



HAL
open science

SURVEILLANCE ET DIAGNOSTIC PAR LE FLUX D'ATELIERS DE PRODUCTION CYCLIQUE

Olfa Fakhfakh

► **To cite this version:**

Olfa Fakhfakh. SURVEILLANCE ET DIAGNOSTIC PAR LE FLUX D'ATELIERS DE PRODUCTION CYCLIQUE. Automatique. Ecole Centrale Lille, 2015. Français. NNT : . tel-01742476

HAL Id: tel-01742476

<https://hal.science/tel-01742476v1>

Submitted on 25 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE DE DOCTORAT

Présentée en vue de l'obtention des titres de

**DOCTEUR DE L'ÉCOLE NATIONALE DES SCIENCES DE
L'INFORMATIQUE**

Spécialité Informatique

&

DOCTEUR DE L'ÉCOLE CENTRALE DE LILLE

Spécialité Automatique, Génie Informatique, Traitement du Signal et Image

par

Olfa FAKHFAKH

Ingénieur en Informatique de l'ENSI

SURVEILLANCE ET DIAGNOSTIC PAR LE FLUX D'ATELIERS DE PRODUCTION CYCLIQUE

Réalisée aux laboratoires

Unité de Recherche Modeling of Automated
Reasoning Systems (MARS), Tunisie

Centre de Recherche en Informatique, Signal,
et Automatique de Lille, France



Soutenue le 02 décembre 2015
devant le jury:

Pr. Moncef TAGINA, *Professeur à l'ENSI, Tunisie*

MdC. Pascale MARANGE, *Maître de Conférences à l'Université de Nancy, France*

Pr. Pierre-Alain YVARS, *Professeur à l'Institut Supérieur de Mécanique de Paris*

MdC. Sadok BOUAMAMA, *Maître de Conférences à l'ENSI, Tunisie*

Pr. Ouajdi KORBAA, *Professeur à ISITCom Hammem Sousse, Tunisie*

Pr. Armand TOGUYENI, *Professeur à Ecole Centrale de Lille, France*

Examineur

Examineur

Rapporteur

Rapporteur

Directeur de thèse

Directeur de thèse

À la mémoire de nos soldats martyrs qui ont donné de leur vie pour assurer notre sécurité,

À tous ceux qui ont souhaité ma réussite,

Je dédie ce travail, en guise d'amour et de reconnaissance.

Remerciements

Ce travail de recherche a été préparé dans le cadre d'une co-tutelle entre l'Ecole Nationale des Sciences de l'Informatique, Unité de Recherche Modeling of Automated Reasoning Systems (MARS) et l'Ecole Centrale de Lille, Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL).

Ce mémoire marque une étape importante dans ma vie. Je voudrais remercier ici toutes les personnes qui m'ont accompagnée pour accomplir l'un des plus importants projets professionnels de ma vie.

Je tiens à exprimer toute ma gratitude et mes plus vifs remerciements envers mes directeurs de thèse. Je remercie le Professeur Ouajdi KORBAA pour la qualité de l'encadrement qu'il m'a assuré, son intérêt envers mon travail, la confiance qu'il m'a accordée et surtout ses encouragements continuels. Son dévouement pour son travail, son professionnalisme et ses qualités humaines ont beaucoup contribué dans la réalisation de ce travail et aussi à mon instrumentation aussi bien sur le plan professionnel que personnel. Je remercie également le Professeur Armand TOGUYENI d'abord pour m'avoir invité et accueilli à l'EC-Lille, ensuite pour l'aide scientifique qu'il m'a toujours apporté et enfin pour sa disponibilité, ses qualités professionnelles et ses remarques pertinentes. J'espère qu'en regardant ce travail, j'aurai rempli toutes les attentes de mes directeurs.

Je voudrais remercier Monsieur Moncef TAGINA, Professeur à l'ENSI, Tunisie, pour l'honneur qu'il m'a fait en acceptant d'examiner mon travail. Je tiens à remercier Monsieur Pierre-Alain YVARS, Professeur des Universités, SUPMECA, France, d'avoir accepté avec Monsieur

Sadok BOUAMAMA, Maître de Conférences, ENSI, Tunisie, d'étudier mes travaux et d'en être les rapporteurs ainsi que pour l'intérêt et l'attention qu'ils ont accordés à cette étude. Je remercie également Madame Pacale MARANGE, Maître de Conférences, Université de Nancy, France de m'avoir fait l'honneur d'examiner mon travail et d'être membre du jury.

Je ne laisserai pas cette occasion passer sans exprimer ma reconnaissance envers tous les enseignants et les personnels de l'ENSI qui m'ont aidé de loin ou de près durant ma période d'enseignement dans ma chère école pour pouvoir avancer dans mes travaux de recherche. Une pensée particulière s'adresse au Pr. Moncef pour son aide précieuse et son soutien moral.

Je remercie chaleureusement mes collègues et mes amis de l'unité de recherche MARS et de l'ENSI et plus particulièrement Med Amin, Imtief, Nessrin, Aroua, Mariem, Raoudha, Faten... pour vos encouragements et toutes nos discussions fructueuses.

Je remercie également mes collègues à la Faculté des Sciences de Gabes et particulièrement le chef de département Fathi pour sa coopération et mes amies Wafa, Raja, Nour et Ons pour l'ambiance sympathique qu'elles ont réussi à instaurer. Je ne saurais oublier de transmettre mes pensées chaleureuses et fraternelles à mon petit entourage Sameh, Leila, Zouheir, Sahar, Nadia, Taha.

Une reconnaissance venant du plus profond du cœur est adressée à tous les membres de ma famille, mes parents, mes frères et ma belle sœur Wafa. Mes chers parents merci pour vos prières, votre irremplaçable et inconditionnel soutien, votre confiance et vos encouragements, tout ce que je fais, c'est pour que vous soyez fier de moi. Cette thèse est aussi la vôtre. Merci Maman et Papa

Table des matières

Introduction générale	1
I Etat de l'art sur le diagnostic des défauts	5
1 Concepts fondamentaux du diagnostic	6
1.1 Terminologies	6
1.2 Surveillance, diagnostic et supervision	8
2 Cadre d'étude	12
2.1 Systèmes Automatisés de Production	12
2.2 Systèmes Flexibles de Production Manufacturière	13
2.3 Commande cyclique d'un SFPM	15
2.3.1 Ordonnancement des systèmes de production	15
2.3.2 Principaux concepts de l'ordonnancement cyclique	16
2.4 Présentation d'exemples illustratifs	18
2.4.1 SFPM à plusieurs types de pièces	18
2.4.2 SFPM à plusieurs en-cours par produit	19
3 Méthodes de diagnostic	20
3.1 Méthodes de diagnostic sans modèles	21
3.2 Méthodes de diagnostic avec modèles	21
3.2.1 Méthodes à base de modèles qualitatifs	22
3.2.2 Méthodes à base de modèles quantitatifs	22
3.3 Méthodes indirectes	23
3.3.1 Approches quantitatives	23

3.3.2	Approches qualitatives	23
3.4	Méthodes incrémentales	24
4	Positionnement des travaux dans le contexte scientifique	25
4.1	Synthèse des méthodes de diagnostic	25
4.2	Problématique et orientation	26
5	Conclusion	27
II Méthode de diagnostic indirect par arbre de résolution basé sur la technique du chaînage double		29
1	Hypothèses de diagnostic	30
2	Modélisation de l'aspect cyclique de l'ordonnancement	31
2.1	Choix conceptuel de la modélisation cyclique	32
2.2	Positionnement de l'approche dans le contexte de l'étude	33
3	Démarche de résolution	34
3.1	Construction de PERT cyclique	35
3.2	Génération des hypothèses par chaînage arrière	36
3.3	Validation des hypothèses par chaînage avant	41
3.4	Modélisation de l'induction du retard	42
3.4.1	Calcul de la marge	43
3.4.2	Calcul du retard	43
4	Diagnostic par chaînage double	44
4.1	Tests et résultats du diagnostic	44
4.2	Analyse du principe de résolution	46
5	Efficacité du diagnostic	47
6	Conclusion	48
III Approche par contraintes pour le diagnostic incrémental		49
1	Programmation par contraintes	51
2	Paramètres du diagnostiqueur	52
2.1	Opération	52
2.2	Données	53
2.3	Variables	53
3	Contraintes du diagnostiqueur	53
3.1	Types de retards	54
3.1.1	Retard propre	55

