leurs structures générales et particulières, leurs avantages et inconvénients, et les différents types de RdPs de haut niveau.

Dans notre contribution, nous avons proposé et présenté le développement d'une stratégie de diagnostic pour deux cellules robotisées d'usinage de parachèvement, en utilisant des RdPs générés à partir d'AdFs des cellules étudiées dans l'environnement LabView. La première étude est basée sur une analyse qualitative pour le parachèvement de pièces parallélépipédiques. La seconde étude, elle, est basée sur une analyse qualitative et quantitative pour le parachèvement de pièces cylindriques.

Dans la seconde étude, après avoir introduit dans notre modèle, les signaux de vibrations ainsi que les conditions de coupe propres à la cellule, nous avons tout d'abord transformé son AdF en RdP à l'aide des équivalences, et ensuite nous avons déterminé les taux de défaillance à considérer sur les transitions pour obtenir notre RdP λ .

Dans les deux cas d'études effectuées, nous avons réalisé des simulations de diagnostic de fautes après implémentation sous environnement LabView. Les résultats obtenus, essentiellement dans la deuxième étude, se révèlent pleinement satisfaisants.

Les principaux développements futurs visent à prendre en compte la dynamique du système. Cette extension permettrait de considérer les changements d'état du système au fil du temps, ce qui pourrait améliorer la précision de la localisation des défauts.

Un autre point de recherche qui nous intéresse particulièrement, est le système de diagnostic combinant différents formalismes de diagnostic et pouvant ensuite intégrer un interfaçage dans l'environnement de SolidWorks pour concrétiser l'approche proposée sur la cellule robotisée d'usinage.

Bibliographie

Bibliographie

- Abdi, Z. (2017). *Étude, diagnostic et intervention sur la scie à ruban PEHAKA ROBOTER 250 SL*. Mémoire Master, Faculté de Technologie, Université Abou Bekr Belkaïd – Tlemcen.
- Abdoulaye, D., Farid, N., Rachid, N., Bernard, K.-F., & Diakarya, B. (2019). Proportional hazard model for cutting tool recovery in machining. *Journal of Risk and Reliability*.
- Adams, D. A. (1969). *A computation model with data flow sequencing*. Computer science dept. Technical report CS-117, Stanford University.
- AFNOR. (1994). *Recueil de normes : Gérer et assurer la qualité, 2 tomes*.
- AFNOR NF X60-500. (1988). Normes nationales et documents normatifs nationaux. *Qualité - audit - lean, Management et performance, QSE*.
- Aghasaryan, A. (1998). *Formalisme HMM pour les réseaux de Pétri partiellement stochastiques : Application au diagnostic de pannes dans les systèmes répartis*. Thèse de doctorat université de Rennes I, France.
- Ahn, J., Lee, Y., & Kim, N. (2023). Federated Learning for Predictive Maintenance and Anomaly Detection Using Time Series Data Distribution Shifts in Manufacturing Processes. *Sensors*, 23(17), 7331.
- Alhelou, H. H. (2019). Fault Detection and Isolation in Power Systems Using Unknown Input Observer. *Adv. Cond. Monit. Fault Diagn. Electr. Mach*, 38-58.
- Alsyouf, I. (2009). Maintenance practices in Swedish industries : Survey results. *International Journal of Production Economics*, 121(1), 212-223.
- Angeli, C., & Chatzinikolaou, A. (2004). On-line Fault Detection Techniques for Technical Systems : A survey. *Int. J. Comput. Sci. Appl*, 1, 12-30.
- ANSI/RIA R15. (1990). *Evaluation Of Point-To-Point And Static Performance Characteristics Of Industrial Robots And Robot Systems*.
- Arena, F., Collotta, M., Luca, L., Ruggieri, M., & Termine, F. G. (2021). *Predictive Maintenance in the Automotive Sector : A Literature Review*. 27(1).
- Arlat, J., & Laprie, J.-C. (1995). Guide de la sûreté de fonctionnement. *Editions Cepaduès*.
- Barka, Y., & Benrar, H. (2018). *Étude et réalisation d'un robot cartésien 3 axes*. Mémoire de Master, Université Aboubakr Belkaïd – Tlemcen – Faculté de Technologie, Département Génie Mécanique, Spécialité : Construction Mécanique.
- Baron, P., Marek, K., & Anton, P. (2023). *Application of Troubleshooting Tools in the Monitored Production Processes*. Springer Nature.
- Basseville, M. (1988). Detecting Changes in Signals and Systems—A Survey. *International Federation of Automatic Control*, 24(3), 309-326.
- Basseville, M., & Cordier, M. O. (1996a). Surveillance et diagnostic de systèmes dynamiques : Approches complémentaires du traitement du signal et de l'intelligence artificielle. *Report INRIA n°2861*.
- Basseville, M., & Cordier, M.-O. (1996b). *Surveillance et diagnostic de systèmes dynamiques : Approches complémentaires du traitement de signal et de l'intelligence artificielle*. Rapport de recherche, RR-2861, INRIA, Institut National de Recherche en Informatique et en Automatique, Rennes, France.
- Belchior, J., Guillo, M., Courteille, Maurine, E., Leotoing, L., & Guines, D. (2013). Off-line compensation of the tool path deviations on robotic machining : Application to incremental sheet forming. *Robotics and Computer-Integrated Manufacturing*, 29(4), 58-69.
- Ben Ali, J., Fnaiech, N., Saidi, L., Chebel-Morello, B., & Fnaiech, F. (2015). Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals. *Appl. Acoust*, 89, 16-27.
- Benhouia, N.-E., Madi, B., & Benadja, M. (2017). La fiabilité prévisionnelle d'une pile à combustible de type PEMFC. *International Conference on Maintenance and Industrial Safety CIMSI*, 20(21).
- Berry, G., Couronné, P., Gonthier, G., & Banatre, J. P. (1987). Programmation synchrone des systèmes réactifs : Le langage Esterel. *TSI. Technique et science informatiques*, 6(4), 305-316.
- Biroolini, A. (1997). Quality and Reliability of Technical Systems. *Springer, Berlin*.
- Blaga, F. S., Pop, A., Hule, V., & Indre, C. I. (2021). The efficiency of modeling and simulation of manufacturing systems using Petri nets. *IOP Conference Series: Materials Science and Engineering, The Annual Session Of Scientific Papers - IMT Oradea 2021*.
- Bonnemains, T. (2009). *Étude du comportement mécanique des machines outils à structure parallèle en Usinage Grande Vitesse*. Thèse de doctorat. Université Blaise Pascal-Clermont-Ferrand II.
- Bouarioua, M. (2013a). *Une approche basée transformation de graphes pour la génération de modèles de réseaux de Petri analysables à partir de diagrammes UML*. Thèse de doctorat. Département d'Informatique Fondamentale et ses Applications, Faculté des Sciences des Nouvelles Technologies de l'Information et de la Communication, Université Constantine 2.
- Bouarioua, M. (2013b). *Une approche basée transformation de graphes pour la génération de modèles de réseaux de Petri analysables à partir de diagrammes UML*. Thèse de doctorat. Faculté des sciences des nouvelles technologies de l'information et de la communication, Université Constantine 2, Algérie.
- Bouchon-Meunier, B., & Marsala, C. (2003). Logique floue, principes, aide à la décision. *Ed. Hermes, Paris*.
- Bouissou, M., & Duitui, Y. (2004). Reliability Analysis of a Dynamic Phased Mission System. *MMR2004 congress, Santa Fe*.
- Bouzgarrou, B. C. (2001). *Conception et modélisation d'une machine-outil à architecture hybride pour l'UTGV*. Thèse de doctorat. Clermont-Ferrand 2.
- Bouزيد, A. (2017). *Stratégie de sélection des meilleures méthodes d'ébavurage*. Mémoire présenté à l'école de technologie supérieure comme exigence partielle à l'obtention de la maîtrise avec Mémoire en Génie Mécanique, École de Technologie Supérieure Université du Québec.
- Brauer, W., Rozenberg, G., & Salomaa, A. (1985). *Petri Nets : An Introduction*. Épringer-Verlag, Berlin Heidelberg, germany.
- Brézillon, P. (2003). Context-based Modeling of Operator's Practices by Contextual Graphs. *Human Centered Processes: 14th Mini Euro Conference, Luxembourg*.
- Brézillon, P., Pasquier, L., & Pomerol, J. C. (2001). Reasoning with contextual graphs. *European Journal of Research*, 136, 290-298.
- Briard, J.-M. (2003). Parachèvement en fonderie. Ebarbage. Finition. *Techniques de l'ingénieur. Matériaux métalliques, M3659*.
- Brusoni, V., Console, L., Terenziani, P., & Theseider Dupré, D. (1995). Characterizing temporal abductive diagnosis. *Proc. of International Workshop on Principles of Diagnosis*, 34-40.
- Brusoni, V., Console, L., Terenziani, P., & Theseider Dupré, D. (1997). An Efficient Algorithm for Temporal Abduction. *Lecture notes in Artificial Intelligence 1321*, 195-206.
- Bukowski, N., & Delaruelle, B. (2011). *L'utilisation des méthodes d'analyse de risques issues de l'industrie dans le domaine des barrages*. Colloque technique.

Bibliographie

- Caillaud, A., Ritou, M., Garnier, S., & Furet, B. (2009). Choix de procédés de parachèvement pour les pièces en matériaux composites= Choice of finishing process for composite parts. In : *JNC 16. AMAC*, 10.
- Capek, K. (1920). RUR—Rossum’s Universal Robots : Kolektivní drama o vstupníkomedii a 3 dejstvích. *Praha: Aventinum*.
- Cassandras, C., & Lafortune, S. (2008). Introduction to Discrete Event Systems. *2nd Edition, Springer - Verlag*.
- Charles, T. S. (2015). *Usinage de métaux durs par Jet d’Eau Abrasif*. These de Doctorat, Institut National des Sciences Appliquées de Toulouse (INSA) de Toulouse.
- Chatila, R. (2014). Robotique et simplicité : Modèles, architecture, décision et conscience. A. Berthoz, & J. Petit, *Complexité-Simplicité. Paris: Collège de France*.
- Cheng, Y.-L. (2000). Uncertainties in Fault Tree Analysis. *Tamkang Journal of Science and Engineering*, 3(1), 23-29.
- Chincholkar, A. K., & Krishnaiah Chetty, O. V. (1996). Stochastic coloured Petri nets for modelling and evaluation, and heuristic rule base for scheduling of FMS. *The International Journal of Advanced Manufacturing Technology*, 12, 339-348.
- Chiola, G., Marsan, M. A., Balbo, G., & Conte, G. (1993). Generalized stochastic Petri nets : A definition at the net level and its implications. *IEEE Transactions on software engineering*, 19(2), 89-107.
- Choi, H., Kulkarni, V. G., & Trivedi, K. S. (1993). Transient analysis of deterministic and stochastic Petri nets. In *Application and Theory of Petri Nets 1993: 14th International Conference Chicago, Illinois, USA*, 166-185.
- Choi, H., Mainkar, V., & Trivedi, K. S. (1993). Sensitivity analysis of deterministic and stochastic Petri nets. In : *Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems*, 271-276.
- Chouchane, A., & Declerck, P. (2022). Diagnosis on a sliding window for partially Observable Petri Nets. *Kybernetika*, 58(4), 479-497.
- Chu, F. (1995). *Conception des systèmes de production à l’aide des réseaux de petri : Vérification incrémentale des propriétés qualitatives*. Thèse de Doctorat, Faculté des Sciences UFR Mathématiques, Informatique, Mécanique, Université de Metz, France.
- Ciardo, G., & Lindemann, C. (1993). Analysis of deterministic and stochastic Petri nets. In : *Proceedings of 5th International Workshop on Petri Nets and Performance Models. IEEE*, 160-169.
- Clavel, R. (1991). *Conception d’un robot parallèle rapide à 4 degrés de liberté*. These de Doctorat, EPFL.
- Cocheteux, P. (2010). *Contribution à la maintenance proactive par la formalisation du processus de pronostic des performances de systèmes industriels*. Thèse de doctorat. Université Henri Poincaré-Nancy 1.
- Cocozza-Thivent, C. (1997). *Processus stochastiques et fiabilité des systèmes* (1-28). Springer Science & Business Media.
- Colin, B. (2023). Maintenance prévisionnelle des équipements critiques, embarqués sur systèmes d’armes terrestres. *e-Journal of Nondestructive Testing (eJNDT) ISSN 1435-4934*, 28(9).
- Cordeschi, R. (2002). The discovery of the artificial : Behavior, mind and machines before and beyond cybernetics. *Springer Science & Business Media*, 28.
- Cottrell, M., Gaubert, P., Eloy, C., François, D., Hallaux, G., Lacaille, J., & Verleysen, M. (2009). Fault Prediction in Aircraft Engines Using Self-Organizing Maps. In *Advances in Self-Organizing Maps, Springer. Berlin, Heidelberg*, 37-44.
- Cousino, T., Brissaud, F., & Marle, L. (2022). Estimation du vieillissement et de l’efficacité de la maintenance d’actifs industriels à partir de données censurées par intervalle. *Congrès Lambda Mu 23 «Innovations et maîtrise des risques pour un avenir durable»-23e Congrès de Maîtrise des Risques et de Sécurité de Fonctionnement, Institut pour la Maîtrise des Risques*.
- Daniel, O. (1995). *Les réseaux de petri stochastiques pour l’évaluation des attributs de la sureté de fonctionnement des systèmes manufacturiers*. Thèse de doctorat, LAG, Grenoble, France.
- David, R., & Alla, H. (1990). Autonomous and timed continuous Petri nets. *11th International Conference on Application and Theory of Petri Nets Paris France*, 367-386.
- David, R., & Alla, H. (2001). *Réseaux de Petri hybrides*. Chapitre 6. p. 119-154.
- David, R., & Alla, H. L. (2010). Discrete, continuous, and hybrid Petri nets. *Berlin : Springer*.
- David, R., & Alla, H. L. (1992b). *Du Grafset aux réseaux de Petri*. 2eme Edition revue et augmentée, Livre, France.
- David, R., & Alla, H. L. (1992a). Petri nets and Grafset : Tools for modelling discrete event systems. *Prentice-Hall, Inc*.
- Davim, J. P., & Jain, V. K. (2008). Advanced (non-traditional) machining processes. *Machining: Fundamentals and recent advances*, 299-327.
- De Micheli, G., & Sangiovanni-Vincentelli, A. (1981). *Numerical properties of algorithms for the timing analysis of MOS VLSI circuits*. Electronics Research Laboratory, College of Engineering, University of California.
- Demongodin, I. (1994). *Les réseaux de Petri Lots : Modélisation des systèmes de production à haute cadence en régime transitoire*. These de Doctorat, Université de Montpellier.
- Demongodin, I. (2001). Generalised batches Petri net : Hybrid model for high speed systems with variable delays. *Discrete Event Dynamic Systems*, 11(1), 137-162.
- Demri, A. (2009). *Contribution à l’évaluation de la fiabilité d’un système mécatronique par modélisation fonctionnelle et dysfonctionnelle*. Thèse de doctorat. Université d’Angers.
- Derbazi, D. (2023). *Contrôle et diagnostic des risques industriels des installations et équipements électromécaniques Cas de la cimenterie de Elmalabiod Tébessa*. Thèse de doctorat. Université de Tébessa.
- Des Métaux, D. (2017). Le tronçonnage à la meule. —Méthodes et outillages modernes. *Prat. Ind. Méc*, 16(2), 50-52.
- Devol, J. G. C. (1961). Programmed article transfer. *U.S. Patent and Trademark Office*, 2,988,237.
- Diaz, M. (2001). Les réseaux de Petri : Modèles fondamentaux. *Hermès science publications*.
- Diaz, M., & Alla, P. (1985). A Formal Definition of a Petri Net Synchronized with a Finite Set of Events. *Information Processing Letters*, 21(4), 23-25.
- Djellal, A., & Lakel, R. (2017). Petri Net Model of Pursuit-Evasion Game in a Graph. *International Journal of Control and Automation*.
- Doche, M.-L., & Hihn, J.-Y. (2021). Électropolissage-Procédé pour le parachèvement des pièces de fabrication additive métalliques. *techniques ingénieur*.
- DONCHI, K. F. K., Ftatsi Mbetmi, G., FOTSA-MBOGNE, D. J., & DJEUMAKO, B. (2022). *Modélisation et estimation de la fiabilité dynamique d’un évacuateur de crues par réseaux de Petri stochastiques*.
- Dubuisson, B. (2001). *Diagnostic, intelligence artificielle et reconnaissance des formes*. Librairie Lavoisier.
- Dugan, J. B. (1984). *Extended stochastic petri nets : Applications and analysis (modeling, reliability, performance)*. Duke University.
- Dumas, C. (2011). *Développement de méthodes robotisées pour le parachèvement de pièces métalliques et composites*. These de Doctorat, Université de Nantes.
- Dutheillet, C., & Haddad, S. (1989). Aggregation of States in Colored Stochastic Petri Nets : Application to a Multiprocessor Architecture. In *PNPM*, 40-49.
- Dutuit, Y., Signoret, J. P., Folleau, C., & Thomas, P. (2018). *Détermination et calcul de la fréquence d’occurrence d’une coupe minimale – Un bref état de l’art –*. 21e Congrès de Maîtrise des Risques et Sécurité de Fonctionnement $\lambda\mu 21$, Reims, France.