3.1.2	Retard induit	55
3.1.3	Retard total	55
3.2	Dates de fin des opérations	56
3.3	Contraintes de précédences entre les opérations	56
3.4	Contraintes d'inductions de retards	58
3.4.1	Contrainte liée au retard propre	58
3.4.2	Contrainte liée au retard induit	59
3.4.3	Contrainte de défaillance unique	60
4	Diagnostic incrémental	60
4.1	Mécanisme de résolution	61
4.2	Taille du problème	62
5	Application du diagnostic à un SFPM sous une commande cyclique	63
5.1	Choix de l'outil de résolution	63
5.2	Description du problème pour le modèle proposé	64
5.3	Tests et résultats du diagnostic indirect et incrémental	64
5.3.1	Scénario 1	65
5.3.2	Scénario 2	67
5.4	Analyses qualitatives des résultats	70
6	Etude de performance du diagnostic	71
6.1	Evaluation du diagnostic d'un seul cycle	72
6.2	Evaluation du diagnostic incrémental complet	74
6.3	Stratégies de recherche	76
6.4	Comparaison avec la littérature	79
7	Conclusion	82

IV Etude de la diagnosticabilité d'une commande cyclique **84**

1	Etude de la diagnosticabilité	85
1.1	Principe et objectifs	85
1.2	Diagnosticabilité des SED	86
1.2.1	Classification selon la modélisation des défaillances	86
1.2.2	Classification selon l'outil de représentation pour le diagnostic	87
1.2.3	Classification selon la structure de la méthode de diagnostic	88
1.2.4	Autres approches formelles	90
2	Proposition d'une nouvelle méthode de diagnosticabilité d'un ordonnancement cyclique	91

2.1	Définition de la diagnosticabilité d'un SFPM	91
2.2	Vérification de la détectabilité	92
3	Application de la diagnosticabilité sur des cas d'étude	94
3.1	Etude de la détectabilité d'un SFPM à plusieurs types de pièce	94
3.2	Etude de détectabilité pour un SFPM à plusieurs en-cours par produit	96
4	Vers un ordonnancement diagnosticable	99
4.1	Solutions de diagnosticabilité	99
4.2	Propriétés d'un ordonnancement diagnosticable	100
5	Conclusion	101
	Conclusion générale et perspectives	102
	Bibliographie	105

Liste des figures

I.1	Les différentes fonctions du Contrôle/Commande d'un SFPM [Toguyeni <i>et al.</i> , 2002]	8
I.2	Les différents types de surveillance des SAP	10
I.3	Structure d'un <i>SAP</i> ¹	12
I.4	Productivité, Flexibilité et Coût [Kermad, 1996]	14
I.5	Diagramme de GANTT dual d'un SFPM	19
I.6	Ordonnancement d'un SFPM à plusieurs en-cours par produit	20
II.1	Processus de diagnostic indirect par analyse des flux	35
II.2	Graphe potentiel étape PERT cyclique	37
II.3	La sémantique des nœuds et des arcs dans un arbre de résolution	39
II.4	Propagation des contraintes temporelles entre des nœuds adjacents	39
II.5	Arbre d'hypothèses généré par chaînage arrière	45
II.6	Validation de l'hypothèse H2 par chaînage avant	46
III.1	Les caractéristiques des dates et des durées d'une opération	54
III.2	Exemple de contraintes de précédences	57
III.3	Contraintes liées au retard induit	59
III.4	Système Flexible de Production Manufacturière [Ly <i>et al.</i> , 2000]	80
III.5	Ordonnancement cyclique d'un SFPM	81

Liste des tableaux

I.1	Classement des méthodes de diagnostic à base de modèles des SED	25
II.1	Les paramètres d'un nœud du PERT	38
III.1	Les données du modèle relatives à une opérations donnée	53
III.2	Les variables du modèle relatives à une opérations donnée	54
III.3	Ordre partiel d'exécution des opérations des produits attendus en sortie	65
III.4	Résultats d'exécution du scénario 1	67
III.5	Retards pronostiqués du scénario 1	68
III.6	Aspect incrémental de résolution	68
III.7	Résultats d'exécution du scénario 2	69
III.8	Retards pronostiqués du scénario 2	70
III.9	Comparaison entre Approche 1 et Mono-cycle Approche 2	73
III.10	Evaluation de l'approche incrémentale	75
III.11	Stratégies de recherche	78
III.12	Aspect incrémental de résolution	82
IV.1	Détectabilité de l'opération op_{21}^β de la machine M_3	95
IV.3	Etude de la détectabilité d'un SPFM à plusieurs types de pièce	95
IV.2	Détectabilité de l'opération op_{51}^β de la machine M_1	96
IV.4	Détectabilité de l'opération op_{11}^β de la machine M_1	97
IV.5	Détectabilité de l'opération op_{12}^β de la machine M_2	98
IV.6	Opération non détectable	98
IV.7	Etude de la détectabilité d'un SPFM à plusieurs en-cours	99

Liste des Algorithmes

1	Fonction défaillance possible	40
2	Chaînage arrière	41
3	Chaînage avant	42
4	Diagnostic incrémental	62
5	Vérification de la détectabilité d'une opération	93

Introduction générale

Contexte & Problématique

Malgré le progrès et les avancées technologiques dans la commande des Systèmes Flexibles de Production Manufacturière (SFPM), la supervision de procédés reste une tâche très importante qui est encore en grande partie une activité manuelle, exécutée par les opérateurs, notamment lorsqu'il s'agit de répondre aux événements anormaux [Combacau *et al.*, 2000]. Cette activité peut selon la nature et la criticité des anomalies ou dysfonctionnements, avoir un impact économique, environnemental et de sécurité plus ou moins significatif, non seulement au niveau des équipements mais aussi des personnes [Morrison et Upton, 1994]. D'autant plus que le fait de gérer une grande quantité d'informations et d'avoir besoin d'agir vite peut mener les opérateurs à prendre des décisions incorrectes, dégradant encore plus la situation. De ce fait, les systèmes de supervision qui intègrent des outils de surveillance, de détection et de diagnostic sont nécessaires afin de fournir à l'opérateur des critères suffisants pour la prise de décision [Ding, 2008]. Dans ce contexte, la détection d'anomalies occupe une place de plus en plus importante dans la conception des systèmes de production cyclique. Ce besoin a conduit les chercheurs à concevoir et développer une variété de techniques de diagnostic dans différents domaines.

L'étude de l'état de l'art a permis de montrer que les méthodes de diagnostic des systèmes automatisés de production peuvent être classées en deux catégories. Les méthodes directes [Sampath *et al.*, 1995] sont basées sur des capteurs de surveillance dédiés permettant de suivre directement les paramètres de ressources [Gentil, 2007]. Les méthodes indirectes sont basées sur des analyses quantitatives [Toguyeni et Korbaa, 2005] ou qualitatives [Telmoudi *et al.*, 2008] de la production. Les méthodes directes sont plus réactives mais nécessitent souvent une instru-

mentation spécifique dont on ne dispose pas toujours quand la surveillance est mise en place à posteriori. A l’opposé, les méthodes indirectes sont moins réactives car elles nécessitent que la défaillance influe de manière importante sur la production (de manière qualitative ou quantitative). L’intérêt des méthodes indirectes est justifié par le fait que l’utilisation d’un grand nombre de capteurs de surveillance supplémentaires, dans les approches directes, diminue notablement la disponibilité du système. La difficulté majeure de ces méthodes réside dans le fait qu’il existe peu d’observations du comportement interne du système. Cela a pour conséquence un manque de sensibilité ne permettant pas de détecter certaines défaillances de faibles amplitudes. Nous verrons dans ce travail que les marges entre les opérations peuvent rendre indétectables des défaillances progressives de faibles amplitudes.

Objectifs

Ces travaux de thèse s’inscrivent dans le cadre du diagnostic indirect des SFPM. Ils sont consacrés au développement d’un outil de diagnostic en ligne d’un SFPM par analyse quantitative de la production. Le principe de diagnostic indirect choisi, consiste à comparer le flux de production réel par rapport au flux prévisionnel. Il s’agit d’analyser tout écart entre le comportement défaillant par rapport au comportement normal du système. Le fait d’effectuer cette comparaison à la fin de la période de production entraîne un retard dans la détection des écarts. Cette stratégie est contraignante dans le contexte du diagnostic en ligne où la rapidité du résultat est une exigence. L’objectif de ce travail est de diagnostiquer les défauts du SFPM le plus tôt possible en s’appuyant sur une méthode de diagnostic indirecte. Afin d’augmenter la réactivité de la méthode de diagnostic adoptée, nous optons pour une commande cyclique qui permet d’avoir des points d’observation du systèmes supplémentaires à la fin de chaque cycle de production.