Bibliographie

- El Hadj Yagoub, K. (2023). *Analyse des risques d'un four industriel etude sur le four H-1501 du complexe GPIZ ARZEW-ORAN*. Thèse de doctorat. Université Kasdi Merbah Ouargla.
- El Hami, A., & Eid, M. (2023). Modélisation par la fiabilité de la performance d'un système : Maintenabilité, événements séquentiels, systèmes multi-états. *ISTE Group, 15*.
- Ernotte, B., Boeraeve, P., Dehard, J., Sener, J., Bortolotti, E., & Zielemann, P. (2005). Assemblage par perçage dans la construction tubulaire. *Cahier de l'APK*.
- Fabre, E., Benveniste, A., Jard, C., & Smith, M. (2001). Diagnosis of distributed discrete event systems, a net unfolding approach. *Publication Interne IRISA*.
- Florin, G. (1985). *Réseaux de Petri stochastiques, théorie et techniques de calcul*. Thèse de doctorat. Thèse d'état de l'Université de Pierre et Marie Curie, Paris VI.
- Frank, P. M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy : A survey and some new results. *Automatica, 26*(3), 459-474.
- Frankel, E. G. (1988). Systems reliability and risk analysis. *Kluwer Academic, Dordrecht, Netherlands*.
- Gao, D., Wu, C., Zhang, B., & Ma, X. (2010). Signed Directed Graph and Qualitative Trend Analysis Based Fault Diagnosis in Chemical Industry. *Chin. J. Chem. Eng., 18*, 265-276.
- Gao, Z., Cecati, C., & Ding, S. X. (2015). A survey of fault diagnosis and fault-tolerant techniques- Part I: fault diagnosis With model-based and signal-based approaches. *IEEE Trans. Ind. Electron., 62*, 3757-3767.
- Garin, H. (1994). *AMDEC/AMDE/AEEL l'essentiel de la méthode*. Association française de normalisation, AFNOR, Paris, France.
- Gertler, J. (2017). *Fault Detection and Diagnosis in Engineering Systems (CRC Press)*.
- Gharahasanlou, N., Mokhtarei, A., Khodayarei, A., & Ataei, M. (2014). Fault tree analysis of failure cause of crushing plant and mixing bed hall at Khoy cement factory in Iran. *Case Stud. Eng. Fail. Anal., 2*, 33-38.
- Ghenai, A. (2017). *Évaluation de la fiabilité des systèmes embarqués dès la phase de conception par réseaux de Petri temporels étendus*. Thèse de Doctorat. Département Technologie des Logiciels et Systèmes d'Information, Faculté des Nouvelles Technologies de l'Information et de la Communication, Université Constantine 2.
- Ghostine, R. (2008). *Influence des fautes transitoires sur la fiabilité d'un système commandé en réseau*. Thèse de doctorat, Institut National Polytechnique de Lorraine, Spécialité Automatique, Traitement du signal et Génie Informatique, France.
- Girault, C., & Pradat-Peyre, J.-F. (2001). Les réseaux de Petri de haut-niveau. *Les réseaux de Petri: Modeles fondamentaux, number ISBN, 2-7462*.
- Girault, F. (1997). *Formalisation en logique linéaire du fonctionnement des réseaux de Petri*. Thèse de doctorat. Université Paul Sabatier-Toulouse III.
- Girault, F., Pradin-Chézalviel, B., Kunzle, L. A., & Valette, R. (1995). Linear logic as a tool for reasoning on a Petri net model. In: *Proceedings 1995 INRIA/IEEE Symposium on Emerging Technologies and Factory Automation. ETFA'95, 1*, 49-57.
- Giua, A., & Seatzu, C. (2007). A systems theory view of Petri nets. In *Advances in control theory and applications. Springer, Berlin, Heidelberg*, 99-127.
- Gognau, D. (2003). *Contraintes résiduelles introduites par des opérations de parachèvement de l'état de surface sur un acier de construction*. Thèse de doctorat. Physique. Sciences des matériaux.
- Gogu, G. (2008). *Structural synthesis of parallel robots*. Dordrecht: Springer Netherlands.
- Graetz, G., & Michaels, G. (2015). Robots at work. *CEPR Discussion Paper, DP10477*.
- Grosclaude, I. (2001a). *Diagnostic abductif temporel—Scénarios de pannes, modèles causaux et traitement de l'information*. Thèse de Doctorat, Université de Rennes 1.
- Grosclaude, I., Cordier, M. O., & Quiniou, R. (2001b). Causal interaction : From a High Level Representation to a Operational Event-Based Representation. *International Joint Conference on Artificial Intelligence, IJCAI 2001*.
- Grosclaude, I., & Quiniou, R. (2000). Dealing with Interacting Faults in Temporal Abductive Diagnosis. *Proc. of the 8th International Workshop on Principles of Diagnosis*.
- Guillaume, A., Vrain, C., & Elloumi, W. (2023). Vers un outil d'évaluation comparative pour la maintenance prédictive : Comment comparer différentes approches? *EGC 2023-Atelier GATS*.
- Guillot, A., & Meyer, J.-A. (2008). La bionique : Quand la science imite la nature. *Dunod*.
- Guillou, G. (2010). *Architecture multi-agents pour le pilotage automatique des voiliers de compétition et extensions algébriques des réseaux de Petri*. Thèse de doctorat. Université de Bretagne occidentale-Brest.
- Guiot, A. (2012). *Modélisation et simulation du procédé de prépolissage automatique sur centre d'usinage 5 axes*. Thèse de doctorat. École normale supérieure de Cachan-ENS Cachan.
- Guo, Y., Wang, J., Chen, H., Li, G., Huang, R., Yuan, Y., Ahmad, T., & Sun, S. (2019). An expert rule-based fault diagnosis strategy for variable refrigerant flow air conditioning systems. *Appl. Therm. Eng., 149*, 1223-1235.
- Haas, P. J. (2006). Stochastic Petri Nets : Modeling, Stability, Simulation. *Springer Science & Business Media*.
- Haas, P. J., & Shedler, G. S. (1989a). Stochastic Petri net representation of discrete event simulations. *IEEE Transactions on software engineering, 15*(4), 381-393.
- Haas, P. J., & Shedler, G. S. (1989b). Stochastic Petri nets with timed and immediate transitions. *Stochastic Models, 5*(4), 563-600.
- Habchi, G. (2018). *Adonis : Modéliser la Fiabilité des Composants et des Systèmes*. Congrès National de la Recherche des IUT.
- Haddad, M. (2013). *Étude de l'impact des défauts d'usinage des structures composites par détournement sur leur comportement mécanique*. Thèse de Doctorat, Université Toulouse III - Paul Sabatier.
- Haddad, S. (1987). *Une catégorie régulière de réseau de petri de haut niveau : Définition, propriétés et réductions, application à la validation de systèmes distribués*. Thèse de doctorat. Paris 6.
- Hadj-Mabrouk, H. (2006, octobre 10). *Méthode d'Analyse Préliminaire des Risques dans les transports ferroviaires*. Congrès lambda mu 15, Palais des Congrès de Lille, France.
- Heng, J. (2002). *Pratique de la Maintenance préventive*. édition Dunod.
- Holt, A. W., Commoner, F., & Massachusetts Computer Associates INC Wakefield. (1970). *Research in Machine-Independent Software Programming. Task Area II. Events and Conditions : An Approach to the Description and Analysis of Dynamic Systems*.
- Hoq, K. M. G. (2014). Information overload : Causes, consequences and remedies-A study. *Philosophy and progress, 49-68*.
- Hua, J., Lu, L., Ouyang, M., Li, J., & Xu, L. (2011). Proton exchange membrane fuel cell system diagnosis based on the signed directed graph method. *Journal of Power Sources, 196*(14), 5881-5888.
- Hura, G. S., & Atwood, J. W. (1989). The Use of Petri Nets to Analyze Coherent Fault Trees. *IEEE Transactions on Reliability, 37*(5), 469-474.
- Hwang, I., Kim, S., Kim, Y., & Seah, C. E. (2010). A Survey of Fault Detection, Isolation, and R configuration Methods. *Fault Detect. Isol. Reconfiguration Methods, 18*, 636-653.

Bibliographie

- IEC. (1990). *IEC 60050-191, International Electrotechnical Commission – Chapter 191, 1st edition*. Geneva: International Electrotechnical Commission.
- Ikhlef, B. (2009). *Contribution à l'Etude de la Supervision Industrielle Automatique dans un Environnement SCADA*. Mémoire Magister, Département Automatisation des procédés industriels et Electrification, Option : Automatisation des procédés industriels et traitement du signal, Faculté des Hydrocarbures et de la Chimie, Université M'Hamed Bougara de Boumerdes, Algérie.
- Isermann, R. (1984). Process fault detection based on modeling and estimation methods—A survey. *Automatica*, 20(4), 387-404.
- Isermann, R., & Ballé, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5), 709-719.
- ISO 8373. (2021). Robots et composants robotiques. *Edition 3*.
- ISO 9283. (1998). *Manipulating industrial robots—Performance criteria and related test methods*.
- Jemmali, M. (2021). *Maintenance prédictive pour une turbine de puissance en utilisant des algorithmes d'apprentissage machine*. Thèse de doctorat. Université du Québec. École de technologie supérieure.
- Jensen, K. (1986). Coloured petri nets. In : *Petri Nets: Central Models and Their Properties: Advances in Petri Nets 1986, Part I Proceedings of an Advanced Course Bad Honnef*, 248-299.
- Jimenez, J. J. M., Schwartz, S., Vingerhoeds, R., Grabot, B., & Salaün, M. (2020). Towards multi-model approaches to predictive maintenance : A systematic literature survey on diagnostics and prognostics. *Journal of manufacturing systems*, 56, 539-557.
- Jubin, J. (2019). *Que nous apprend la tache de table de verite probabiliste sur le raisonnement conditionnel ?* Thèse de doctorat. Faculté de psychologie et des sciences de l'éducation de l'Université de Genève.
- Kaddachi, R. (2021). *Contrôle de la production et de la maintenance des systèmes industriels non fiables pour les produits périssables*. Thèse de doctorat. École de technologie supérieure.
- Karp, R. M., & Miller, R. E. (1969). Parallel Program Schemata. *Journal of Computer and System Sciences*, 3(2), 147-195.
- Khalil, W., & Dombre, E. (1999). Modélisation, identification et commande des robots. *Hermes Science Publications*.
- Kharouati, A. (2021). *Contribution à l'étude de la sûreté de fonctionnement des systèmes instrumentés intelligents*. Thèse de doctorat. Département d'Electronique, Faculté des Sciences de l'Ingénieur. Université Badji Mokhtar – Annaba.
- Kurban, T., & Beşdok, E. (2009). A Comparison of RBF Neural Network Training Algorithms for Inertial Sensor Based Terrain Classification. *Sensors*, 9(8).
- Labadi, K. (2005). *Contribution à la modélisation et à l'évaluation de performances des systèmes logistiques à l'aide d'un nouveau modèle de réseaux de Petri stochastiques*. Thèse de doctorat. Université de Technologie de Troyes.
- Ladet, P. (1989). Réseaux de petri. *Techniques de l'Ingénieur*, 7252, 1-17.
- Lajmi, F., Ghabi, J., & Dhoubi, H. (2020). Applying Interval Fuzzy Petri Net to failure analysis. *International Journal of Engineering and Technology*, 11(1).
- Latrous, M. O. (2021). *Modèle d'intégration maintenance conditionnelle, prévisionnelle et maîtrise statistique des procédés*. Thèse de doctorat. Spécialité : Génie Industriel, Université de Technologie de Compiègne.
- Lee, C. S. G., & Wang, J.-S. (2000). Self-Adaptive Neuro-Fuzzy Systems : Structure and learning, IEE/RSJ. *International Conference on Intelligent Robots and Systems*.
- Lee, C. S. G., & Wang, J.-S. (2001). Efficient Neuro-Fuzzy Control Systems for Autonomous Underwater Vehicle Control. *IEEE International Conference on Robotics and Automation, Seoul, Corée*.
- Lee, W. S., Grosh, D. L., Tillman, F. A., & Lie, C. H. (1985). Fault Tree Analysis, Methods, and Applications-A Review. *IEEE Transactions on Reliability*, 34(3), 194-203.
- Lefebvre, D. (2000). *Contribution à la modélisation des systèmes dynamiques à événements discrets pour la commande et la surveillance*. Habilitation à Diriger des Recherches, Université de Franche-Comté/ IUT Belfort.
- Lesobre, R. (2015a). *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*. Thèse de doctorat. Spécialité : Automatique - Productique, Université Grenoble Alpes.
- Lesobre, R. (2015b). *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*. Thèse de doctorat. Université Grenoble Alpes.
- Lindemann, C. (1998). Performance modelling with deterministic and stochastic Petri nets. *ACM sigmetrics performance evaluation review*, 26(2), 3.
- Liu, R., Yang, B., Zio, E., & Chen, X. (2018). Artificial intelligence for fault diagnosis of rotating machinery : A review. *Mech. Syst. Signal Process*, 108, 33-47.
- Liu, Y. J., Meng, Q. H., Zeng, M., & Ma, S. G. (2016). Fault diagnosis method based on probability extended SDG and fault index. In *2016 12th World Congress on Intelligent Control and Automation (WCICA), IEEE, Guilin, China*, 2868-2873.
- Lung, B. (2019, mai 23). De la maintenance prédictive à la maintenance prescriptive : Une évolution nécessaire pour l'industrie du futur. *Conference on Complexity Analysis of Industrial Systems and Advanced Modeling (CAISAM 2019)*.
- Mahdaoui, R. (2013). *Contribution à la surveillance dynamique des systèmes de production évolutifs par les systèmes Neuro-Flous Temporels*. Thèse de doctorat, Université de Batna 2, Département de Génie Industriel, Algérie.
- Marsan, M. A., Balbo, G., Conte, G., Donatelli, S., & Franceschinis, G. (1998). Modelling with generalized stochastic Petri nets. *ACM SIGMETRICS performance evaluation review*, 26(2), 2.
- Marsan, M. A., & Chiola, G. (1986). On Petri Nets with Deterministic and Exponential Transition Firing times. In *Proc. 7th European Workshop on Application and Theory of Petri Nets*, 151-165.
- Marsan, M. A., & Chiola, G. (1987). On Petri nets with deterministic and exponentially distributed firing times. In *Advances in Petri Nets 1987, Springer Berlin Heidelberg*, 132-145.
- Marsan, M. A., Conte, G., & Balbo, G. (1984). A Class of Generalized Stochastic Petri Nets for the Performance Analysis of Multiprocessor Systems. *ACM Transactions on Computer Systems (TOCS)*, 2(2), 93-122.
- Mazouni, M. H. (2008). *Pour une meilleure approche du management des risques : De la modélisation ontologique du processus accidentel au système interactif d'aide à la décision*. Thèse de doctorat. Institut National Polytechnique de Lorraine-INPL.
- Medjoudj, M. (2006). *Contribution à l'analyse des systèmes pilotés par calculateurs : Extraction de scénarios redoutés et vérification de contraintes temporelles*. Thèse de doctorat. Université Paul Sabatier-Toulouse III.
- Mehar, Z., Noureddine, R., & Noureddine, F. (2022). Diagnostic by Fault Tree and Petri Nets of a Robotic Machining Cell. *Algerian Journal of Research and Technology (AJRT)*, 6(1), 10-18.
- Mehar, Z., Rachid, N., & Farid, N. (2023). PETRI NET MODELING OF A MACHINING ROBOT CELL. *UPB Scientific Bulletin, Series D: Mechanical Engineering*, 85(4), 89-108.