Dans ce cadre, l’objectif de notre travail de thèse est de proposer une solution de diagnostic indirect et en ligne d’un SFPM sous ordonnancement cyclique. La fonction de diagnostic à proposer consiste à localiser et identifier les causes premières des défaillances qui engendrent la diminution des performances du procédé. En outre, l’approche de diagnostic devra mettre à jour les résultats du diagnostic lorsque de nouvelles observations seront disponibles à chaque cycle de production. Nous nous positionnerons à la sortie du système pour observer le flux de production.

Outre le diagnostic, ces travaux de thèse s'intéressent à l'étude de la diagnosticabilité de l'ordonnancement d'un SFPM afin de garantir la qualité du diagnostic et prévoir la capacité d'une approche à diagnostiquer un SFPM. Dans le cadre du diagnostic indirect, la diagnosticabilité doit permettre de savoir si les points d'observabilité prévus pour le système sont suffisants pour détecter, localiser et identifier au bout d'un temps fini les causes premières de défaillances.

Plan du manuscrit de thèse

Le présent manuscrit de thèse s'articule autour de quatre chapitres.

Le premier chapitre introduit dans un premier temps les concepts de base du diagnostic des défauts des systèmes automatisés de production. Ensuite, les domaines d'application et le cadre d'étude sont introduits tels que les Systèmes Automatisés de Production (SAP) et les Systèmes Flexibles de Production Manufacturière (SFPM). Le choix d'une commande cyclique et déterministe est expliqué et justifié, en se basant sur l'optimisation d'un ensemble de mesures. Ce chapitre présente après l'état de l'art relatif aux différentes méthodes de diagnostic. Dans ce contexte les principaux travaux de référence se déclinent en plusieurs catégories qui sont répertoriées selon la nature de l'information disponible (qualitative ou quantitative) et la disposition des points d'observabilité (directes ou indirectes). La dernière partie de ce chapitre, introduit l'idée de la minimisation de l'utilisation des capteurs de surveillance tout en assurant une bonne qualité de la fonction de diagnostic en ligne.

Le deuxième chapitre présente la première approche de diagnostic indirect par arbre de résolution basée sur la technique de chaînage double. Ce chapitre décrit d'abord, le mécanisme de résolution de la première contribution qui se base sur l'analyse quantitative des flux de production. La mise en œuvre de l'approche a permis de tirer un certain nombre d'enseignements illustrant ses limites et de justifier le besoin d'une approche incrémentale de diagnostic pour résoudre le problème de convergence.

Le troisième chapitre présente la deuxième approche de diagnostic qui se base sur une modélisation mathématique et une mise en œuvre par programmation par contraintes. Cette approche de diagnostic permet de répondre aux objectifs de la thèse et de combler les limites de la première approche. Dans ce cadre, les paramètres et les contraintes du diagnostic relatives

aux précédences des opérations et celles relatives à l'induction des retards, sont introduits. La modélisation de l'aspect cyclique de l'ordonnancement est détaillée permettant de mettre en œuvre les principes de diagnostic incrémental et du pronostic. A la fin de ce chapitre, la résolution et le test du modèle sont détaillés. Ils concernent deux systèmes de production : un SFPM à plusieurs types de pièce et un SFPM à plusieurs en-cours par produit où plusieurs gammes opératoires identiques sont en cours de production simultanément.

Le quatrième chapitre concerne l'étude de la diagnosticabilité d'une commande cyclique d'un SFPM. Dans ce cadre, nous proposons d'abord une classification de méthodes d'analyse de la diagnosticabilité. Ensuite une proposition d'une nouvelle méthode d'analyse de diagnosticabilité est présentée permettant la vérification de la détectabilité d'un défaut dans un SFPM. La dernière partie de ce chapitre présente les résultats de mise en œuvre et de test de la solution proposée.

Le présent manuscrit est clôturé par une conclusion et quelques perspectives à explorer en continuité de cette thèse.

Chapitre I

Etat de l'art sur le diagnostic des défauts

Sommaire

1	Concepts fondamentaux du diagnostic	6
1.1	Terminologies	6
1.2	Surveillance, diagnostic et supervision	8
2	Cadre d'étude	12
2.1	Systemes Automatisés de Production	12
2.2	Systemes Flexibles de Production Manufacturière	13
2.3	Commande cyclique d'un SFPM	15
2.4	Présentation d'exemples illustratifs	18
3	Méthodes de diagnostic	20
3.1	Méthodes de diagnostic sans modèles	21
3.2	Méthodes de diagnostic avec modèles	21
3.3	Méthodes indirectes	23
3.4	Méthodes incrémentales	24
4	Positionnement des travaux dans le contexte scientifique	25
4.1	Synthèse des méthodes de diagnostic	25
4.2	Problématique et orientation	26
5	Conclusion	27

Ce premier chapitre présente une synthèse de l'état de l'art sur le diagnostic des défauts des systèmes automatisés de production. Il se compose de quatre parties. Dans la première nous introduisons les concepts de base liés à la surveillance et au diagnostic à travers la revue de quelques terminologies. Dans la deuxième partie de ce chapitre, nous présentons le type des systèmes vers lesquels nous souhaitons orienter nos travaux. Ce sont les systèmes automatisés de production, et plus particulièrement les systèmes flexibles de production manufacturière. Dans la troisième partie, nous exposons des travaux réalisés autour de la thématique étudiée. Nous proposons à cet effet une classification non exhaustive de ces travaux selon la nature de l'information disponible et la disposition des points d'observabilités. Une critique de l'existant et une synthèse de la problématique seront présentées dans la dernière partie de ce chapitre. L'objectif de ce chapitre est de voir globalement le domaine du diagnostic des défauts et de se positionner par rapport à l'existant afin d'introduire le diagnostic indirect des SFPM autour duquel sera développé le deuxième chapitre.

1 Concepts fondamentaux du diagnostic

Cette partie introduit les concepts de base du diagnostic des défauts des systèmes automatisés de production.

1.1 Terminologies

Cette partie vise à rappeler les terminologies utilisées pour la surveillance et le diagnostic, rencontrées dans la littérature et retenues dans ce mémoire. Les définitions que nous présentons sont extraites des références suivantes : [Villemeur, 1988], [Toguyeni, 1992], [Combacau *et al.*, 2000], [Philippot, 2006] et [Ding, 2008].

- **Défaut** ou **Faute** (Fault) : Un défaut est considéré comme un écart du comportement normal qui n'empêche pas le système de remplir sa fonction. Il s'exprime par une déviation d'une propriété du système pouvant présager d'une défaillance à venir.
- **Défaillance** (Failure) : Une défaillance est une anomalie fonctionnelle qui empêche partiellement ou totalement l'aptitude d'un système à remplir sa fonction. Une défaillance conduit à l'existence d'un défaut, puisqu'elle aboutit à un écart entre la caractéristique mesurée et la caractéristique spécifiée. Inversement, un défaut n'induit pas nécessairement une défaillance. En effet, le système peut conserver son aptitude à assurer une fonction

requis, si les défauts qui l'affectent n'ont pas d'impacts significatifs sur la fonction. Si une défaillance peut conduire à une suspension de l'exécution de la fonction principale du système, ce dernier est déclaré en état de panne [Zwingelstein, 1995].

- **Panne** (Break-down) : Une panne représente la conséquence d'une défaillance dans la réalisation du fonctionnement du système. Elle provoque un arrêt complet du système dans un état où il devient incapable d'assurer le service spécifié. Ainsi, une panne résulte toujours d'une défaillance. Deux types de pannes peuvent être distingués :
 - **Les pannes intermittentes** : sont présentes uniquement pendant un laps de temps déterminé. Le système peut retrouver son fonctionnement nominal après l'occurrence de la panne sans intervention extérieure. Une panne intermittente est généralement le résultat d'une dégradation partielle et progressive d'un composant du système, pouvant aboutir à une panne permanente.
 - **Les pannes permanentes** : une fois qu'une panne permanente s'est produite, elle persiste au cours du temps jusqu'à une intervention de maintenance et nécessite une action de réparation.

- **Symptôme** : Un symptôme correspond à une ou plusieurs observations révélant un dysfonctionnement. Il s'agit d'un effet qui est la conséquence d'un comportement anormal.

- **Observation** : Une observation est une information obtenue à partir du comportement ou du fonctionnement réel du système. C'est la seule information connue par le module de surveillance pour la détection.