Bibliographie

- Mejri, S. (2016). *Identification et modélisation du comportement dynamique des robots d'usinage*. Thèse de doctorat. Université Blaise Pascal-Clermont-Ferrand II.
- Mellouli, N., & Bouchon-Meunier, B. (2000). Fuzzy Approaches to Abductive Inference. *Conference on Non-Monotonic Reasoning, Berckenridge, Colorado*.
- Meraghni, S. (2021). *La maintenance prédictive dans les usines intelligentes*. Thèse de doctorat. Université de mohamed kheider biskra.
- Merlet, J.-P. (1996). *Parallel Robots*. Springer. Paris. France.
- Merlin, P. M. (1974). *A Study of the Recoverability of Computing Systems*. Thèse de doctorat. Department of Information and Computer Science, University of California, Irvine, CA.
- M'halla, A. (2023). Contribution à la Surveillance d'un système automatisé de production par des Réseaux de Pétri Hybrides. *Revue des sciences et sciences de l'ingénieur (ASJP)*, 9(1), 1-9.
- Miessner, B. F. (1912). The wirelessly directed torpedo. Some new experiments in an old field. *Sci. Ameri*.
- Mihalache, A. G. (2007a). *Modélisation et évaluation de la fiabilité des systèmes mécatroniques : Application sur système embarqué*. Thèse de Doctorat, Institut des Sciences et Techniques de l'Ingénieur d'Angers, Université d'Angers.
- Mihalache, A. G. (2007b). *Modélisation et évaluation de la fiabilité des systèmes mécatroniques : Application sur système embarqué*. Thèse de doctorat. Université d'Angers.
- Minca, E., FILIP, F., & ZEMOURI, R. (2008). Approche de la Surveillance des Systèmes par Réseaux de Petri Synchronisés Flous. *Proceedings of the Romanian Academy, Series A*, 9(2).
- Minca, E., Racoceanu, D., & Zerhouni, N. (2003). *Approche temporelle floue pour la surveillance*.
- Molloy, M. K. (1982). Performance analysis using stochastic Petri nets. *IEEE Transactions on computers*, 100(9), 913-917.
- Moncelet, G. (1998). *Application des RdP à l'évaluation de la sûreté de fonctionnement des systèmes mécatroniques du monde automobile*. Thèse de doctorat, LAAS, Toulouse, France.
- Monchy, F., & Vernier, J.-P. (2000). Maintenance. *Méthodes et organisation, Dunod*.
- Morandea, A., Leroy, R., Bouchou, A., & Bonhoure, D. (2011). Usinage des composites renforcés en fibres de carbone : Stratégie de surfacage pour limiter les efforts de coupe. *17èmes Journées Nationales sur les Composites (JNC17)*, 126.
- Moraru, V. M. (2005). *Réseaux de Petri*. Institut de la Francophonie pour l'Informatique.
- Morel, J., Sourisse, C., & Merlaude, C. (1992). *La sûreté des machines et installations automatisées*. Sadave-CITEF, Paris, France.
- Moshou, D., Kateris, D., Sawalhi, N., Loutridis, S., & Gravalos, I. (2010). Fault Severity Estimation in Rotating Mechanical Systems Using Feature Based Fusion and SelfOrganizing Maps. *In Artificial Neural Networks – ICANN 2010, Springer. Berlin, Heidelberg*, 410-413.
- Motrani, A. (2022). *Contribution à la mise en œuvre du Prognostic and Health Management (PHM) industriel*. Thèse de doctorat. Institut de Maintenance et de sécurité Industrielle, Université d'Oran 2.
- Moulières-Seban, T. (2017). *Conception de systèmes cobotiques industriels : Approche cognitive : Application à la production pyrotechnique au sein d'Ariane Group*. Thèse de doctorat, École Doctorale Sciences Physiques et de l'Ingénieur Automatique, Productique, Signal et Image, Ingénierie Cognitive Spécialité Ingénierie Cognitive, Université de Bordeaux, France.
- Moussavi, S. (2016). *Analyse de la stabilité de la coupe d'un procédé d'usinage robotisé*. Thèse de doctorat. Université Blaise Pascal-Clermont-Ferrand II.
- Moussavi, S., Gagnol, V., Bouzgarrou, B. C., & Ray, P. (2014). Modélisation dynamique et prédiction de la stabilité d'un robot d'usinage. *Conférence: MUGV 2014 At : Ifma, Clermont Ferrand*.
- Msaaf, M., & Belmajdoub, F. (2015). L'application des réseaux de neurone de type " feedforward " dans le diagnostic statique. *In Xème Conférence Internationale : Conception et Production Intégrées. Tanger, Morocco*.
- Munaweera, I., Muthukumarana, S., Gillis, D. M., Watkinson, D. A., Charles, C., & Enders, E. C. (2021). Assessing movement patterns using Bayesian state space models on Lake Winnipeg walleye. *Canadian Journal of Fisheries and Aquatic Sciences*, 78(10), 1407-1421.
- Murata, T. (1989). Petri nets : Properties, analysis and applications. *Proc. IEEE* 77, 541-580.
- Natkin, S. O. (1980). *Les Réseaux de petri stochastiques et leur application de l'évaluation des systèmes informatiques*. Thèse de doctorat Ingénieur. CNAM Paris.
- Nauck, D., & Kruse, R. (1998). A neuro-fuzzy approach to obtain interpretable fuzzy systems for function approximation. *IEEE International Conference on Fuzzy Systems, Anchorage*, 2, 1106-1111.
- NF EN 13306. (2018, janvier). *Maintenance—Terminology of the maintenance*. European standard.
- Noureddine, F. (1996). Conception d'un outil logiciel graphique pour l'analyse comportementale d'un robot en mode dégradé. *Revue Internationale de CFAO et d'informatique graphique*, 11(1-2), 199-214.
- Noureddine, R. (2008). *Implémentation de la maintenance prédictive dans les systèmes de production*. Thèse de Doctorat. Université des Sciences et de la Technologie d'Oran - Mohamed Boudiaf.
- Noureddine, R., Benamar, A., & Noureddine, F. (2005). Intégration d'une technique de maintenance prédictive de l'outil de coupe dans une cellule d'usinage robotisée. *Colloque International Francophone sur les Performances et Nouvelles Technologies en Maintenance (PENTOM 2005), Actes PENTOM 2005, Presses Universitaires de Valenciennes, Marrakech, Maroc, N°6*, 133-142.
- Noureddine, R., Benamar, A., & Noureddine, F. (2008). Maintenance prédictive de l'outil de coupe en usinage. *Mechanics & Industry*, 9(3), 205-212.
- Noyes, D., & Pérès, F. (2007). Analyse des systèmes-Sûreté de Fonctionnement. *Techniques de l'ingénieur*.
- Oeing, J., Holtermann, T., Welscher, W., Severins, C., Vogel, M., & Kockmann, N. (2023). preHAZOP: Graph-Based Safety Analysis for Early Integration into Automated Engineering Workflows. *Chemie Ingenieur Technik*.
- Omrane, A., Kammoun, A., & Seaman, C. (2018). Entrepreneurial burnout : Causes, consequences and way out. *FIIB Business Review*, 7(1), 28-42.
- Orlianges, J.-C., El Moustakime, Y., Crunteanu Stanescu, A., Carrizales Juarez, R., & Allegret, O. (2024). Retour vers le perceptron-fabrication d'un neurone synthétique à base de composants électroniques analogiques simples. *Les journées de l'interdisciplinarité*.
- Ould Abdeslam, D. (2002). *Utilisation des Réseaux Neuro-Flous pour le Pronostic et le Diagnostic. Application à la classification*. Université de FrancheComté, Besançon, France.
- Paccot, F. (2009). *Contribution à la commande dynamique référencée capteurs de robots parallèles*. These de Doctorat, Université Blaise Pascal - Clermont-Ferrand II.
- Pagès, A., & Gondran, M. (1980). Fiabilité des systèmes. *Eyrolles, Paris*.
- Palade, V., Patton, R. J., Uppal, F. J., Quevedo, J., & Daley, S. (2002). Fault diagnosis of an industrial gas turbine using neuro-fuzzy methods. *15th IFAC World Congress, Barcelona, Espagne*, 2477-2482.
- Palluat, N. (2006). *Méthodologie de surveillance dynamique à l'aide des réseaux neuro-flous temporels*. Thèse de Doctorat, Université de Franche-Comté.

Bibliographie

- Palluat, N., Racoceanu, D., & Zerhouni, N. (2004). Diagnosis aid system using a neuro-fuzzy Approach. *Advances in Maintenance and Modeling, Simulation and Intelligent Monitoring of Degradation*.
- Palluat, N., Racoceanu, D., & Zerhouni, N. (2005). A neuro-fuzzy monitoring system. Application to flexible production systems. *Computers in Industry, Preprint submitted to Elsevier Science*.
- Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E., & Loncarski, J. (2018). Machine Learning approach for Predictive Maintenance in Industry 4.0. *14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 1-6.
- Pasquier, L. (2002). *Modélisation de raisonnement tenu en contexte : Application à la gestion d'incidents sur une ligne de métro*. Thèse de l'Université Pierre et Marie Curie.
- Petri, C. A. (1962). *Kommunikation mit automaten*. Thèse de doctorat. University of Bonn, West Germany.
- Petri, C. A., & Reisig, W. (1985). Interpreted Petri Nets : A Model for Concurrent Systems. *Acta Informatica*.
- Pham, H., & Wang, H. (1996). Imperfect maintenance. *European journal of operational research*, 94(3), 425-438.
- Pierre, M. J. (1997). Les Robots Paralleles. In *[SIJ] : Ed*.
- Potes Ruiz, P. A. (2014). *Génération de connaissances à l'aide du retour d'expérience : Application à la maintenance industrielle*. Thèse de doctorat. Institut National Polytechnique de Toulouse (INP Toulouse), Université de Toulouse.
- Purba, J. H. (2014). A fuzzy-based reliability approach to evaluate basic events of fault tree analysis for nuclear power plant probabilistic safety assessment. *Ann. Nucl. Energy* 70, 21-29.
- Quenec'hdu, Y., & Zaytoon, J. (2001). Sur la modélisation des systèmes hybrides, Systèmes Dynamiques Hybrides. *Traité Systèmes automatisés, Information commande et communication, Hermes, Paris*, 87-92.
- Rachidi, A., Talbi, A., & Khatory, A. (2013). The new forms of industrial maintenance : Which impact on the performance of the industrial companies? (Case study). *International Journal of Engineering and Advanced Technology (IJEAT), Exploring Innovation*, 2249-8958.
- Racoceanu, D. (2006). *Contribution à la Surveillance des Systèmes de Production en Utilisant l'Intelligence Artificielle*. Thèse de doctorat, Université de Franche-Comté, France.
- Racoceanu, D., & Zerhouni, N. (2005). Monitoring Approach Using Recurrent Radial Basis Function Neural Networks and Neuro-Fuzzy Systems. *International Conference on Neural Networks and Brain, 2005. ICNN&B'05*. 2.
- Rahamani, K., Arabshahi, P., Yan, T.-Y., Pham, T., & Finley, S. G. (1998). An Intelligent Fault Detection and Isolation Architecture For Antenna Arrays. *TDA Progress Report*, 42-132.
- Ramchandani, C. (1974). *Analysis of asynchronous concurrent systems by timed Petri nets*. Thèse de doctorat. Massachusetts Institute of Technology, Cambridge, MA, Project MAC Report MAC-TR-120.
- Rasovska, I., Chebel-Morello, B., & Zerhouni, N. (2007). *Classification des différentes architectures en maintenance*.
- Reisig, W. (1991). Petri nets and algebraic specifications. *Theoretical Computer Science*, 80(1), 1-34.
- Robin, V. (2007). *Contribution à la mise en oeuvre et l'optimisation d'une cellule robotisée : Application au parachèvement de pièces de fonderie*. Thèse de doctorat. Université Blaise Pascal-Clermont-Ferrand II.
- Romain, B. (2009). *Analyses de sûreté de fonctionnement multi-systèmes*. Thèse de doctorat. Université Sciences et Technologies, Bordeaux I.
- Ruijters, E., Budde, C. E., Nakhaee, M. C., Stoelinga, M., Bucur, D., Hiemstra, D., & Schivo, S. (2019). *FFORT: A benchmark suite for fault tree analysis*. The 29th European Safety and Reliability Conference, Singapore.
- Ruijters, E., & Stoelinga, M. (2015). Fault tree analysis : A survey of the state-of-the-art in modeling, analysis and tools. *Comput. Sci. Rev.*, 15-16, 29-62.
- Sallak, M. (2007). *Évaluation de paramètres de sûreté de fonctionnement en présence d'incertitudes et aide à la conception : Application aux Systèmes Instrumentés de Sécurité*. Thèse de doctorat. Institut National Polytechnique de Lorraine-INPL.
- Sallak, M., Aubry, J.-F., & Simon, C. (2005). *Impact de l'imprécision des taux de défaut-lances dans les Arbres de Défaillances et facteurs d'importance flous*. JDMACS, Lyon, France.
- Saludes, S., Corrales, A., Miguel, L. J., & Perán, J. R. (2003). A SOM and Expert System Based Scheme for Fault Detection and Isolation in a Hydroelectric Power Station. *IFAC Proc*, 36, 999-1004.
- Schoenig, R., Aubry, J.-F., Cambois, T., & Hutinet. (2006). An aggregation method of Markov graphs for the reliability analysis of hybrid systems. *Reliability Engineering & System Safety*, 91(2), 137-148.
- Schoose, C. (2016). *De l'ébarbage traditionnel à l'usage d'un cobot : Analyse du geste professionnel dans un objectif de prévention des troubles musculo-squelettiques*. These de Doctorat, Université Grenoble Alpes.
- Senoussi, H. (2015). *Sélection de Données pour l'Apprentissage des Réseaux de Neurones, Arbres de Décision et les k-Plus Proches Voisins : Application en Diagnostic de Pannes*. Thèse de doctorat. Département d'Électronique, Faculté de Génie Électrique, Université des Sciences et de la Technologie d'Oran Mohamed BOUDIAF.
- Skliros, C., Esperon Miguez, M., Fakhre, A., & Jennions, I. K. (2019). A review of model based and data driven methods targeting hardware systems diagnostics. *Diagnostyka*, 20(1).
- Soualhi, A., Elyousfi, B., Hawwari, Y., Medjaher, K., Clerc, G., Hubert, R., & Guillet, F. (2019). PHM Survey : Implementation of Diagnostic Methods for Monitoring Industrial Systems. *International Journal of Prognostics and Health Management*.
- Srivastava, N. P., Srivastava, R. K., & Vashishtha, P. (2014). Fault detection and isolation (FDI) via neural networks. *Journal of Engineering Research and Applications*, 4(1), 81-86.
- Stewart, D. (1965). A platform with six degrees of freedom. *Proceedings of the institution of mechanical engineers*, 180(1), 371-386.
- Talebberrouane, M., Khan, F., & Kamil, M. Z. (2019). Dynamic RAMS Analysis Using Advanced Probabilistic Approach. *Dynamic RAMS Analysis Using Advanced Probabilistic Approach, Chemical Engineering Transactions*, 77, 241-246.
- Talebberrouane, M., Khan, F., & Lounis, Z. (2016). Availability Analysis of Safety Critical Systems Using Advanced Fault Tree and Stochastic Petri Net Formalisms. *Journal of Loss Prevention in the Process Industries* 44.
- Talebberrouane, M., & Lounis, Z. (2016). Safety assessment of flare system by fault tree analysis. *Journal of Chemical Technology & Metallurgy*, 51(2), 229-234.
- Thomas, R. K. (2005). *Robotics and Automation Handbook*.
- Toudic, J., M. (1982). *Algorithmes d'algèbre linéaire pour l'analyse structurelle des réseaux de Pétri*.
- Tromp, L. (2000). *Surveillance et diagnostic de systèmes industriels complexes : Une approche hybride numérique/symbolique*. Thèse de doctorat, Université de Rennes 1, France.
- Upadhyay, V., Jain, P. K., & Mehta, N. K. (2013). In-process prediction of surface roughness in turning of ti-6al-4v alloy using cutting parameters and vibration signals. *Measurement*, 1 (46), 154-160.
- Uppal, F. J., Patton, R. J., & Palade, V. (2002). Neuro-fuzzy based Fault Diagnosis applied to an Electro-Pneumatic Valve. *15ème IFAC World Congress on Automatic Control, Barcelone, Espagne*, 2477-2482.