- **Observabilité** : Un événement est considéré observable s'il est issu des capteurs physiques ou des actionneurs d'un procédé. Cet événement peut être observé par l'environnement du système et en particulier par le module de surveillance. Tandis que les événements non observables correspondent aux événements internes du système et aux défauts dont les occurrences ne peuvent pas être directement observées. Quelquefois, l'occurrence des événements de fautes est déduite indirectement, en s'appuyant sur le comportement observable qui suit l'occurrence de tels événements. La détection et l'isolation des fautes sont des fonctions du diagnostic. Cette notion a été introduite par [Lin et Wonham, 1988] et reprise dans [Sampath *et al.*, 1995].

1.2 Surveillance, diagnostic et supervision

Le taux d'apparition de pannes dans un système automatisé de production (SAP) justifie la nécessité de suivre un SAP par des systèmes de surveillance. Ces derniers permettent d'alerter l'opérateur d'une défaillance et de remonter l'information du diagnostic établi, afin de pouvoir décider à temps des actions correctives [Morrison et Upton, 1994]. La figure I.1 ; inspirée des travaux de [Toguyeni, 1992], [Toguyeni *et al.*, 2002] ; présente l'architecture fonctionnelle du système de contrôle/commande d'un SAP.

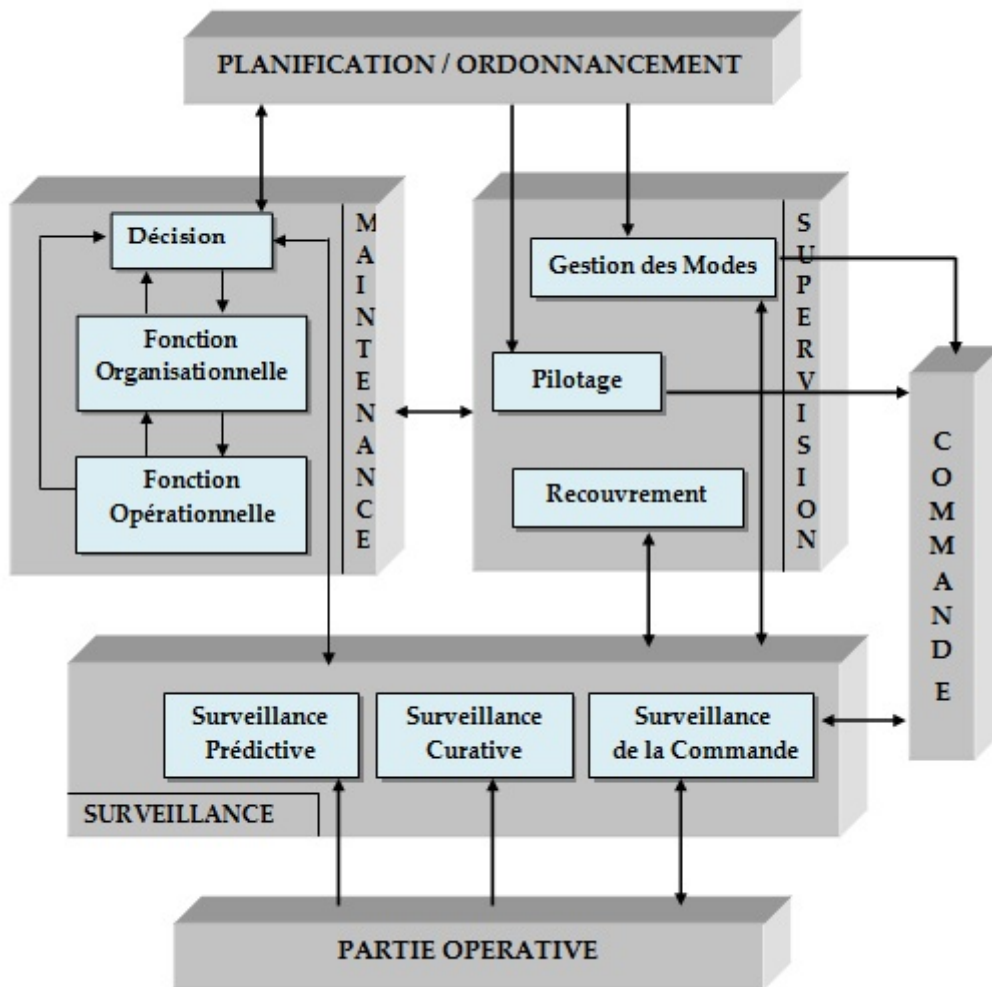


Figure I.1 – Les différentes fonctions du Contrôle/Commande d'un SFPM [Toguyeni *et al.*, 2002]

- **Surveillance** : La surveillance des systèmes industriels consiste à identifier l'état du système à partir des informations délivrées par des capteurs. Son objectif est d'informer

l'opérateur de supervision voire la commande, de l'occurrence d'un ou plusieurs événements susceptibles d'affecter le bon fonctionnement du système. La surveillance recueille les informations provenant du système de la commande et des données du procédé pour déterminer l'état présent du système. Elle a un rôle passif vis-à-vis du système de commande et du procédé [Combacau *et al.*, 2000], limité au traitement des données. Nous pouvons spécifier deux types de surveillance. La figure I.2 résume les différents types de surveillance.

- **Surveillance de la commande** [Combacau, 1991], [Lee, 2006] : Elle a pour objectif de s'assurer que la commande émise est cohérente (en parfaite adéquation) avec l'état du procédé. Une telle surveillance permet d'éviter la propagation de défaillance par la commande elle-même.
- **Surveillance du procédé** [Brunet *et al.*, 1990], [Toguyeni, 1992] et [Ly *et al.*, 2000] : Elle permet le suivi, la détection, le diagnostic et le pronostic des défaillances du procédé.

Dans la suite nous nous intéressons à la surveillance des pannes permanentes du procédé. Elle peut être classée en deux types, préventive ou corrective.

- **Surveillance préventive** [Nabli, 2000], [Ly *et al.*, 2000] : Elle a pour objectif d'éviter l'occurrence de l'état de panne. Elle concerne les défaillances progressives. Ce type de surveillance s'appuie sur l'étude des paramètres représentatifs reflétant l'état de procédé afin d'assurer le suivi de l'évolution, la détection des dérives, l'identification de l'origine de la dérive et finalement l'analyse de la cause de cette dérive.
- **Surveillance corrective** [Toguyeni, 1992], [Berruet *et al.*, 2000] : Elle permet de maintenir une disponibilité élevée du système et de conserver son caractère opérationnel face à la présence de perturbations de type panne par la recherche d'un nouvel état suite à un ensemble d'actions correctives. L'objectif de ce type de surveillance est de détecter et de diagnostiquer les anomalies du procédé afin de permettre la mise en œuvre d'actions correctrices permettant la poursuite de la production en mode dégradé et la maintenance corrective du système.

Dans la suite nous nous intéressons à la surveillance corrective, qui peut être directe ou indirecte.

- **Surveillance directe** [Sampath *et al.*, 1995] : Elle permet de suivre directement les paramètres des ressources du procédé. Une technique de surveillance est dite directe

si elle s'appuie sur des capteurs de surveillance physique dédiés et des modèles de surveillance du système. Elle nécessite une instrumentation spécifique du procédé de manière à permettre la détection et le diagnostic.

- **Surveillance indirecte** [Toguyeni et Korbaa, 2005], [Telmoudi, 2011] : La surveillance indirecte se base sur la surveillance des ressources à partir des perturbations qu'elles engendrent sur les produits. Ces perturbations peuvent se traduire par une diminution significative du flux de certains produits voire par une modification de leur qualité. Elle se base sur des analyses quantitatives ou qualitatives des produits. Les techniques indirectes exploitent dans certaines étapes de la surveillance, des capteurs « logiques ». Ces capteurs logiques sont conçus à partir d'algorithmes de fusion de données ou de corrélation d'événements.