Bibliographie

- Urmos, A., Farkas, M., Kóczy, L., & Nemcsics, A. (2013). Quantum structure classification by Kohonen SelfOrganizing Map and by Fuzzy C-Means algorithm. *In Hungarian Consortium*, 313-318.
- Vadeboncoeur, R. (2022). *Démarche d'implantation progressive de la maintenance 4.0*. Thèse de doctorat. École de technologie supérieure.
- Valette, R. (2002). *Moyens de la sûreté de fonctionnement*. Thèse de doctorat, LAAS, Toulouse, France.
- Valette, R., & Atabakhche, H. (1987). Petri nets for sequence constraint propagation in knowledge based approaches. *In: Concurrency and Nets: Advances in Petri Nets. Heidelberg: Springer Berlin*, 555-569.
- Valette, R., & Künzle, L. A. (1994). *Réseaux de Petri pour la détection et le diagnostic*.
- Vathoopan, M., Johny, M., Zoitl, A., & Knoll, A. (2018). Modular fault ascription and corrective maintenance using a digital twin. *IFAC-PapersOnLine*, 1041-1046.
- Vaxevanidis, N. M., Kechagias, J., Fountas, N. A., & Manolagos, D. E. (2015). Evaluation of machinability in turning of engineering alloys by applying artificial neural networks. *The Open Construction & Building Technology Journal*, 8(1), 389-399.
- Villemeur, A. (1988a). Sûreté de fonctionnement des systèmes industriels. *Collection de la Direction des Etudes et Recherches d'Electricité de France*, 67.
- Villemeur, A. (1988b). *Sûreté de Fonctionnement des Systèmes Industriels Fiabilité – Facteurs Humains Informatisation*. Eyrolles, Paris, France.
- Vrignat, P. (2010a). *Génération d'indicateurs de maintenance par une approche semi-paramétrique et par une approche markovienne*. Thèse de doctorat. Université d'Orléans.
- Vrignat, P. (2010b). *Génération d'indicateurs de maintenance par une approche semi-paramétrique et par une approche markovienne*. Thèse de doctorat. École Doctorale Sciences et Technologies, Université d'Orléans.
- Wang, H. (2002). A survey of maintenance policies of deteriorating systems. *European journal of operational research*, 139(3), 469-489.
- Wang, J. (1998). *Timed Petri Nets : Theory and Application*. Kluwer Academic Publishers.
- Wang, J.-S., & Lee, C. S. G. (2002). Self-Adaptative Recurrent Neuro-Fuzzy Control for an Autonomous Underwater Vehicle. *IEEE International Conference on Robotics & Automation, Washington, 19(2)*, 283-295.
- Weber, P., & Simon, C. (2015). *Méthode de modélisation par Réseaux Bayésien de la fiabilité d'un système multi-états*. 11ème Congrès International Pluridisciplinaire en Qualité, Sûreté de Fonctionnement et Développement Durable, QUALITA 2015.
- Willsky, A. S. (1976). A Survey of Design Methods for Failure Detection in Dynamic Systems. *Pergamon Press, 1976, Printed in Great Britain*, 12, 601-611.
- Xavior, A., & Adithan, M. (2009). Determining the influence of cutting fluids on tool wear and surface roughness during turning of AISI 304 austenitic stainless steel. *J Mater Process Technol*, 209, 900-909.
- Xu, G., Liu, M., Wang, J., Ma, Y., Wang, J., Li, F., & Shen, W. (2019). Data-driven fault diagnostics and prognostics for predictive maintenance : A brief overview. *IEEE 15th international conference on automation science and engineering (CASE)*, 103-108.
- Yang, F., Shah, S., & Xiao, D. (2012). Signed directed graph based modeling and its validation from process knowledge and process data. *Int. J. Appl. Math. Comput. Sci*, 22, 41-53.
- Yang, Z., Zhong, J., & Wong, S. F. (2011). Machine learning method with compensation distance technique for gear fault detection. *In 2011 9th World Congress on Intelligent Control and Automation*, 632-637.
- Zappone, A., Di Renzo, M., & Debbah, M. (2019). Wireless networks design in the era of deep learning : Model-based, AI-based, or both? *IEEE Transactions on Communications*, 67(10), 7331-7376.
- Zaytoon, J. (2001). *Systèmes dynamiques hybrides. Traité IC2 Systèmes Automatisés, Ed. Hermès, ISBN 2-7462-0247-6*.
- Zemouri, R. (2003). *Contribution à la surveillance des systèmes de production à l'aide des réseaux de neurones dynamiques : Application à la e-maintenance*. Thèse de Doctorat. Université de Franche-Comté.
- Zenia, S. (2017). *Modélisation numérique de l'usinage des matériaux composites à matrice polymère et fibres longues de carbone*. These de Doctorat, Université de Lorraine.
- Zhang, Q. (1999). *Identification et Surveillance de systèmes Dynamiques*. Habilitation à diriger les recherches, Université de Rennes1, Institut de Formation Supérieure en Informatique et en Communication, France.
- Ziegler, C. (1996). *Sûreté de fonctionnement d'architectures informatiques embarquées sur automobile*. Thèse de doctorat, LAAS, Toulouse, France.
- Zwingelstein, G. (1995). *Diagnostic des défaillances : Théorie et pratique pour les systèmes industriels*. Hermès.
- Zwingelstein, G. (1996). *La maintenance basée sur la fiabilité—Guide pratique d'application de la RCM*. Hermes, Paris, France

Publications

PETRI NET MODELING OF A MACHINING ROBOT CELL

Zohra MEHAR¹, Rachid NOUREDDINE^{2*}, Farid NOUREDDINE³

¹ PhD student, Industrial Production and Maintenance Engineering Laboratory, Institute of Maintenance and Industrial Safety, University of Oran 2 Mohamed Ben Ahmed, Oran, Algeria, e-mail: mehar.zohra@univ-oran2.dz

^{2*} *corresponding author*, Prof., Industrial Production and Maintenance Engineering Laboratory, Institute of Maintenance and Industrial Safety, University of Oran 2 Mohamed Ben Ahmed, Oran, Algeria, e-mail: noureddine.rachid@univ-oran2.dz

³ Ass. Prof., Production Engineering Laboratory, National School of Engineering in Tarbes, National Polytechnic Institute of Toulouse, France, e-mail: farid.noureddine@enit.fr

In this paper, we propose a new approach for modeling and analyzing faults in a machining robot cell. Therefore, we decided to use a hybrid analysis associating a Petri Net (PN) to a Fault Tree and thus obtained our called Lambda Petri Net (λ PN). This work has been implemented in the LabView environment (Laboratory Virtual Instrument Engineering Workbench). The Lambda Petri Net showed its modeling power for the developed monitoring system. Lambda Petri Nets in fault analysis allow natural language descriptions of process entities. A graphical method is used to describe the relationships between conditions and events. Mathematical generic properties have been used to validate our whole research technique. The simulations and results obtained from the state of the operating system without and with fault are presented and discussed.

Keywords: Petri Net, Monitoring, Machining Robot Cell, Modeling, Fault Tree

1. Introduction

The multiple reconfigurations and the complexity of modern production systems, such as machining robot cells, lead to the design of increasingly efficient monitoring support systems. Considering a robotic cell, we have been interested in modeling the changes in the system dynamics when one or more faults occur. The faults are supposed to be permanent, i.e. when a fault occurs, the system enters a degraded mode and will no longer return to a nominal mode without being repaired first. It can thus end up in a failure mode, in which the system is no longer operational.

In case of failures, minimization of restart time is essential to avoid penalizing the productivity of the concerned cell. We therefore propose and this is our main objective a diagnostic system based on the use of a Petri Net implemented from a fault tree. Petri Nets (PN) are intelligent diagnostic models, constitute a graphical and mathematical tool allowing qualitative and quantitative analysis. They are composed of a set of places (input), transitions (output), and arcs (integer) which are effective for modeling availability and for both production and security systems. The analysis is performed by simulations of behavioral and explanatory models. PNs are capable of managing all sorts of probabilistic distributions. The state of a Petri Net is defined by its marking and by a distribution of tokens in different places.

The most important contribution of this work lies in our designing and implementing of Lambda Petri Nets (λ PN) which are a mathematical formalism that manipulates discrete variables, adequate to the framework of systems failure using tokens and failure rates. These failure rates are deduced from the analysis and modelling of the fault locations of each piece of equipment in a machining robot cell.

For this purpose, our contribution was established in three steps: In the first step, the analysis by FT of the machining robot cell was carried out. In the second step, the transfer of the logic gates from the FT to the PN was completed, followed by the system modeling into

PN. The third step consisted in transforming the PN to the λ PN by introducing the respective failure rates, according to hierarchical expertise of robustness. The failure data for each component were used to calculate the λ PN of the system. The results indicate that the λ PN of the system is strongly influenced by the most critical components. Indeed, components with the highest failure rates have the greatest impact on the system's failure probability. This approach would allow modeling the interactions between failures as events that can occur simultaneously or sequentially.

Generic properties of normative assessment have been applied. The mathematical properties allow the writing of incidence matrices and marking vectors. These qualitative and quantitative analyses were paramount to validate the system. Figure 1 summarizes the monitoring components and focuses on our contribution to the diagnostic system of the machining robot cell. We thus will obtain the different possible causes associated with a degree of credibility and degree of severity for each cause. These degrees will help managers to evaluate and plan maintenance actions.

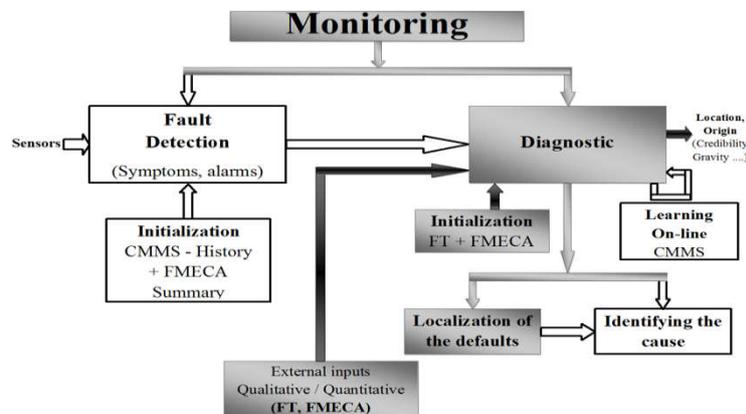


Fig. 1. Block diagram of the monitoring system

State of the art

[1] Propose a new approach for both modeling and failure analysis. They combine graphical representations provided by Petri Nets and fuzzy logic. They used Petri Nets in failure analysis, allowing to replace logic gate functions in fault trees. Fuzzy logic allows natural language descriptions of process entities and thus applies these rules to diagnose breakdowns.

[2] Identified two distinct formalisms for analyzing failure scenarios and availability of considered systems. These two formalisms are Generalized Stochastic Petri Nets (GSPN) on the one hand and Fault Tree Driven Markov Processes (FTDMP) on the other hand.

[3] Propose a methodology to assess the probability of electrical failure of a floodgate by using fault tree and the implementation of stochastic Petri Nets. The authors constructed the stochastic Petri Net with repair by modeling the different operating states of the valve.

[4] Present how Petri Net models have been developed for a wide variety of manufacturing systems. The modeled system was a robot cell for the preparation and analysis of metallographic samples. The model is very complex and with many elements. They achieved the simplification of the obtained model by switching to colored Petri Nets (CPN).

[5] They propose a Petri net model to validate search techniques in graphs, in order to improve the Worst-Case method through dynamic optimization of the number of involved agents.

[6] We propose an algebraic approach to study fault diagnosis classes in a labeled Petri net (LPN) system based on state estimation over a sliding window of length h , and fault detection is performed by solving an Integer Linear Programming (ILP) problem.

2. λ PN Formalization

According to [7], different PNs have particular structures, i.e. they have characteristics and properties that most common networks do not have. State graph, event graph, conflict-free PN, free-choice PN, basic PN, pure PN, self-loop-free PN, Generalized PN, Capacity PN, Autonomous PN, Non-autonomous PN. There are also different types of high-level Petri Nets: synchronized, timed, interpreted, stochastic, colored, hierarchical, continuous, and hybrid.

We propose a Lambda extension of Petri Nets (λ PN) specific to the modeling and analysis of system monitoring activities. The uncertain knowledge associated with these activities requires specific reasoning and modeling methods adapted to the various failure rates.

In cooperation with the ordinary Petri Net and Capacity PN, which model the system to be monitored, this new tool makes it possible to carry out a complete diagnosis of the faults locations and degradations of the system. The lambda Petri Net approach provides more detailed information about the operating state of the monitored system.

Generally speaking, mathematical properties of a PN are: [5,8]

- 1) A Petri Net (PN) is a couple $\{R, M\}$ where: R is a PN, denoted by a quadruple $R = \{P, T, f, M_0\}$ (1)
 - ✓ M is an application from P to \mathbb{N} . $M(p)$ equals the number of marks in a place $p \in P$.
 - ✓ $F: (P \times T) \cup (T \times P) \rightarrow N_0$ Defines the set of directed arcs weighted by non-negative integer values.
- 2) The incidence matrix (W) of the Petri Net translates the global cost of firing a transition for each place.
- 3) Denoted by: $W = W^+ \quad W^-$ ou $W = Post \quad Pre$ (2)

The (i, j) element of matrix W gives the balance for a place i of the firing of the transition j .

 - ✓ The connecting arcs of Transitions to Places $Pre (P_j, T_j)$ can be represented in a matrix with $W^+(i, j) = Pre(P_i, T_j)$
 - ✓ The connecting arcs of Places to Transitions $Post (P_i, T_j)$ can be represented in a matrix with $W^-(i, j) = Post(P_i, T_j)$
- 4) The marking vector of the Petri Net is constructed by the characteristic vector of the sequence \underline{S} which is formed by the number of occurrences of each transition. Let S be a firing sequence, then the state obtained after firing the transitions of S is obtained by (3)

$$M_K = M_0 + (W \times \underline{S})$$

Generic properties of a PN are: liveness, deadlocks, reversibility, repetitive components, effectiveness, reachability, and safety, [5,9].

We started this work with a thorough examination of the monitoring components; more precisely we focused on the diagnostic system with qualitative external inputs (Fault Tree (FT)). As an output, we will find the possible causes associated with fault location. These locations will help the managers to assess and plan maintenance actions. In the overall classification of monitoring methods and models, we have concentrated on monitoring methods with models, exactly on the methods by functional and material modeling (FT and PN).

We propose a new tool called Lambda Petri Net (λ PN). This Network describes the functioning of systems that are not autonomous. Their operation is conditioned by failure rates.

A Lambda Petri Net consists of two parts: a static part and a dynamic one. The static part defines the structure of the lambda Petri Net, where the data is stored, and how the data interacts with each other. The dynamic part defines the initial state of the system. Indeed, the same lambda Petri Net will not have the same behavior depending on its initial state, so it is important to separate the two concepts.

The modeling of our diagnostic system can be done by different types of Petri Nets (ordinary PN, high-level PN), assuming that the possible faults are known a priori and modeled by specific mechanisms. Our approach deals with Lambda Petri Net modeling at the level of the transitions.

To model the monitoring function, we use an extension of the PN, which integrates through the lambda aspect the failure rate in the monitored system. The λ PN is oriented for modeling a base of failure rate rules which follows from the logical expression of the fault tree of the monitored system. The λ PN tool models the set of the logical reasoning of the FT, according to the specific concepts of a logical expression. The analysis aspect offers refined information at the level of each defect by the transfer of fault signals.

The main advantage of Petri Lambda Nets is their strong mathematical foundations. In addition, a great deal of software programs make it possible to simulate and analyze Lambda Petri Nets. Using lambda Petri Nets in industrial systems has several advantages in terms of reduced wiring and ease of monitoring and maintenance. Inputs and outputs in λ PN allow easy modeling and access to the markings of all places at any time. These elements make λ PN an effective and adequate tool for our modeling needs to get simulation support.

The disadvantages of Lambda Petri Nets is their modeling complexity, which lead to producing errors. When using λ PN, the delays are no longer negligible and must be taken into account, especially when the quality service of the Network changes over time which results in non-periodic activation moments.

3. Transfer PN-FT

FT is a method of deductive analysis used in dependability, [10,11]. This method analyzes the reliability, availability, and security of more widely used systems, [12]. FT is the simplest and most used technique to assess reliability, [13-15].

According to [16,17], a fault tree in general is divided into two categories: coherent and non-coherent. A coherent fault tree consists only of logical operations “AND” and “OR”, while a non-coherent tree also contains other logical operations.

This article is mainly concerned with the application of Petri Net modeling to coherent fault tree analysis [18].

The use of knowledge from a Fault Tree with the Petri Nets principle through the reasoning must be done qualitatively but also quantitatively to provide effective analysis, like the case in industrial applications, [19,20].

A logical relationship exists between Petri Nets and fault trees. The formalism used in our work allows to implement the PN from the FT. We thus use a formalism that associates each gate of the FT as a symbolization of the PN.

1) The “AND” logic gate is as follows, figure 2:

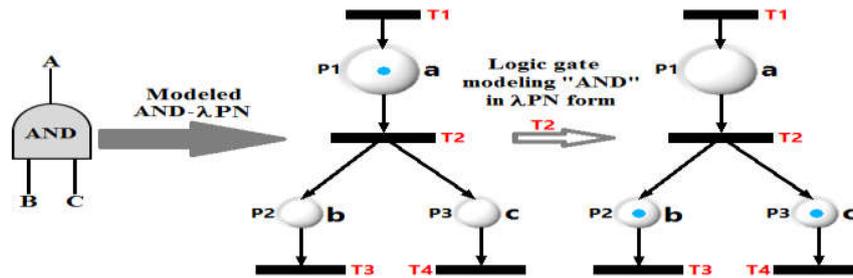


Fig. 2. Transformation of the « AND » logical gate of the FT into the PN

2) The “OR” logic gate is as follows, figure 3:

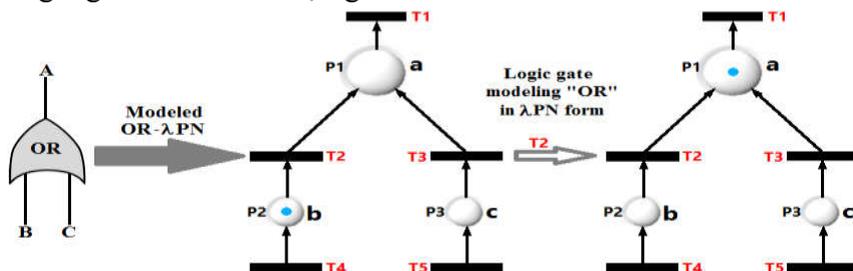


Fig. 3. Transformation of the « OR » logical gate of the FT into the PN

4. Case study

4.1 Problem

Our case study has been conducted in collaboration with the Machining Robot Cell of the Production Engineering Laboratory of the National Engineering School in Tarbes (ENIT), France. Our machining robot cell in Figure 4 is composed of:

- A KUKA KR120 robot,
- An electro spindle,
- A milling tool,
- The part to be machined in its clamping feature,
- The lubrication system.



Fig. 4. Machining robot cell

Many authors have been interested in the reliability field of cutting tools and in modeling the surface roughness of machined parts.

[21] Have studied and modeled a new basic hazard function called GEV proportional hazard function. This generalized function with extreme values is obtained thanks to the Gumbel function and the property of being non-monotonic, an increasing then decreasing function, thus capable of taking into account the cutting tool behavior with different

mechanisms of self-repair and self-recovery after a tiny breakage. The authors introduce parameters considering operating and environmental conditions, including vibration signals, material hardness, and lubrication/cooling. Their results show the impact of all these variables on the surface roughness of machined parts.

- 1) Vibration signal: Two types of vibration occur during machining that are sustained and self-sustained vibrations. Self-sustained vibrations are characterized by an absence of periodic external forces but depend on the chip formation process. On the other hand, sustained vibrations originate from periodic external forces such as those acting on the part during intermittent cutting.

[22] Have studied similar cutting tool conditions and the acceleration amplitude of tool vibrations in the axial, radial, and tangential directions. Table 1 shows the impact of experimental values of acceleration amplitude of vibration (Root Mean Square (RMS) value) in axial (V_x), radial (V_y), and tangential (V_z) directions on the surface roughness (R_a). The worst case is the variation in $V_y(g)$.

- 2) Material hardness: Both a decrease in the used material hardness and the reliability of the applied heat treatment lead to the deterioration of the surface roughness.

[23] Have studied the influence of cutting tool conditions and material hardness on surface roughness. Their obtained results are given in Table 1.

- 3) Lubrication/cooling conditions: High cutting temperature during machining can lead to unacceptable surface quality. Compared to dry cutting, the lubrication/cooling conditions lead to a significant improvement in surface roughness.

[24] Show all the impacts of cutting conditions on surface roughness and tool wear in the turning process. The contribution of cooling/lubrication is around 14%.

[21] Studied the influence of vibrations, material hardness defects, and lubrication. The results are presented in Table 1.

We can deduce that all the contributions of the variables deteriorate the basic hazard function.

Table 1.

Machining parameters considered				
Conditions de coupe				
Vitesse de coupe (V_c) (m/min)	Vitesse d'avance (f) (mm/rev)		Profondeur de coupe (d) (mm)	
250	0.1		0.5	
Impact des vibrations sur la rugosité de surface				
	$V_x(g)$	$V_y(g)$	$V_z(g)$	$R_a(\mu m)$
Expérience 1	0,37	3,92	1,63	1,51
Expérience 2	0,39	4,84	1,65	1,73
Impact de la dureté du matériau sur la rugosité de surface				
	Dureté (HB)		$R_a(\mu m)$	
Expérience 1	130		1,53	
Expérience 2	240		1,35	
Conditions de lubrification/refroidissement				
Signal de vibration (covariable X_1)	Dureté du matériau (covariable X_2)		Lubrification (covariable X_3)	
$\varphi(\beta_1 X_1) = 1,15$	$\varphi(\beta_2 X_2) = 1,13$		$\varphi(\beta_3 X_3) = 0,86$	

4.2 FT Analysis

The Fault Tree (FT) of the robot cell was the centerpiece of our PN-based strategy and is presented in figure 5. The analysis and research of the dreaded event in our FT also referred to as the tree top event, highlights the non-conforming piece (a), Figure 5, in the robot cell.

If we search for the cause of an undesirable event, it can be due to a fault on this very element or to a fault on any other element of the system.

We used CABTREE software to build and process our fault trees. We have limited our study to two levels, which show the first elementary elements, as shown in figure 5.

- First level: Defective KUKA KR 120 Robot, Faulty Spindle, Faulty Tool, Failing Piece, Malfunctioning Lubrication System (the lubrication item is mainly a pump).
- Second level: Faulty Feed rate (f), Faulty Depth of cut (d), Vibration in axial (Vx) directions, Vibration in radial (Vy) directions, Vibration in tangential (Vz) directions, Faulty Cutting speed (Vc), Break on the Tool, Bad positioning of the Piece, Failing Hardness, Low Flow, Bad Liquid of the lubrication pump.

Logic gates can model the Boolean function F of the dreaded event of our FT. In our work, we used the "OR" logic gate. To illustrate our approach, we consider the logical equation F of the fault tree:

$$F = (b1 \text{ OU } b2 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (c1 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (d1 \text{ OU } b3 \text{ OU } b4 \text{ OU } b5) \text{ OU } (e1 \text{ OU } e2) \text{ OU } (f1 \text{ OU } f2)$$

The (+) \Leftrightarrow OR operator represents the union of logical variables {a, b, c, d, e, f, b1, b2, b3, b4, b5, c1, d1, e1, e2, f1, f2}.

$$F = \underbrace{\left(\underbrace{(b1 + b2 + b3 + b4 + b5)}_b + \underbrace{(c1 + c3 + b4 + b5)}_c + \underbrace{(d1 + b3 + b4 + b5)}_d + \underbrace{(e1 + e2)}_e + \underbrace{(f1 + f2)}_f \right)}_a \tag{4}$$

Such as:

$$\begin{array}{l|l|l} b = b1 + b2 + b3 + b4 + b5 & d = d1 + b3 + b4 + b5 & f = f1 + f2 \\ c = c1 + b3 + b4 + b5 & e = e1 + e2 & \end{array}$$

$$F = [b + c + d + e + f] \tag{5}$$

$$F = a \tag{6}$$

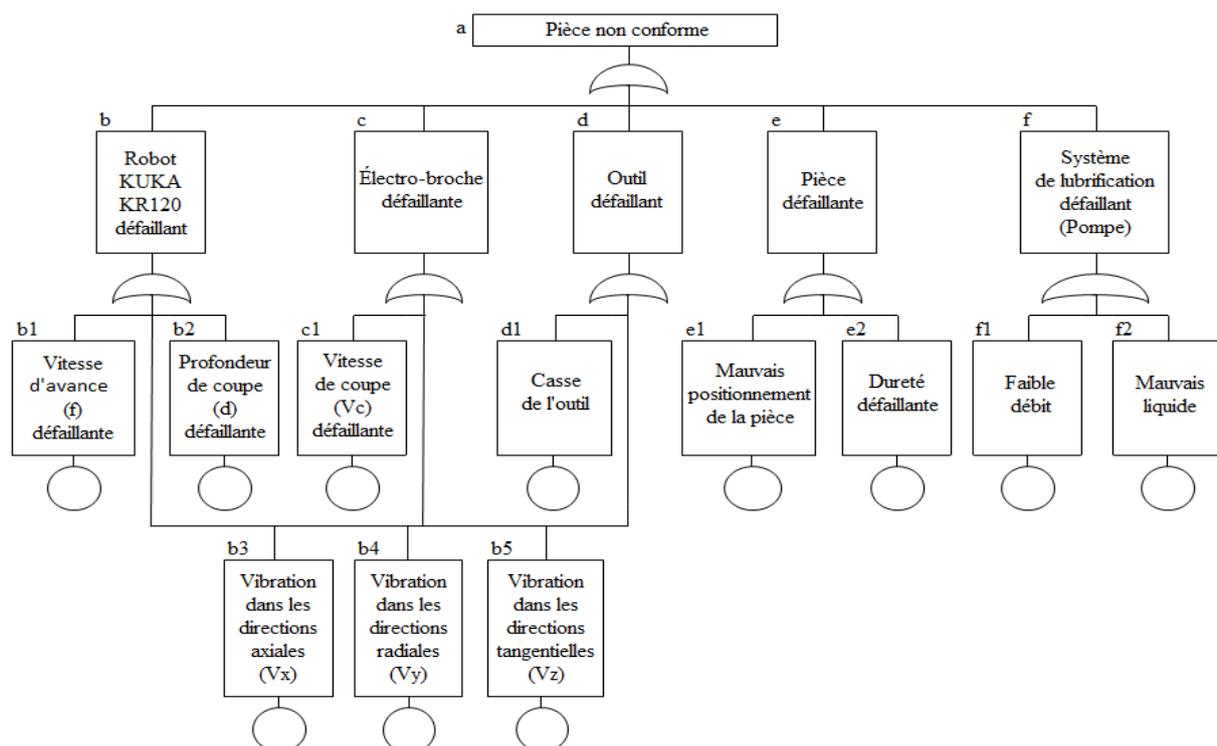


Fig. 5. Fault Tree of the robot cell, corresponding to F

4.3 Analysis by λ PN

The machining robot cell is shown in figure 4. Its operation using the PN appears in figure 6. Our PN model is represented by 17 places and 28 transitions, respectively denoted from P1 to P17 and the transitions from T1 to T28. The number of tokens in P_i place represents the number of failures in the robot cell. The transition T_i located between the places P_i are transitions for all the actions performed by the robot cell, with all their connections. This transition is immediate. The number of tokens in place P_i corresponds to the number of faulty elements in our robot cell. The weights of the arcs are indicated on the model next to the arcs. The absence of a firing means that the arc in question is implicitly weighted at 1.

Our PN can represent the successive assembly and disassembly of a single element (P1). Thus, there are two-state processes (Stop-Start), and the passage from one state to the other mobilizes a resource, symbolized by the token contained in the arcs, which was added between the places and the transitions. Containing a source place (P1) and a source transition (T1), this transition is always sensitized but with a capacity counter ($Cap(P1) = 5$).

Additional capacity has been given to the PN as in the generalized PN; there is no limitation on the number of tokens per place. Capacity Petri Net is a PN in which capacity that are strictly positive whole numbers, is associated with the places. The firing of an input transition of a place P_i which capacity is a $Cap(P_i)$, is only possible if the firing does not lead to a number of tokens in P_i greater than $Cap(P_i)$.

In our example given in figure 6, firing T1 leads to 5 tokens in P1. T1 can thus no longer be fired.

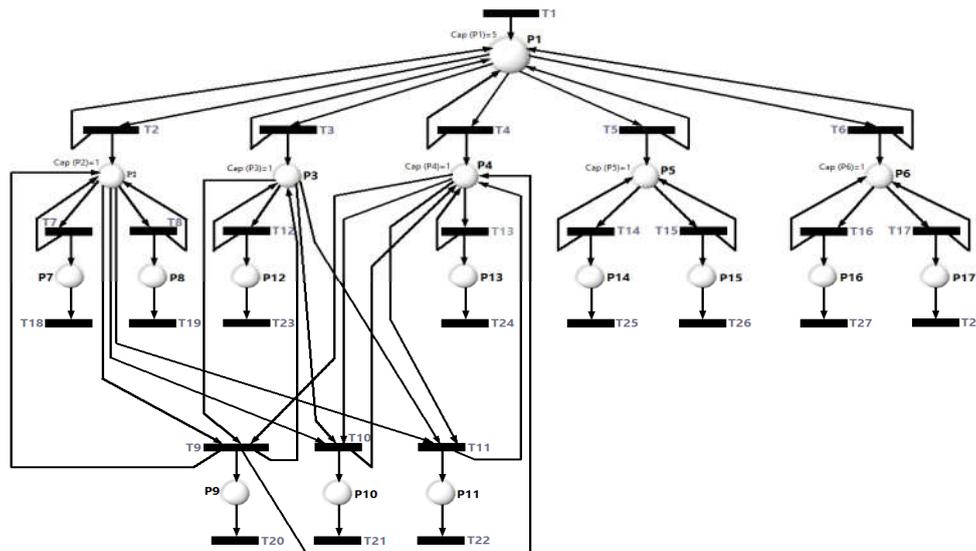


Fig. 6. Petri Net before firing of transitions

Our objective is to control and automate the considered FT system using its PN model and using lambda failure rates. To do this, it is necessary to convert the PN model shown in Figure 8 into its equivalent λ PN model. The possible transformation from ordinary PN to high-level λ PN is shown in figures 7 and 8.

Our λ PN before firing of transitions is shown in figure 7.

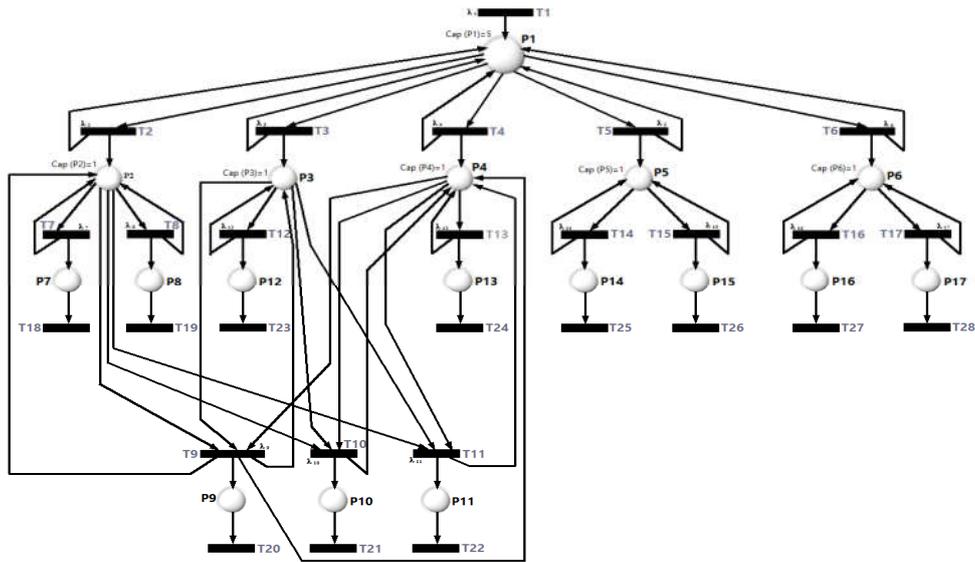


Fig. 7. λ PN before firing of transitions

Our λ PN after firing of transitions is shown in figure 8.

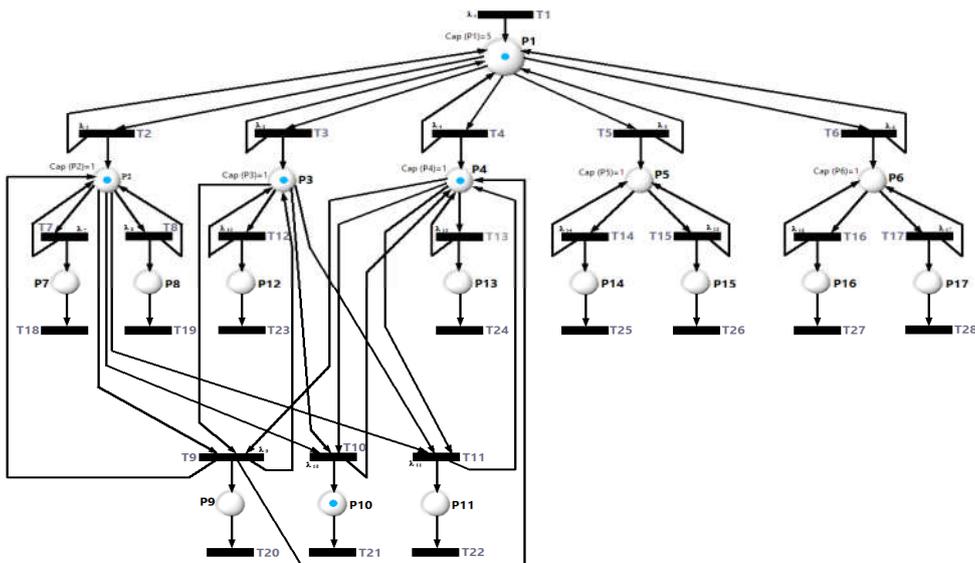


Fig. 8. λ PN after firing of transitions

Transitions (T1, T2, ... T28) of the Petri Net are materialized by sensors. The messages used on the transitions are failure rates between 0 and 1. This communication type allows a relatively simple modeling of all states of the system dynamical behaviors. Table 2 describes the significance of each place and the different failure rates used in the Lambda Petri Net of our system are shown in table 3.

Table 2.

Significance of the places

Places	Significance of each place
P1	Non-conforming piece
P2	Defective KUKA KR 120 robot
P3	Faulty spindle
P4	Faulty tool
P5	Failing piece

P6	Malfunctioning lubrication system (pump)
P7	Faulty feed rate (f)
P8	Faulty depth of cut (d)
P9	Vibration in axial (Vx) directions
P10	Vibration in radial (Vy) directions
P11	Vibration in tangential (Vz) directions
P12	Faulty cutting speed (Vc)
P13	Break on the tool
P14	Bad positioning of the piece
P15	Failing hardness
P16	Low flow
P17	Bad liquid

5. LabView Implementation

We have applied our fault tree transformation technique (FT) in a Lambda Petri Net (λ PN). We used the LabView environment platform for modeling and simulation. LabView is based on a graphical development environment of «National Instruments», and is mainly used for instrument control and industrial automation.