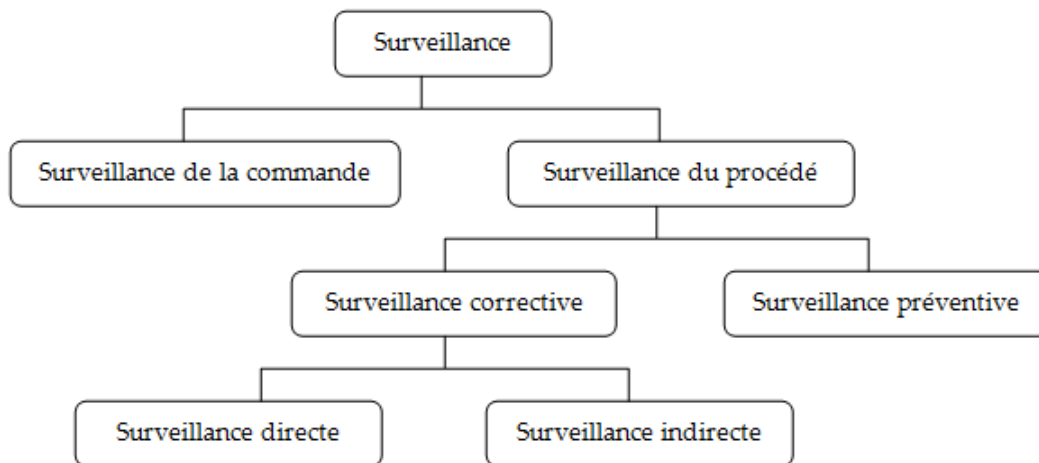


Figure I.2 – Les différents types de surveillance des SAP

- **Détection** : La fonction de détection permet de discerner tout écart du système par rapport à son état de fonctionnement normal. Elle indique la présence d'un défaut. Pour assurer cette fonction, il est indispensable de pouvoir distinguer entre les situations normale et anormale. Dans certains travaux, cette fonction est considérée comme un élément distinct de la fonction de diagnostic et plutôt une entité de la surveillance ([Combacau, 1991], [Toguyeni, 1992], [Boufaied, 2003]). D'autres travaux tels que [Isermann, 1984] et [Darkhovski et Staroswiecki, 2003] considèrent cette fonction comme une information primordiale et indissociable du diagnostic.

- **Diagnostic** : C'est une fonction d'information. Son objectif est de localiser les éléments défectueux, qui ont entraîné la dégradation du système et éventuellement de déterminer les causes [Morrison et Upton, 1994]. Pour les systèmes à événements discrets, la fonction de diagnostic établit donc un lien de cause à effet entre un symptôme observé et la défaillance qui est survenue, tout en considérant qu'un même symptôme peut apparaître pour différentes causes [Combacau *et al.*, 2000]. Pour les systèmes continus, la fonction de diagnostic indique l'amplitude des fautes. Cette fonction suit la fonction de détection et inclut les fonctions de localisation et d'identification.
 - **Localisation de défauts** : Elle permet de déterminer le sous-système affecté par le défaut. Cette fonction consiste à déterminer l'origine de l'anomalie et de localiser le ou les composants défectueux. Cette localisation est importante puisque la propagation d'une panne provoque souvent l'apparition de nouveaux défauts. Il est facile de détecter une défaillance résultant d'un défaut. Cependant il est difficile d'identifier la cause d'une défaillance, puisque une défaillance peut résulter d'un ou plusieurs défauts.
 - **Identification de défauts** : Elle consiste à identifier les causes qui ont mené à une situation anormale. Pour les systèmes à événements discrets, cette fonction consiste à identifier le(s) composant(s) dont la défaillance a entraîné celle du sous-système où elle a été localisée.
- **Supervision** : La supervision a pour objectif de contrôler l'exécution d'une opération et le fonctionnement d'une installation. En présence d'un défaut, la supervision doit prendre toutes les décisions correctives nécessaires pour assurer le retour vers un fonctionnement normal, en ayant la connaissance des causes ou des composants ayant générés une défaillance. Elle a donc un rôle décisionnel et opérationnel en vue de la reprise de la commande.
- **Pronostic** : La fonction de pronostic a pour rôle de prédire l'évolution et la propagation des défauts ou des défaillances en indiquant leurs conséquences inévitables sur le fonctionnement futur du système. Le pronostic permet l'identification de l'ensemble des tâches qui ne peuvent plus être exécutées. Il est donc intéressant d'étudier le pronostic puisqu'il permet une intervention préventive sur le procédé pour éviter les conséquences d'une panne.

2 Cadre d'étude

Cette partie permet de se positionner dans le contexte de l'étude tout en présentant le type des systèmes vers lesquels nous souhaitons orienter nos travaux.

2.1 Systèmes Automatisés de Production

Un système de production est un espace constitué de moyens de production ou ressources parmi lesquels des machines (M), des ressources de transport (T) et des produits (P). Les moyens de production permettent de réaliser des opérations (OP) permettant de transformer ou réarranger les produits. Un Système Automatisé de Production (SAP) permet de réaliser automatiquement ces opérations. Il se décompose en deux parties (voir figure I.3¹) :

- Partie Commande (PC) : Elle assure l'envoi des ordres vers la partie opérative et permet la communication avec l'opérateur. Les trois composantes principales de cette partie concernent la commande séquentielle, la surveillance et la supervision.
- Partie Opérative (PO) qui est composée des capteurs et actionneurs du procédé : Elle exécute les ordres envoyés par la PC à l'aide d'actionneurs. Les transformations réalisées par le procédé sont alors mesurées par des capteurs qui retournent à la PC les informations ainsi collectées (compte-rendus). Le procédé regroupe l'ensemble des organes physiques ayant pour but la transformation produits.

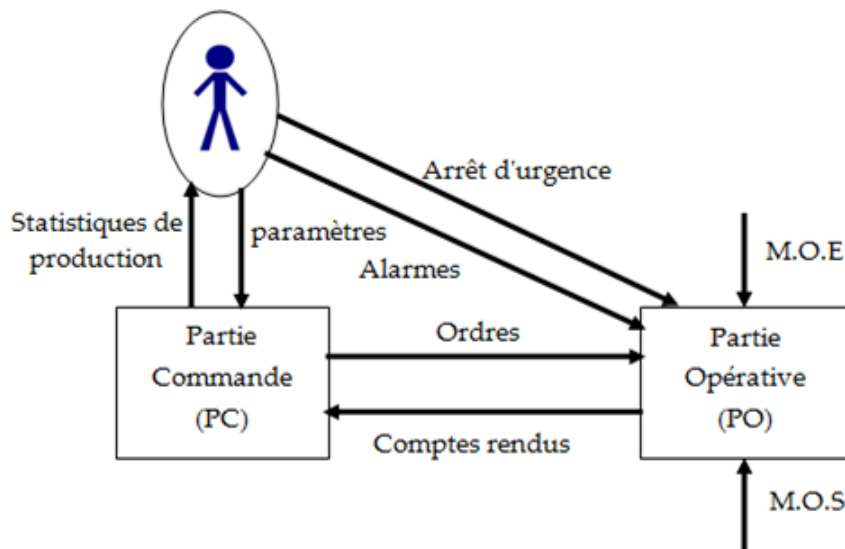


Figure I.3 – Structure d'un *SAP*¹

1. Schéma de la norme NF C03-190

Un SAP comprend donc un système d'information qui collecte, transmet, traite et mémorise des informations, des données et des décisions. Les produits fabriqués qui représentent les entités transformées doivent répondre aux exigences de qualité, de coût et de délai [Perrin *et al.*, 2004]. Pour produire efficacement, un système de production doit être doté de plusieurs caractéristiques telles que la flexibilité, la réactivité, la proactivité² et la robustesse. La partie suivante précise l'intérêt de la flexibilité dans la qualité du procédé.

Les SAP peuvent être caractérisés à travers la dynamique exprimant leur fonctionnement. Selon l'objectif considéré, il y a trois abstractions possibles pour modéliser cette dynamique : les systèmes continus, les systèmes à événements discrets et les systèmes dynamiques hybrides.

- Un système discret est tel que toutes ses variables d'état sont discrètes.
- Un système continu est tel que toutes ses variables d'état sont continues.
- Un système hybride contient des variables d'état continues et des variables d'état discrètes.

Une variable est dite discrète si elle peut prendre ses valeurs dans un ensemble de valeurs dénombrables (qui peut être infini). Une variable est dite continue si elle peut prendre ses valeurs dans un ensemble de valeurs continues (donc non dénombrables) et si ses variations ne présentent pas de discontinuités.

Les systèmes à événements discrets (SED) peuvent être rencontrés dans plusieurs domaines d'applications tels que la logistique [Abed *et al.*, 2013], les systèmes de transport, les réseaux de télécommunication, les systèmes informatiques, la production industrielle, les systèmes flexibles de production manufacturière (SFPM), etc.

2.2 Systèmes Flexibles de Production Manufacturière

Les Systèmes Flexibles de Production Manufacturière (SFPM) représentent une large classe des systèmes de production. Ces systèmes correspondent à des systèmes de production de petites à moyennes séries de produits sous un certain nombre de degrés de liberté. D'où le mot flexibilité qui signifie l'opportunité d'utiliser différentes procédures pour fabriquer des types de pièces différents. Ces systèmes sont capables de s'adapter aux changements de production afin de répondre aux exigences de diversité, de productivité et de qualité.