5.1 FT – LabView Implementation

We have proposed the implementation in the LabView environment of the Lambda Petri Net of the case study with failure rates of the different FT components. Our model is shown in the following figures 9 and 10.

3) The implementation-modeled AND- λ PN under LabView will be as follows, figure 9:

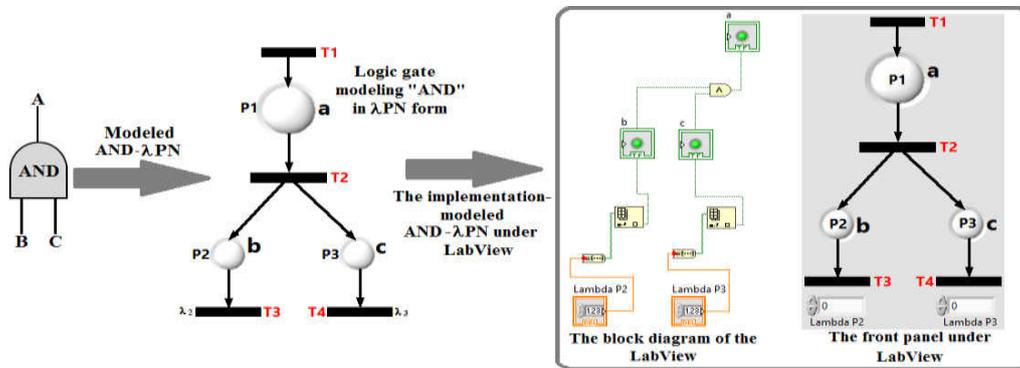


Fig. 9. Transformation of the « AND » logical gate of the FT into the λ PN

4) The implementation-modeled OR- λ PN under LabView will be as follows, figure 10:

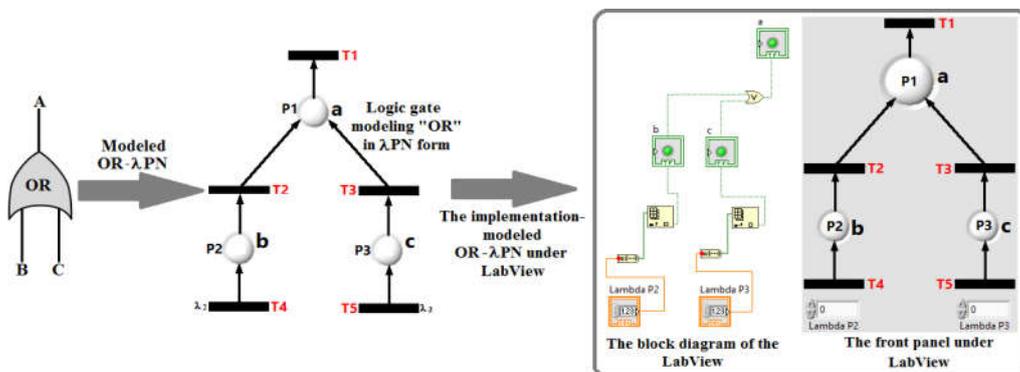


Fig. 10. Transformation of the « OR » logical gate of the FT into the λ PN

5.2 λPN - LabView Implementation

We have associated the λPN to a LabView state machine. The resulting structure is shown in figures 11 and 14. The front panel is the user interface of VIs (Virtual Instruments) in our system. It is shown in figures 11 and 14 and describes an analysis application called Dominant Failure Mode. The λPN: contains 17 places that are circular LEDs (Light Emitting Diodes) emitting 2 phases of light, white and grey, and 28 transitions. Each place presents an event of our FT and describes its state: inputs (commands) and outputs (indicators) of the program. String Indicators model these states and another String Indicator displays the state of our system (State of the non-conforming piece). This indicator is used to display Normal or Abnormal Operation.

Our application consists of a box that contains a Digital Indicator for the sum of the different failure rates of our FT. It also includes two other Digital Indicators, the first one to display the highest failure rate of level 1 to the FT and the second to display the highest failure rate of level 2 to the FT. 11 Numeric Controls and 5 Digital Indicators are materialized by failure rates. The different failure rates are values between 0 and 1.

The inputs variables are the failure rate of each component. They are labelled as «Numeric Control» in LabView and the results appear as «Digital Indicator» on the front panel.

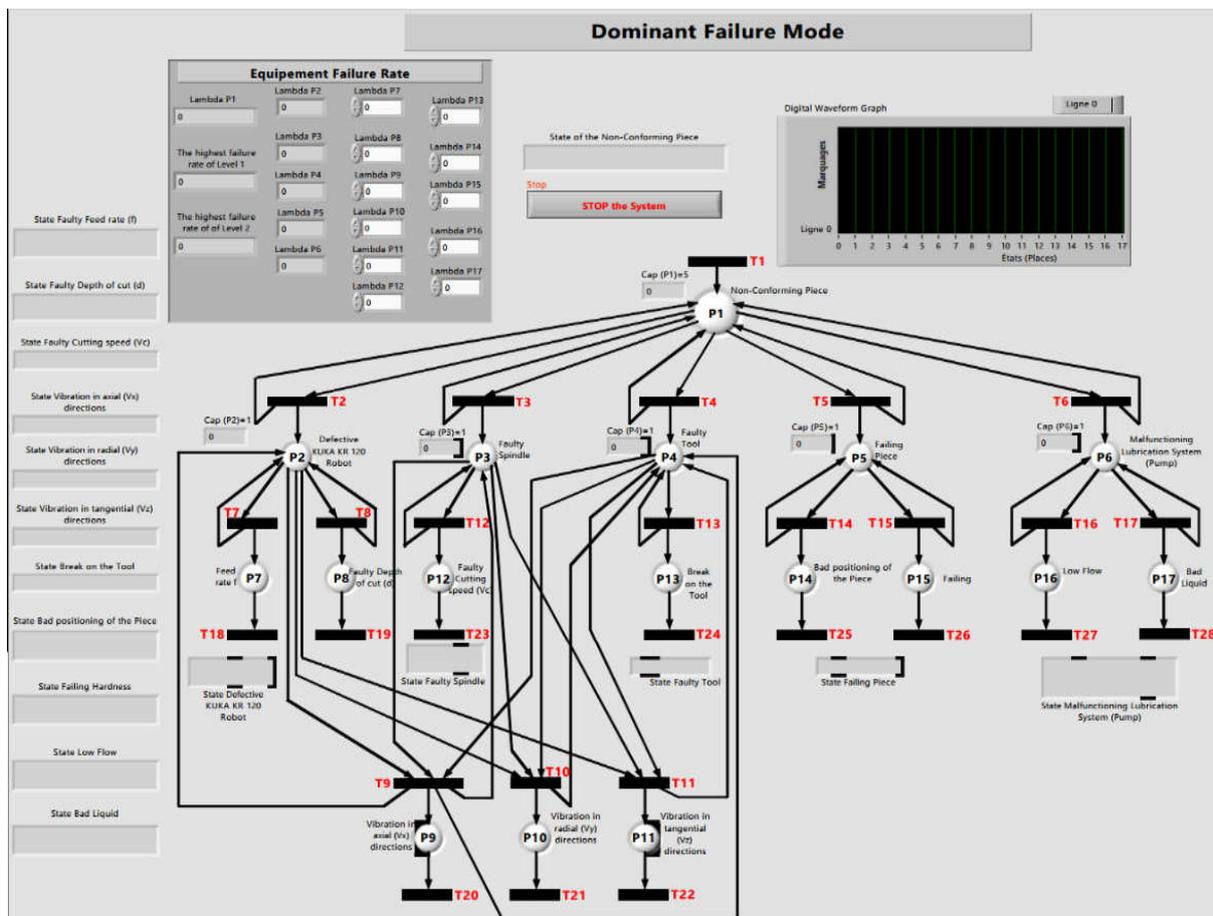


Fig. 11. Front panel under LabView of the PN system - Modeling without fault

The block diagram, figure 12, represents the application program written in the form of a data flow diagram. This figure illustrates how command and indicator are materialized by digital displays for a state 0 or 1 in the block diagram of LabView.

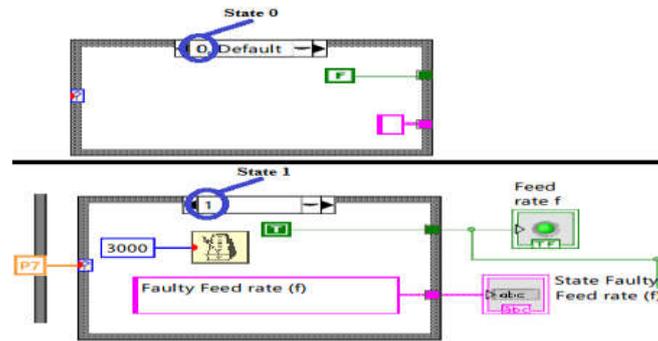


Fig. 12. LabView block diagram of the λPN system

6. Simulations and results

As shown in figure 13, the simulation of the proposed diagnostic system was carried out in three essential steps. First, we assessed the failure rates of all system components and then we used LabView to perform the simulation, allowing to observe the distribution of the system states. Finally, in the third step, we successively identified the marking of the Lambda Petri Net simulation, obtained its generic properties for automatic checking, determined its incidence matrix and obtained its marking vector needed to validate the model.

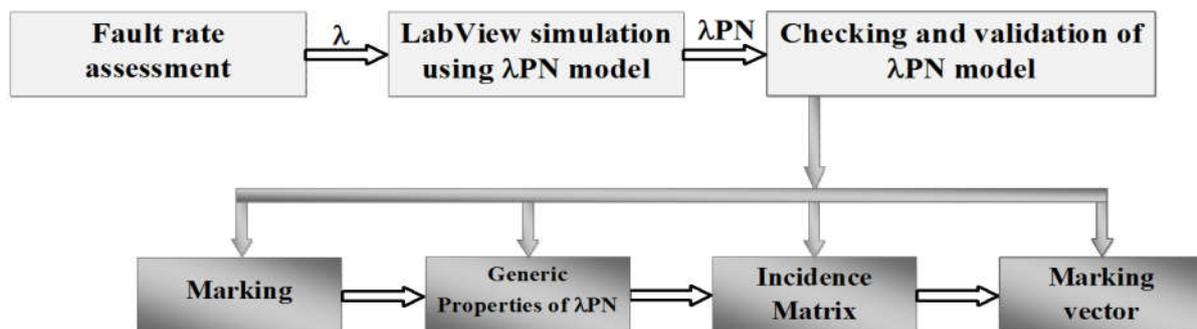


Fig. 13. Steps of the proposed diagnostic system

6.1 Failure rate

According to the hierarchical expertise of robustness and thanks to the values given in table 1, the failure data for each component is given in the following table 3. Vibrations are the most critical events. Vibrations in radial direction have the greatest value of failure rate ($\lambda_{10} = 0.006$).

Table 3.

Failure rate of level 2 and dreaded event – FT - λPN

Niveau 2 - AdF - RdPλ			
λ_7	0.0001	λ_{13}	0.0003
λ_8	0.0001	λ_{14}	0.0003
λ_9	0.0004	λ_{15}	0.0002
λ_{10}	0.0006	λ_{16}	0.0002
λ_{11}	0.0005	λ_{17}	0.0003
λ_{12}	0.0001		

According to equations (4), (5), and (6) and Table 3, we obtained the following failure rates shown in table 4. The faulty tool is the most critical event with a failure rate $\lambda_4 = 0.0018$.

Table 4.

Niveau 2 - AdF - RdPλ				
λ_2	λ_3	λ_4	λ_5	λ_6
0.0017	0.0016	0.0018	0.0005	0.0005
Événement redouté - AdF - RdPλ				
λ_1			0.0061	

6.2 LabView Simulation

The obtained simulations and results are shown in figure 14. We use tokens in the graph places to signal the state of each resource at a given moment, it is marked in gray.

At the second FT level and corresponding to its failure rate, the component « Vibration in radial (Vy) directions » is in failure mode, therefore the corresponding state is activated.

According to the diagnostic characteristics of FT and equations (4), (5), (6), the dreaded event is $\lambda_1 = 0.0061$. After comparing all failure rates of level 1, the highest failure rate is $\lambda_4 = 0.0018$, so the highest level 2 failure rate is $\lambda_{10} = 0.0006$.

- So respectively, place (P10) is colored gray, and its signal is in state 1.
- If the place (P10) is faulty then (P4) is initially faulty and then (P1) is faulty. These two places are colored in gray, and their signal is in state 1. If there is a faulty place, the triggered diagnostic process makes the system fail. So the capacity Cap(P4) = 1 of place (P4) is displayed as 1 and the finite capacity Cap(P1) = 5 of place (P1) is displayed as 1.
- To carry out a deductive analysis in our λ PN, we proceed by firing transition (T1), and then place (P1) has a token. If there is a token in place (P1), then we have to go to transition (T4) directly. If (T4) is fired then place (P8) has a token. If place (P8) has a token, then (P10) also has a token after firing transition (T10). This diagnostic process is obtained through the return arcs building our Lambda Petri Net.

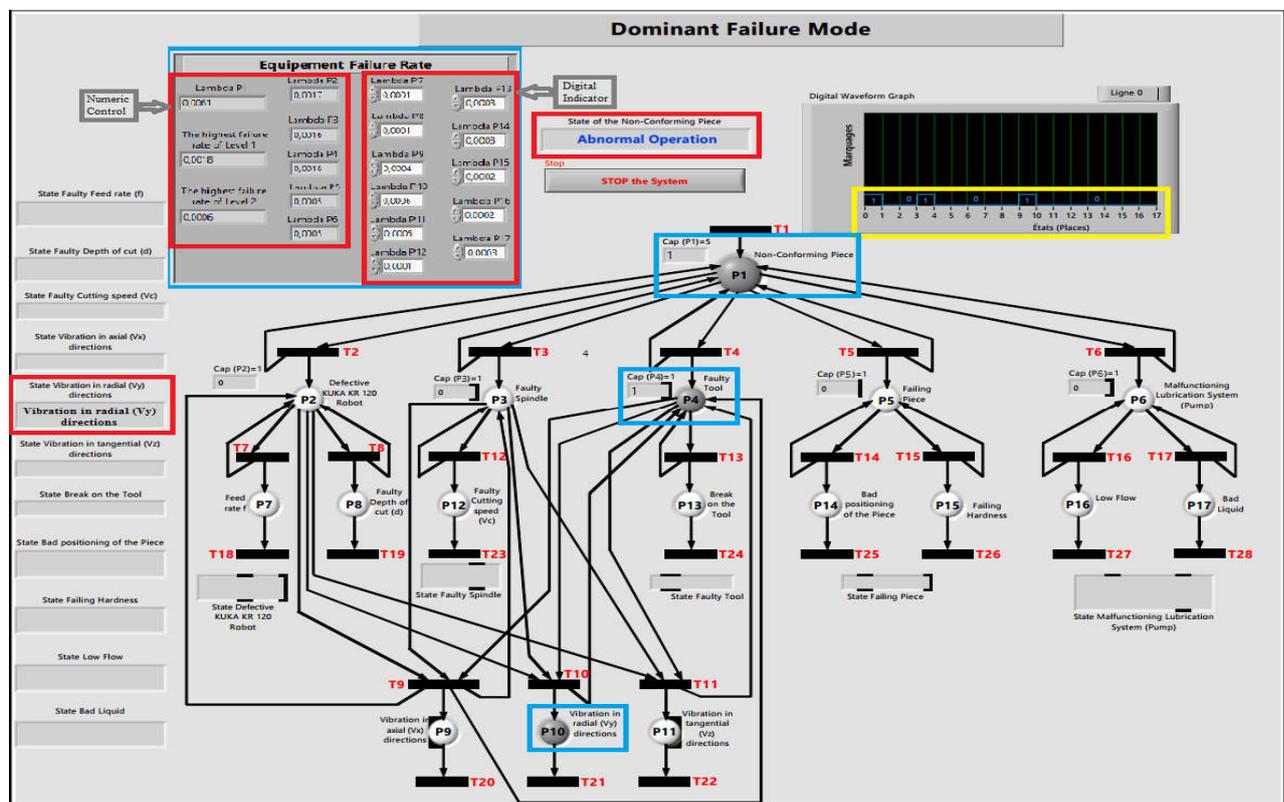


Fig. 14. Front panel under LabView of the λ PN system - Modeling with fault

6.3 Marking

Our interest at this level lies in the control of the machining robot cell presented in figure 4. Its fault tree is given in figure 5, and is associated with the λ PN of figure 7. The result after the firing according to the faults is the model shown in figure 8. We also defined the marking here to interpret the Lambda Petri Net simulation results. It allows us to understand the behavior of the system, identify the states of the system at different times and track its evolution by following transitions.