2. La proactivité d'un système de production est définie comme sa capacité à prévoir les changements d'état, à adapter ses règles de fonctionnement et à se réorganiser pour pouvoir ajuster son environnement

Les SFPM sont un compromis entre, d'une part les lignes de transfert automatisées à très haute productivité, conçues pour des productions de grandes séries (appelés aussi les systèmes de production de masse où la production est linéaire, sans flexibilité de routage), et d'autre part des installations manuelles à très faible automatisation permettant une production unitaire (appelés aussi les ateliers totalement flexibles où la production est unitaire) [Korbaa, 2003]. Leur but est de fabriquer et de transformer une matière première en produit fini ou semi-fini (voir figure I.4).

D'après [Kermad, 1996] un SFPM est « un atelier constitué d'un ensemble d'éléments capables de fabriquer un ensemble de pièces par programmation ». Ces systèmes, selon [Barbier et Jaulent, 1992], « forment une solution et un outil pour résoudre le problème de fabriquer le bon produit au bon moment en quantités nécessaires et suffisantes et au moindre coût ».

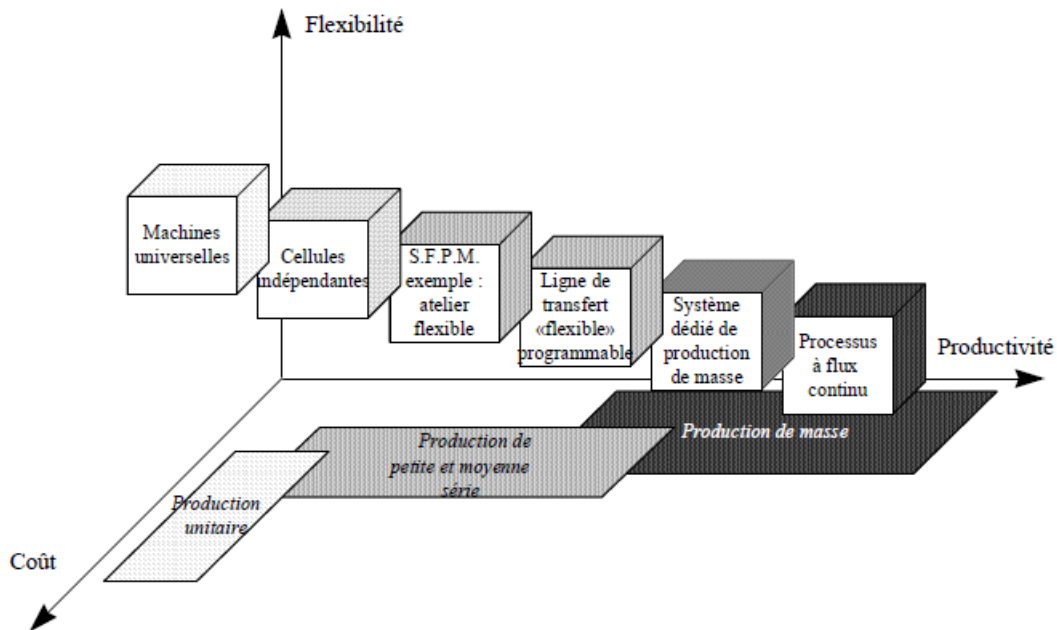


Figure I.4 – Productivité, Flexibilité et Coût [Kermad, 1996]

Les SFPM sont caractérisés par le fait que :

- La production de plusieurs types de pièces est effectuée selon des gammes opératoires (ensemble d'opérations successives, ou séquence fixée de tâches à accomplir).
- Certaines machines (ressources) peuvent être partagées par plusieurs gammes opératoires.

Pour un tel système, différents types de pièces se trouvent simultanément dans le système et sollicitent les mêmes ressources qui sont de deux types :

- les ressources de transformation qui sont capables d'effectuer différents types d'opérations (usinage, taraudage, tournage, fraisage, assemblage, etc.) sur différents types de pièces.
- les ressources de transport qui n'affectent que la position des pièces. Ces ressources peuvent être soit des palettes circulant sur des convoyeurs, soit des robots (par exemple pour le chargement/déchargement de machines), soit chariots filoguidés ...

Nous considérons des SFPM où il n'y a pas de préemption des tâches lors du fonctionnement normal du système (c'est à dire les opérations sont non préemptives). En pratique, cela signifie que les ressources utilisées par une opération ne sont libérées que lorsque l'opération n'en a plus besoin (à la fin de l'opération).

A travers ces définitions, nous pouvons percevoir deux avantages de ce type de système. Le premier concerne le fait de pouvoir produire simultanément un certain nombre de pièces de types différents (flexibilité de production). Quant au deuxième, c'est la capacité de modifier la production et les flux de manière réactive. Le coût assez élevé des investissements engendrés par les SFPM nécessite de bien spécifier les besoins en production et de maîtriser le fonctionnement de l'atelier de production. Il faut donc être en mesure de dimensionner l'atelier, d'optimiser le flux de production, de gérer les différents types de pièces cohabitant dans le système et de pouvoir guider l'évolution du système en faisant face aux aléas de la production [Korbaa, 1998]. Une telle démarche d'ingénierie nécessite une phase d'analyse préliminaire de performances et de détermination de la commande des SFPM [Korbaa *et al.*, 2003].

2.3 Commande cyclique d'un SFPM

Après avoir présenté les différentes caractéristiques des SFPM, nous justifions dans le paragraphe suivant le choix de la commande cyclique pour ce type de système de production. Ensuite nous présentons les concepts de base de l'ordonnancement cyclique.

2.3.1 Ordonnancement des systèmes de production

Les systèmes de production manufacturière mettent en œuvre des commandes de manière à répondre à des demandes de production. Ces demandes sont traduites hors ligne en ordonnancement de la production qui aboutit à définir les taux de production par ressource et les

dates d'entrées des produits. Les ordonnancements peuvent être classés en deux catégories selon l'aspect cyclique ou acyclique de la commande. Un ordonnancement acyclique est destinée essentiellement à une production de petite taille, voire unitaires, étant donné qu'elle considère toutes les tâches à effectuer.

Plusieurs travaux ont été menés dans le contexte des ordonnancements cycliques : [Hillion, 1989], [Hanan, 1995], [Gentina *et al.*, 2001], [Filho *et al.*, 2014] et [Candan et Yazgan, 2015]. Ces travaux ont résolu progressivement les flexibilités qui caractérisent ces systèmes de production afin d'optimiser la production et construire une commande cyclique permettant d'atteindre les performances optimales du système. Ces travaux ont opté pour une commande cyclique essentiellement pour deux raisons. La première justification concerne la taille de la demande initiale de production qui est adaptée aux SFPM. Ceci est justifié par le fait que les SFPM produisent par définition de petites et moyennes séries. Il ne s'agit pas alors d'une production unitaire ou de très petite taille. Dans ce sens, l'application d'une commande cyclique permet de limiter le nombre de pièces à produire sur un horizon donné afin d'optimiser la performance de production sur un cycle. Ensuite, il suffit de répéter suffisamment de cycles de production pour satisfaire la demande initiale. Des travaux ont montré que cette commande cyclique peut être adaptée aux petites séries moyennant un effort d'étude et optimisation des régimes transitoires associés [Gentina *et al.*, 2001]. La seconde justification de cette stratégie cyclique est la nécessité de maîtriser la complexité de ce type de problème. En effet, l'optimisation d'un cycle est insuffisante pour l'optimisation de la production totale qui reste sous-optimale en termes de performances. De plus, le problème considéré est souvent non déterministe polynomial au sens fort. Dans ce cas, la recherche de l'optimum dans le cas général est inconcevable à cause d'une très grande complexité combinatoire [Korbaa et Gentina, 2003].

2.3.2 Principaux concepts de l'ordonnancement cyclique

L'ordonnancement a pour objectif d'ordonner dans le temps l'exécution répétitive d'un ensemble d'opérations en utilisant un nombre limité de ressources [Benamar, 2013]. Ces opérations sont liées par des contraintes de précédence. Dans le contexte des systèmes de production manufacturière, après la demande initial de la part du client, vient l'étape de la planification [Korbaa, 1998]. Cette étape décompose la production en plusieurs régimes permanents et détermine le nombre de pièces à produire dans chaque régime. La transition entre ces différents régimes permanents sera gérée à travers des régimes transitoires [Belkahla *et al.*, 2007].