According to equation (1), we have:

- The initial marking of our λ PN, corresponding to figure 10, is $M_0 = [000000000000000000]$.
 - The marking of our λ PN after the firing, corresponding to figure 12, is $M_1 = [100100000100000000]$.
- $P = \{P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12, P13, P14, P15, P16, P17\}$
 $T = \{T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15, T16, T17, T18, T19, T20, T21, T22, T23, T24, T25, T26, T27, T28\}$

6.4 Generic Properties of Lambda Petri Nets

In order to check the consistency of the model and detect possible undesirable behaviors we used the generic properties. Generic properties are fundamental characteristics of Petri Nets that allow the analysis of their behavior. These properties include Boundedness; Safeness; Liveness; Deadlock; Reversibility; Repetitive; conflictual; Reachability.

We obtained the following generic properties for the automatic checking of our system: our λ PN is unbounded but safe. λ PN is lively, deadlock-free, non-reversible, non-repetitive, conflict-free, reachable, and safe, with a graph of infinite markings.

6.5 Incidence Matrix

The incidence matrix represents the relationships between places and transitions in a Petri Net, allowing to calculate the marking of the system after a transition. The incidence matrix (W) of our Lambda Petri Net (λ PN), after applying equation (3) is a matrix of 17 rows (places) and 28 columns (transitions), and is represented in the following matrix:

$$W = \begin{matrix} & \begin{matrix} T1 & T2 & T3 & T4 & T5 & T6 & T7 & T8 & T9 & T10 & T11 & T12 & T13 & T14 & T15 & T16 & T17 & T18 & T19 & T20 & T21 & T22 & T23 & T24 & T25 & T26 & T27 & T28 \end{matrix} \\ \begin{matrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \\ P7 \\ P8 \\ P9 \\ P10 \\ P11 \\ P12 \\ P13 \\ P14 \\ P15 \\ P16 \\ P17 \end{matrix} & \begin{bmatrix} 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \end{matrix}$$

6.6 Marking vector

The marking vector represents the marking of the λ PN in a compact form; it proves useful for the analysis of the results. The marking vector of our λ PN is obtained after a firing sequence according to equation (3): $M_k = M_0 + (W \times \underline{S}_1)$.

So we use the M_1 marking for the calculation of the transition vector.

$\underline{S} = [T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15, T16, T17, T18, T19, T20, T21, T22, T23, T24, T25, T26, T27, T28]$

REFERENCES

- [1] F. Lajmi, J. Ghabi, et H. Dhoubi, « Applying Interval Fuzzy Petri Net to failure analysis », *International Journal of Engineering and Technology*, vol. 11, n° 1, 2020.
- [2] M. Talebberouane, F. Khan, et M. Z. Kamil, « Dynamic RAMS Analysis Using Advanced Probabilistic Approach », *Dynamic RAMS Analysis Using Advanced Probabilistic Approach, Chemical Engineering Transactions*, vol. 77, p. 241-246, 2019.
- [3] K. F. K. Donchi, G. Ftatsi Mbetmi, D. J. FOTSA-MBOGNE, et B. DJEUMAKO, « Modélisation et estimation de la fiabilité dynamique d'un évacuateur de crues par réseaux de Petri stochastiques », 2022.
- [4] F. S. Blaga, A. Pop, V. Hule, et C. I. Indre, « The efficiency of modeling and simulation of manufacturing systems using Petri nets », *IOP Conference Series: Materials Science and Engineering, The Annual Session Of Scientific Papers - IMT Oradea 2021*, 2021.
- [5] A. Djellal et R. Lakel, « Petri Net Model of Pursuit-Evasion Game in a Graph », *International Journal of Control and Automation*, 2017.
- [6] A. Chouchane et P. Declerck, « Diagnosis on a sliding window for partially Observable Petri Nets », *Kybernetika*, vol. 58, n° 4, p. 479-497, 2022.
- [7] R. David et H. L. Alla, « Du Grafset aux réseaux de Petri ». 2eme Edition revue et augmentée, Ed. Hermes Science Publications, 1992.
- [8] M. Diaz, « Fundamental Models, Verification and Applications », *ISTE & Wiley, N 978-1-84821-079-0*, 2009.
- [9] L. Hongwei *et al.*, « Concurrency bugs in multithreaded software: modeling and analysis using Petri nets », *Discrete Event Dynamic Systems*, vol. 23, p. 157-195, 2013.
- [10] R. Ghostine, J.-M. Thiriet, J.-F. Aubry, et M. Robert, « A Framework for the Reliability Evaluation of Networked Control Systems », *Proceedings of the 17th World Congress, The International Federation of Automatic Control, Seoul, Korea*, vol. 41, n° 2, p. 6833-6838, 2008.
- [11] N. Yel, R. Noureddine, et F. Noureddine, « Model-based approach for fault monitoring in machining », *Euro-Mediterranean Seminar on Environment and Industrial Safety*, 2011.
- [12] E. Ruijters *et al.*, « FFORT: A benchmark suite for fault tree analysis ». The 29th European Safety and Reliability Conference, Singapore, 2019.
- [13] M. Bouissou, Y. Dutuit, et S. C. Maillard, « Reliability Analysis of a Dynamic Phased Mission System: Comparaison of two approaches », *MMR2004*, vol. 2, p. 1-19, 2004.
- [14] M. Talebberouane et Z. Lounis, « Safety assessment of flare system by fault tree analysis », *Journal of Chemical Technology & Metallurgy*, vol. 51, n° 2, p. 229-234, 2016.
- [15] M. Talebberouane, F. Khan, et Z. Lounis, « Availability Analysis of Safety Critical Systems Using Advanced Fault Tree and Stochastic Petri Net Formalisms », *Journal of Loss Prevention in the Process Industries 44*, 2016.
- [16] W. S. Lee, D. L. Grosh, F. A. Tillman, et C. H. Lie, « Fault Tree Analysis, Methods, and Applications-A Review », *IEEE Transactions on Reliability*, vol. 34, n° 3, p. 194-203, 1985.
- [17] G. S. Hura et J. W. Atwood, « The Use of Petri Nets to Analyze Coherent Fault Trees », *IEEE Transactions on Reliability*, vol. 37, n° 5, p. 469-474, 1989.
- [18] S. Kabir et Y. Papadopoulos, « Applications of Bayesian networks and Petri nets in safety, reliability, and risk assessments: A review », *Safety Science*, vol. 115, p. 154-175, 2019.
- [19] M. Monnin, D. Racoceanu, et N. Zerhouni, « Overview on diagnosis methods using artificial intelligence application of fuzzy Petri nets », *IEEE Conference on Robotics, Automation and Mechatronics*, vol. 2, p. 740-745, 2004.
- [20] N. Palluat, D. Racoceanu, et N. Zerhouni, « Diagnosis aid system using a neuro-fuzzy Approach », *Advances in Maintenance and Modeling, Simulation and Intelligent Monitoring of Degradation*, 2004.
- [21] D. Abdoulaye, N. Farid, N. Rachid, K.-F. Bernard, et B. Diakarya, « Proportional hazard model for cutting tool recovery in machining », *Journal of Risk and Reliability*, 2019.
- [22] V. Upadhyay, P. K. Jain, et N. K. Mehta, « In-process prediction of surface roughness in turning of ti-6al-4v alloy using cutting parameters and vibration signals. », *Measurement*, vol. 1 (46), p. 154-160, 2013.
- [23] N. M. Vaxevanidis, J. Kechagias, et N. A. Fountas, D. E. Manolakos « Evaluation of machinability in turning of engineering alloys by applying artificial neural networks. », *The Open Construction & Building Technology Journal*, vol. 8 (1), p. 389-399, 2014.
- [24] A. Xavier et M. Adithan, « Determining the influence of cutting fluids on tool wear and surface roughness during turning of AISI 304 austenitic stainless steel. », *J Mater Process Technol*, vol. 209, p. 900-909, 2009..

Diagnostic by Fault Tree and Petri Nets of a Robotic Machining Cell

Zohra MEHAR ^{1*}, Rachid NOUREDDINE ², Farid NOUREDDINE ³

¹Production and Maintenance Engineering Laboratory, Institute of Maintenance and Industrial Security, University of Oran 2 Mohamed Ben Ahmed, B.P 1015 El M'naouer, Oran, 31000, Algeria

²Production and Maintenance Engineering Laboratory, Institute of Maintenance and Industrial Security, University of Oran 2 Mohamed Ben Ahmed, B.P 1015 El M'naouer, Oran, 31000, Algeria

³Production Engineering Laboratory, National School of Engineering in Tarbes, National Polytechnic Institute of Toulouse, 47 avenue d'Azereix- BP 1629 - 65016 Tarbes CEDEX, France

*E-mail: mehar.zohra@univ-oran2.dz

Abstract. The objective of this work is to propose a diagnostic technical method for a robot cell machining. The developed approach is based on a hybrid diagnostic FT-PN (Fault Tree - Petri Nets). This technique consists of the Petri Nets implementation in the LabView environment (Laboratory Virtual Instrument Engineering Workbench), based on the knowledge of the Fault Tree of the robot cell. The simulations and results are obtained from the state of operational system without and with fault are presented and discussed.

Key words. Diagnostic, Robotisation, FT, PN, LabView.

1. Introduction

In Industry 4.0, robotisation has imposed as an element of primary factor. The current performance of robots in terms of precision tracking trajectories allow to consider robotisation in machining and more specifically on operations such as surfacing, drilling, trimming, sawing, deburring, grinding and polishing. Like any technological system, maintenance is mandatory and necessary to guarantee the performance. In the case of failures, the minimization of the restart time is essential so as not to penalize the productivity of the cell concerned.

It is in this context we propose a diagnostic system based on a petri nets implemented from a fault tree.

The objective of the diagnostic system is to search for the causes and locate the organs that caused a particular anomaly which consist of two elementary functions: localization and identification, (Isermann & Ballé, 1997) (Mahdaoui, 2013).

In the proposed approach, the diagnostic system input (figure 1) shows the external inputs of qualitative or quantitative type that the operator can add to improve the diagnostic. As output, we will find the different possible causes associated with a degree of credibility and a degree of gravity for each of them. These degrees will help the manager now to evaluate and plan maintenance actions, (Palluat, 2006) and (Palluat et al., 2004).

In this work, we present a diagnostic method based on a deductive analysis from a Fault Tree of a robot cell machining followed by an implementation of a Petri Nets (PN).

In the following section, we present the Petri Nets formalism. It follows the study of the studied case (robotic cell), where the implementation of the FT in a PN in the LabView environment is realized. Finally, the simulations and results are presented.

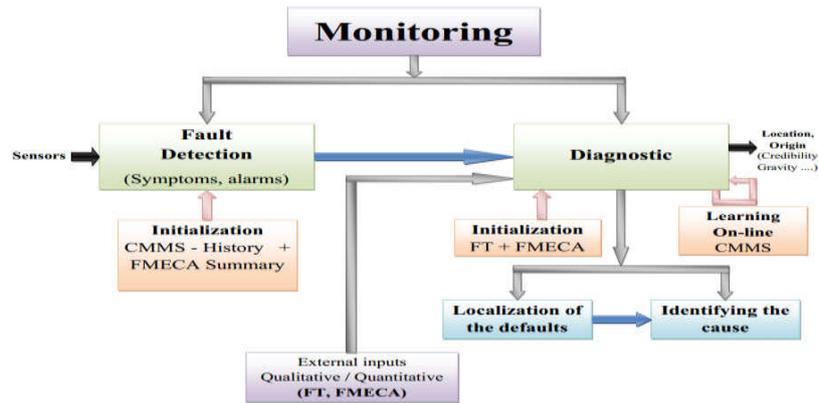


Fig.1. Diagnostic system

2. Petri Nets modeling

Petri nets (PN) are a graphical and mathematical tools, (Brauer et al., 1985). They can describe existing relations between conditions and events, to model the behavior of dynamic systems with discrete events. This tool allows qualitative and quantitative analysis, (Petri, 1962) (Morère, 2002).

The petri nets (PN) is composed of a set of places, a set of transitions, and a set of arcs that associate the input places with the transitions and transitions to output places weight integers associated with the arcs (Boucherit, 2019), (Haiouni, 2010). The state of a petri network is defined by its marking which associates with each place a positive or zero finite number, and graphically represents by marks or tokens, (Petri, 1962) (Morère, 2002) and (Frank & Olivier, 2003). There are different types of Petri nets (David & Alla, 1992) : timed, interpreted, stochastic, colored, continuous and hybrid.

The use of knowledge from a Fault Tree with the Petri Nets formalism through the reasoning must be done qualitatively but also quantitatively to provide an effective analysis, like the case in industrial applications, (Palluat, 2006) and (Racoceanu, 2006). We therefore use the formalism that associates with each gates of the FT a symbolization of the PN. So, the logical gate « OR » and the logical gate « AND » will be transformed according to figure 2 and 3:

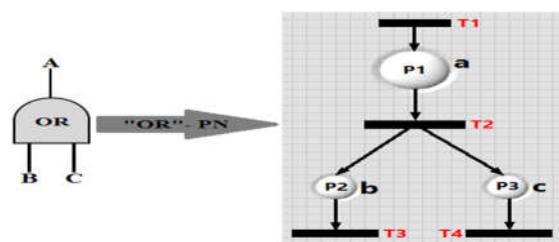


Fig.2. Transformation of the « OR » logical gate of the FT into the PN

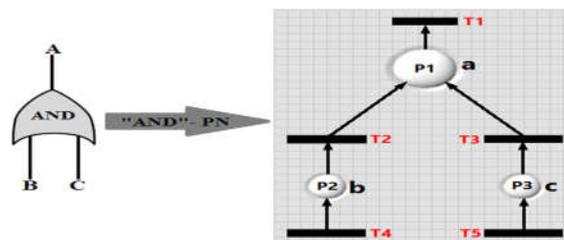


Fig.3. Transformation of the « AND » logical gate of the FT into the PN

3. Case study - Robotic cell

The object of our study is the advanced robotic cell machining of the Production Engineering Laboratory of the National School of Engineering in Tarbes. This cell shown in figure 4 comprises:

- KUKA KR120 robot, (b),
- Pneumatic grinder, (c),
- Tool adapted to the task to be performed, (d),
- Clamped piece in a vise arranged on a grooved table, (e),
- The vision system camera (National Instruments Image Processing Software), (f).



Fig.4. Robotic machining cell

3.1 Analysis by Fault Tree

The FT is a method of deductive analysis used also in the dependability, (Ghostine, 2008), (Noureddine et al., 2005). This is a method of analysing the reliability, availability, and security of more widely used systems, (Ruijters et al., 2019) (Dutuit et al., 2018). The diagnostic system we are developing they based on the learning base initiated from the Fault Tree (FT) of the robotic cell considered. This FT will be the centerpiece of our PN based strategy and they presented in Figure 5.

The analysis and research of the dreaded event in our FT highlights the non-conformity of the workpiece (a) in the robotic cell.

CABTREE software we used to build and process our fault trees. We have limited our study at two levels, which show the first elementary elements, as shown in figure 5:

- First level: Defective KUKA KR 120 Robot, (b); Faulty Grinder, (c); Faulty Tool, (d); Failing Piece, (e); Faulty Vision System Control (Camera), (f).
- Second level: Control Cabinet of a KUKA KR 120, (b1) ; Articulated Mechanical System, (b2) ; Wrong couple, (c1) Total failure, (c2) ; Break on the tool, (d1); Bad sharpening, (d2); Bad positioning of the piece, (e1); Non-conforming characteristics of the blank, (e2); Blurred Vision (False Results), (f1); Wrong Camera setting, Wrong treatments, (f2).