L'une des caractéristiques du régime permanent est la notion du temps de cycle qui représente la durée qui sépare deux occurrences d'une même opération entre deux cycles consécutifs. Le temps de cycle est qualifié d'optimal s'il est fixé par la charge³ de la ou des ressource(s) critique(s), également appelée ressource menante⁴. Cela permet d'atteindre une productivité maximale pour ces ressources (taux d'occupation = 100%) ce qui est envisageable afin de permettre l'optimisation de la totalité de l'ordonnancement. Une autre caractéristique du régime permanent est la notion d'en-cours qui correspond au nombre de produits en cours de fabrication dans chaque cycle. Dans le cas général, l'en-cours correspond au nombre maximum de gammes⁵ opératoires identiques en cours de production simultanément. La minimisation de l'en-cours est considérée comme l'un des critères de performance d'un ordonnancement cyclique [Benamar, 2013], et ce pour deux raisons. D'un part, la minimisation d'en-cours permet de minimiser le nombre de palettes utilisées dans le système, ce qui réduit en conséquence les coûts d'utilisation de ces en-cours. D'autre part, la minimisation de l'en-cours évite la saturation du système par un nombre de palettes qui dépasse sa capacité [Korbaa et Gentina, 2003].

Parmi les méthodes d'ordonnancement cyclique, nous pouvons citer :

- L'ordonnancement de la machine critique [Erschler *et al.*, 1982] avec minimisation des tailles des stocks d'entrée (buffers) des différentes machines.
- L'ordonnancement k-cyclique [Chretienne *et al.*, 1997] pour les systèmes d'information où la minimisation de l'en-cours est non abordée. Avec ce type d'ordonnancement, les machines répètent exactement le même fonctionnement tous les k cycles. Le recours aux ordonnancements k-cyclique peut améliorer le débit de production. Cependant il requiert d'ordonner un nombre plus important d'opérations sur une fenêtre temporelle plus grande [Kats et Levner, 2011].
- L'ordonnancement 1-cyclique [Hillion, 1989], [Valentin, 1994], [Camus *et al.*, 1996], [Korbaa *et al.*, 2002] et [Benamar, 2013] avec respect du temps de cycle optimal et minimisation de l'en-cours. Dans ce cas, les machines répètent à chaque cycle exactement le même fonctionnement (mêmes opérations aux mêmes dates). Dans certains cas, le chevauchement de cycle est une solution envisageable pour atteindre un temps de cycle optimal. Le chevauchement de cycles autorise qu'une opération commence dans un cycle et se termine au cycle suivant [Korbaa, 1998].

3. Charge d'une machine : somme des temps opératoires des opérations.

4. Ressource menante : la ressource la plus lente au vu de la charge effectuée.

5. Gamme : Ensemble des opérations à exécuter pour fabriquer un type de produit.

Afin de garantir les performances optimales du système de production, nous nous intéressons dans la suite à une commande 1-cyclique par rapport aux machines. Cela signifie qu'à chaque cycle, chaque machine effectue la même séquence d'opérations. Les objectifs principaux de cette méthode sont la maximalisation des flux de production sur un horizon donné, tout en minimisant l'en-cours.

2.4 Présentation d'exemples illustratifs

Les deux systèmes exposés dans cette partie représentent les cas d'études utilisés dans les chapitres suivants.

2.4.1 SFPM à plusieurs types de pièces

Nous représentons dans cette partie un exemple de SFPM à plusieurs types de pièces ordonnancées par [Valentin, 1994]. Le système est composé de trois machines (M_1, M_2 et M_3) utilisées pour réaliser deux types de pièces P_1 et P_2 .

La notation suivante $op_{\alpha i}^\beta$ présente la $i^{\text{ème}}$ opération de la gamme α . Une pièce de type P_1 est soumise à 3 opérations, la première est effectuée par la machine M_3 durant 2 unités de temps (u.t.), la deuxième est effectuée par la machine M_1 durant 3 u.t., la dernière est effectuée par la machine M_2 durant 2 u.t.. Une pièce de type P_2 est soumise à 2 opérations, la première est effectuée par la machine M_1 durant 1 unités de temps (u.t.) et la deuxième est effectuée par la machine M_3 durant 2 u.t.,

Dans la description du système, nous utilisons la notation suivante :

- $P_1 : op_{\alpha 1}^\beta(M_3, 2 \text{ u.t.}), \quad op_{\alpha 2}^\beta(M_1, 3 \text{ u.t.}), \quad op_{\alpha 3}^\beta(M_2, 2 \text{ u.t.})$
- $P_2 : op_{\alpha 1}^\beta(M_1, 1 \text{ u.t.}), \quad op_{\alpha 2}^\beta(M_3, 2 \text{ u.t.})$

La production cyclique consiste à produire dans chaque cycle 3 pièces de type P_1 notées $\{P_{11}, P_{12}, P_{13}\}$ et 2 pièces de type P_2 notées $\{P_{21}, P_{22}\}$, soit 5 pièces par cycle. Même si les trois pièces de P_1 sont identiques, nous avons besoin de les identifier et de les distinguer. En effet, étant donné que chaque pièce possède une date de sortie différente, il est nécessaire de savoir quelle pièce sort du système de production. D'où la nécessité de dupliquer trois fois la gamme du type P_1 et deux fois la gamme du type P_2 , afin de représenter les opérations sur chaque pièce.

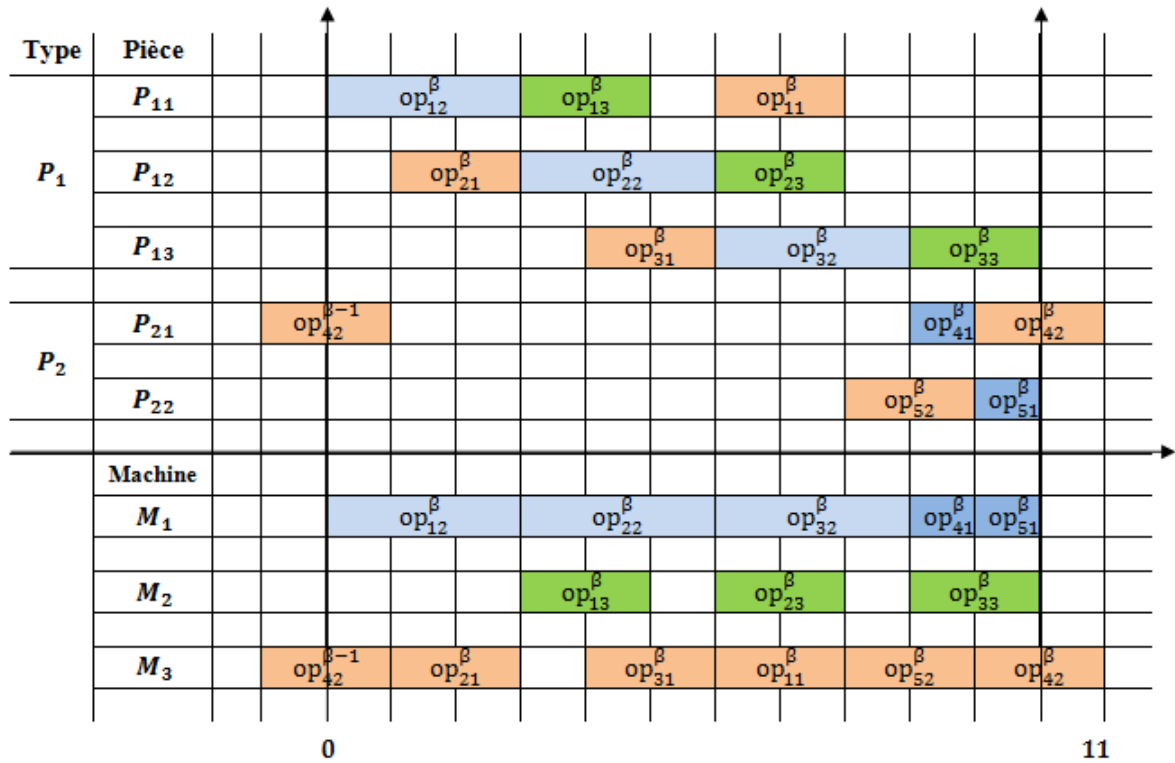


Figure I.5 – Diagramme de GANTT dual d'un SFPM

L'ordonnement cyclique déjà effectué par [Valentin, 1994], fixe le temps de cycle optimal à 11 u.t.. La machine menante, celle qui travaille à 100% du temps, est M_1 . L'en-cours utilisé est optimal et égal à 5 (un en-cours par pièce à fabriquer). Le résultat est représenté par un diagramme de Gantt dual (point de vue des machines et point de vue des pièces - Figure I.5). En analysant le taux d'occupation des en-cours dans ce système, nous constatons que les trois pièces de type P_1 occupent 64% du temps de cycle (soit $\frac{7}{11}$), alors que les deux pièces de type P_2 occupent 27% du temps de cycle (soit $\frac{3}{11}$). D'autre part, l'analyse du taux d'occupation des machines montre que la machine M_1 est utilisée à 100% du temps, M_2 à 54% du temps et M_3 à 91%. Ces dernières analyses affirment qu'il y a encore moins de marge dans ce système.