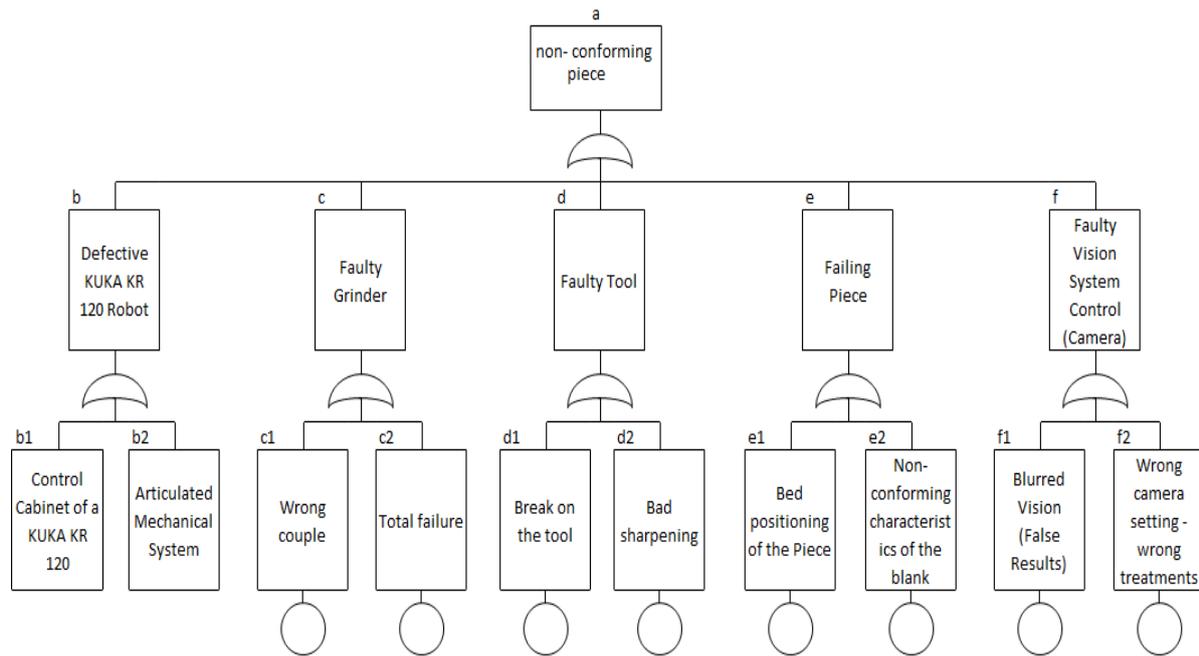


Fig.5. Fault Tree of the robot cell

3.2 Modeling FT-PN

We have applied our fault tree transformation technique (FT) in a Petri net (PN); we used the LabView software platform for modeling and simulation. LabView is based on a graphical development environment of «National Instruments», and it is used mainly for instrument control and industrial automation, (Decourt & Dordor, 2003) (Frey et al., 2007) and (Sagar & Narayana, 2014).

We have implemented our PN into a LabView state machine; the structure obtained they shown in figure 6 and 9. These figures represent the front panel (the user interface of the virtual instruments, in our system, Figures 6 and 9, of our PN that contains 16 places that are circular LEDs (Light Emitting Diodes) and 26 transitions. Each place presents an event of our FT, and contains their status: inputs (commands) and outputs (indicators) of the program. The controls and indicators are materialized by digital displays for 0 or 1.

A digital graph displays the data as impulses, binary number generator that will simulate the marking test (if there is a mark in a place, the signal displays the state 1 else the state 0). The X-axis is the state scale (places) and the Y-axis of the markings.

Our PN is limited to 1, alive, not resettable, not repetitive, graph of infinite markings. The initial marking, corresponding to Figure 6, is $m_0 = [000000000000000000]$.

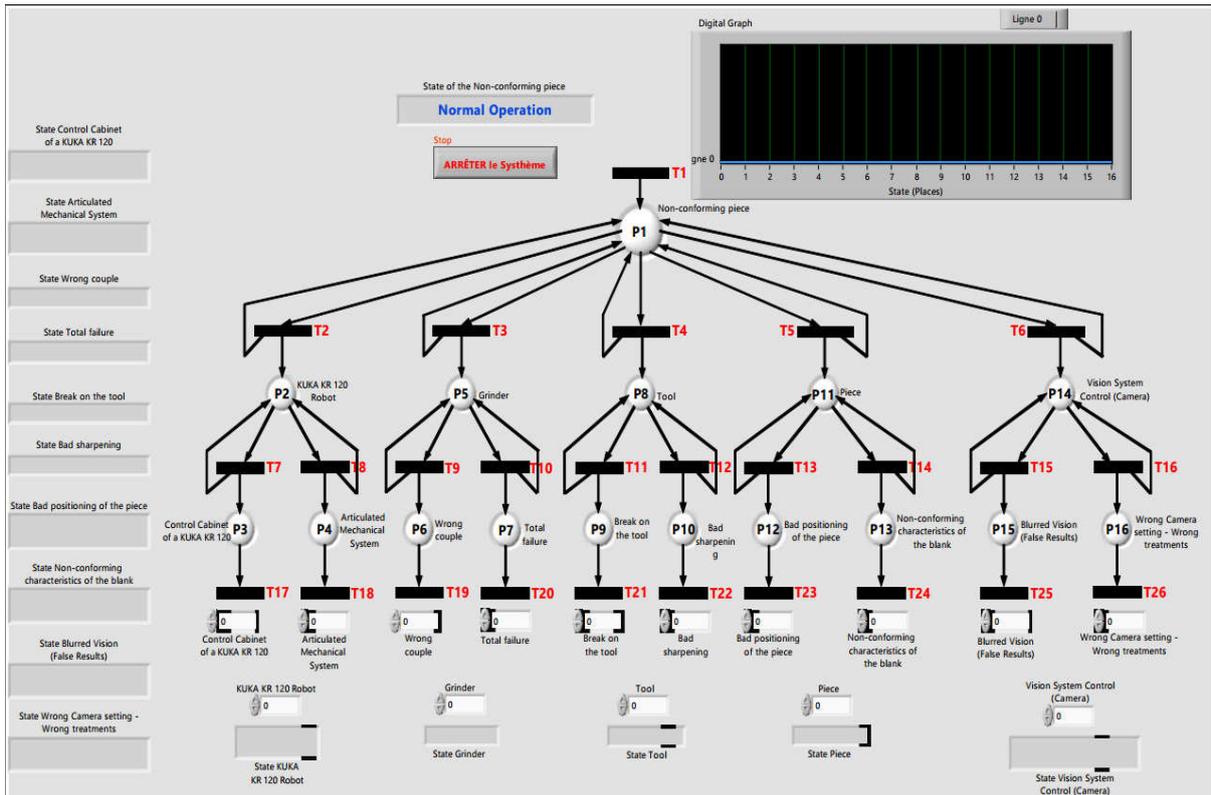


Fig.6. The front panel under LabView of the PN system - Modeling without fault

Block diagram, figure 7, represents a part of the application program developed in the form of a data flow diagram.

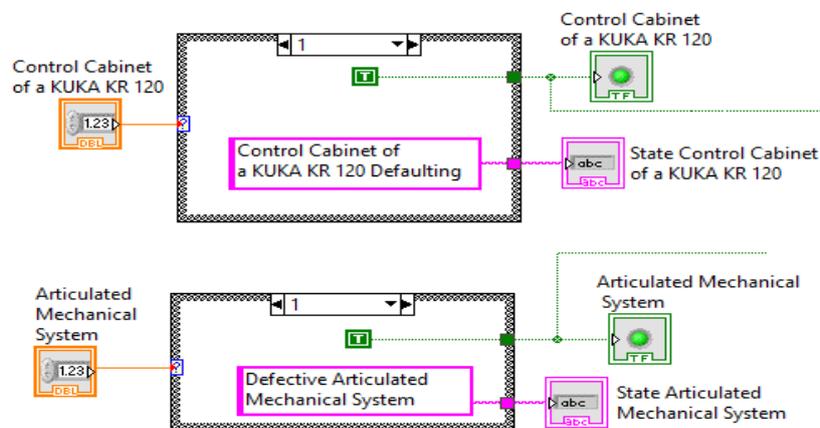


Fig.7. The block diagram of the LabView - part of the PN system

The following figure (figure 8), illustrates how a command and indicator materialized by digital displays for a state 0 or 1, in the block diagram of the LabView.

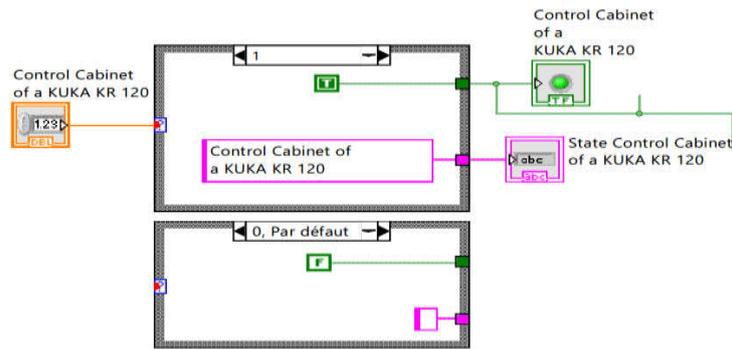


Fig.8. The block diagram of the LabView - example of a command

The transitions (T1, T2,, T26) of our petri net system are materialized by sensors. Table 1 describes the significance of each place.

Table 1. Significance of each place

Places	The significance of each place
P1	Non-conforming piece
P2	KUKA KR 120 Robot
P3	Control Cabinet of a KUKA KR 120
P4	Articulated Mechanical System
P5	Grinder
P6	Wrong couple
P7	Total failure
P8	Tool
P9	Break on the tool
P10	Bad sharpening
P11	Piece
P12	Bad positioning of the piece
P13	Non-conforming characteristics of the blank
P14	Vision System Control (Camera)
P15	Blurred Vision (False Results)
P16	Wrong Camera setting, Wrong treatments

4. Simulations and results

Simulations and results obtained are illustrated in figure 6. The corresponding marking is $m_1 = [1100000000110000]$; the state of the non-conforming piece is in operation with fault, the KUKA KR 120 robot is faulty in the first level of the FT, and the bad positioning of the piece, in the second level.

- The digital indicators of the KUKA KR 120 robot and the bad positioning of the piece is in 1: in « State KUKA KR 120 Robot » mark « Defective KUKA KR 120 Robot »; in « State Bad positioning of the piece » mark « Bad positioning of the piece ».
- So respectively, the place (P2) and the place (P12) are colored gray, and their signal is in the state 1.
- If the place (P2) faulty then (P1) faulty; If the place (P12) is faulty then (P11) is initially faulty and then (P1) is also faulty: If there is a faulty place, the triggered diagnostic process makes the failed system.

To do a deductive analysis for our PN, we proceed by crossing the transition (T1), and then the place (P1) has a token, if there is a token in the place (P1) then cross the transitions (T2) and (T5) directly. If (T2) is crossed then the places (P2) and (P1) have a token for each. Thus crossing (T5), the places (P11) and (P1) have a token for each. If the place (P11) has a token,

then (P12) has a token also after crossing the transition (T13): This diagnostic process is obtained through the return arcs constituting our petri net.

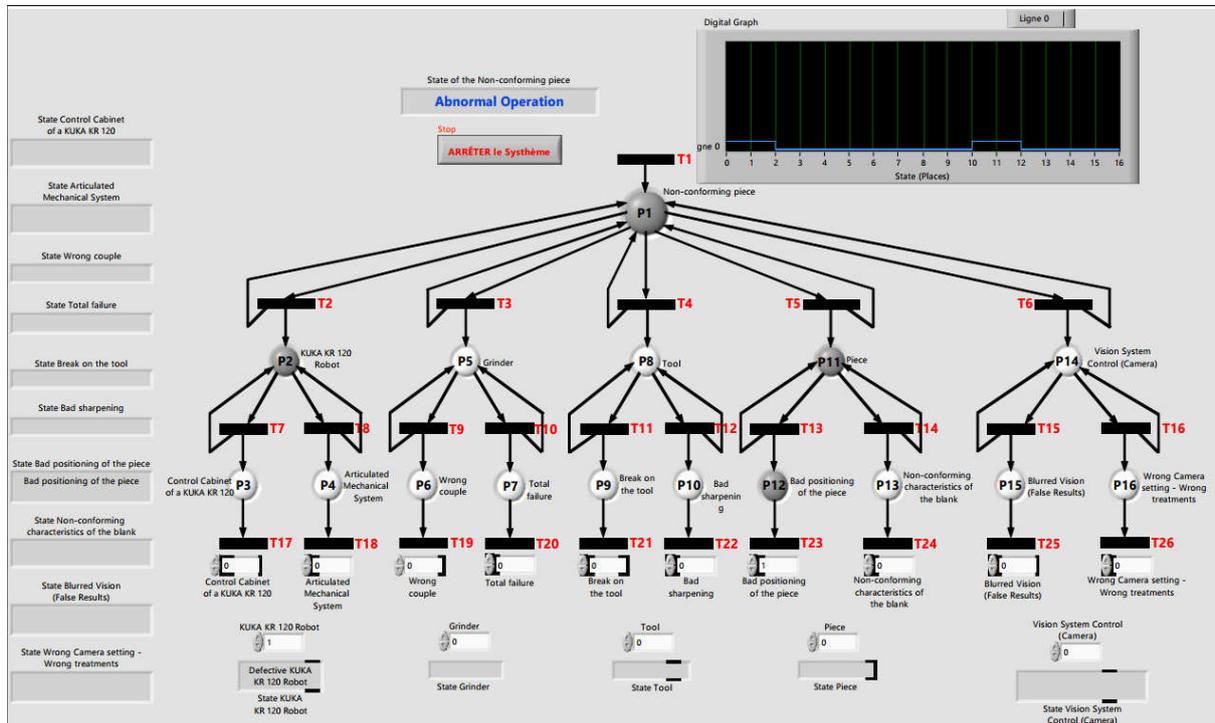


Fig.9. The front panel under LabView of the PN system - Modeling with fault

5. Conclusion

This work presents the development of a diagnostic strategy for a robot cell machining using a petri net generated from the realized knowledge of an FT on this cell, in the LabView environment. This first step, in the context of performing a diagnostic process on the cell, shows encouraging results.

In the following of this work, we plan to quantify the PN through the integration of failure rates in the FT on the one hand and, on the other hand to develop an interface in the environment of Solid Works to concretize the proposed approach on the robot cell machining.

References

- Boucherit, A. (2019). *Contribution à la Conception d'Architecture Logicielle Sûre des Systèmes Critiques Basés Agent*. Thèse de doctorat, Département d'Informatique, Option : Génie Logiciel, Université de Ferhat Abbas - Setif 1, Algérie.
- Brauer, W., Rozenberg, G., & Salomaa, A. (1985). *Petri Nets: An Introduction*. Épringer-Verlag, Berlin Heidelberg, Germany.
- David, R., & Alla, H. L. (1992). *Du Grafset aux réseaux de Petri*. 2eme Edition revue et augmentée, Livre, France.
- Decourt, R., & Dordor, P. (2003). *Introduction à LabVIEW*. Édition: 2003 - 2004.
- Dutuit, Y., Signoret, J. P., Folleau, C., & Thomas, P. (2018). *Détermination et calcul de la fréquence d'occurrence d'une coupe minimale – Un bref état de l'art –*. 21e Congrès de Maîtrise des Risques et Sûreté de Fonctionnement $\lambda\mu 21$, Reims, France.
- Frank, C., & Olivier, H. R. (2003). Traduction structurelle des Réseaux de Petri Temporels vers les Automates Temporisés. *Hermès Science, Éditeur: 4ème Colloque Francophone Sur La Modélisation Des Systèmes Réactifs (MSR 03)*, 54.
- Frey, D., Degryse, P.-A., & Amalberti, J.-L. (2007). *Le langage de programmation Labview*. IUT1 Grenoble Département GEIII, France.

- Ghostine, R. (2008). *Influence des fautes transitoires sur la fiabilité d'un système commandé en réseau*. Thèse de doctorat, Institut National Polytechnique de Lorraine, Spécialité Automatique, Traitement du signal et Génie Informatique, France.
- Haiouni, H. (2010). *Approche mixte de modélisation par Réseaux de Petri et SMA*. Mémoire Magister, Département d'Informatique, Option : Intelligence Artificielle, Université Mentouri de Constantine, Algérie.
- Isermann, R., & Ballé, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5), 709–719.
- Mahdaoui, R. (2013). *Contribution à la surveillance dynamique des systèmes de production évolutifs par les systèmes Neuro- Flous Temporels*. Thèse de doctorat, Université de Batna 2, Département de Génie Industriel, Algérie.
- Morère, Y. (2002). *Cours de réseau de Petri*.
- Noureddine, R., Benamar, A., & Noureddine, F. (2005). Intégration d'une technique de maintenance prédictive de l'outil de coupe dans une cellule d'usinage robotisée. *Colloque International Francophone Sur Les Performances et Nouvelles Technologies En Maintenance (PENTOM 2005), Actes PENTOM 2005, Presses Universitaires de Valenciennes, Marrakech, Maroc, N°6*, 133–142.
- Palluat, N. (2006). *Méthodologie de surveillance dynamique à l'aide des réseaux neuro-flous temporels*. Thèse de Doctorat, Université de Franche-Comté, France.
- Palluat, N., Racoceanu, D., & Zerhouni, N. (2004). Diagnosis aid system using a neuro-fuzzy approach. *Advances in Maintenance and Modeling, Simulation and Intelligent Monitoring of Degradation*.
- Petri, C. A. (1962). *Kommunikation mit Automaten*. Thèse, Institut für Instrumentelle - Mathematik, Bonn, Germany.
- Racoceanu, D. (2006). *Contribution à la Surveillance des Systèmes de Production en Utilisant l'Intelligence Artificielle*. Thèse de doctorat, Université de Franche-Comté, France.
- Ruijters, E., Budde, C. E., Nakhaee, M. C., Stoelinga, M., Bucur, D., Hiemstra, D., & Schivo, S. (2019). *FFORT: A benchmark suite for fault tree analysis*. The 29th European Safety and Reliability Conference, Singapore.
- Sagar, S. C., & Narayana, M. (2014). Ambulance Controlled Traffic System Using RFID Technology with LabVIEW Simulation. *International Journal Of Engineering And Computer Science*, 3, 6048–6052.