2.4.2 SFPM à plusieurs en-cours par produit

Nous représentons dans cette partie un exemple de SFPM à plusieurs en-cours par produit ordonnancé par [Korbaa, 1998]. Le système est composé de 4 machines $M_i/i \in \{1, \dots, 4\}$ utilisées pour réaliser 1 seul type de pièce :

$P : op_{11}(M_1, 3 \text{ u.t.}), \quad op_{12}(M_2, 3 \text{ u.t.}), \quad op_{13}(M_3, 3 \text{ u.t.}), \quad op_{14}(M_1, 2 \text{ u.t.}), \quad op_{15}(M_4, 4 \text{ u.t.});$

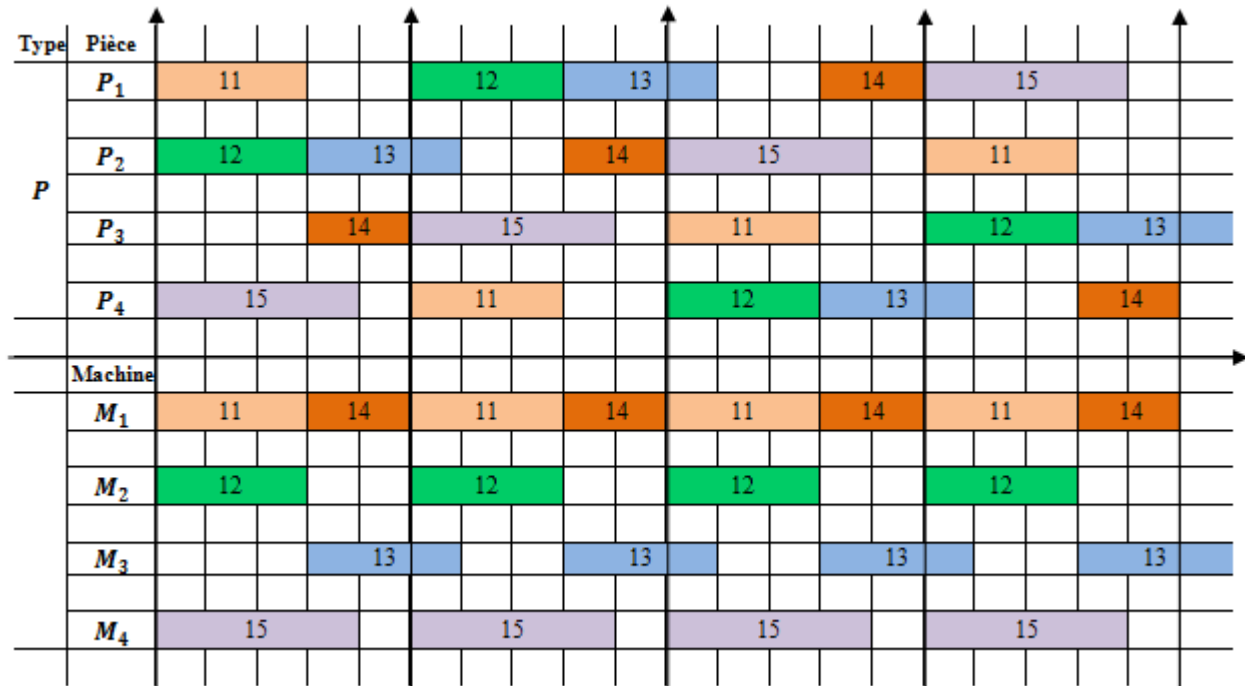


Figure I.6 – Ordonnancement d'un SFPM à plusieurs en-cours par produit

La production cyclique consiste à réaliser une seule pièce par cycle. L'ordonnancement cyclique fixe le temps de cycle optimal à 5 u.t.. La machine menante, celle qui travaille à 100% du temps, est M_1 . L'ordonnancement donne qu'il faut avoir 4 en-cours associés dans le système pour avoir une pièce finie en sortie du système par cycle (voir figure I.6). C'est à dire qu'il existe simultanément, 4 pièces brutes ou semi-finies dans le système. En effet, lors de chaque cycle on voit entrer une pièce et sortir une pièce du système, mais en fait il ne s'agit pas de la même pièce. Il est à noter que dans cet ordonnancement une pièce entière est créée sur 4 cycles.

3 Méthodes de diagnostic

L'étude de l'état de l'art a permis de montrer que les méthodes de diagnostic des défauts utilisées dans le milieu industriel sont très variées et diffèrent selon différents critères : la dynamique du système (discret [Simeu-Abazi *et al.*, 2010], continu ou hybride [Fliss et Tagina., 2013]), l'implémentation du diagnostic (en ligne [Chiacchio et Tommasi, 2009] ou hors ligne),

la complexité de procédé, la nature de l'information disponible (qualitative ou quantitative), la structure de prise de décision (centralisée ou décentralisée [Philippot, 2006], [Sayed-Mouchaweh *et al.*, 2008]).

Dans la suite, nous proposons une classification non exhaustive des principales méthodes de diagnostic rencontrées dans la littérature [Philippot, 2006]. Le principe général de ces méthodes se base sur une comparaison entre les données observées au cours du fonctionnement du système et les connaissances acquises sur son comportement normal et ses comportements en défaillance. Selon la connaissance dont on dispose sur le fonctionnement du système, ces méthodes peuvent être divisées en deux axes : avec ou sans modèles.

3.1 Méthodes de diagnostic sans modèles

Dans certains systèmes complexes la connaissance disponible sur le fonctionnement du système est insuffisante pour construire le modèle du système. D'où le recours à la connaissance a priori et au savoir-faire sur la structure et le comportement du système. Les méthodes sans modèles exploitent des données issues des capteurs ainsi que des informations reposant sur l'expertise humaine. Parmi ces méthodes, on trouve :

- Les méthodes à base de connaissance [Zwingelstein, 1995] qui utilisent une information experte pour lier les symptômes aux défauts au travers de règles. Parmi elles, on trouve les systèmes experts qui établissent des associations empiriques entre effets et causes, ou les approches inductives ou déductives comme l'AMDEC : Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité.
- Les méthodes d'apprentissage automatique qui construisent un modèle du système par apprentissage en utilisant un ensemble de données historiques sur ses différents modes de fonctionnements normaux et défectueux. Par exemple on trouve les méthodes de classification et de reconnaissance de formes [Dubuisson, 1990] et les réseaux de neurones [Zemouri, 2003].

3.2 Méthodes de diagnostic avec modèles

Les méthodes à base de modèles s'appuient sur des modèles de références pour avoir une base d'informations moins coûteuses par rapport à celles sans modèles. Ces méthodes se basent sur une comparaison du comportement du système avec le comportement du modèle qualitatif ou quantitatif établi [Combacau *et al.*, 2000].

3.2.1 Méthodes à base de modèles qualitatifs

Les modèles qualitatifs permettent de représenter le comportement du système avec un certain niveau d'abstraction à travers des modèles symboliques. Ces méthodes représentent de manière qualitative les systèmes continus, discrets et/ou hybrides et peuvent être classifiées selon le niveau d'abstraction considéré du système à diagnostiquer.

- Pour les systèmes continus, les approches ont été développées à base de graphes causaux et de graphes causaux temporels [Mosterman, 2001].
- Le diagnostic des systèmes à événements discrets est basé sur l'utilisation de modèles discrets tels que les automates d'états finis, les équations logiques, les chroniques [Dousson, 1994], l'interprétation de Signatures Temporelles Causales (STC) ([Toguyeni, 1992], [Saddem *et al.*, 2012]) et les réseaux de Petri avec observation partielle ou totale du fonctionnement du procédé [Genc et Lafortune, 2003], [Liu, 2014].
- Pour les systèmes hybrides, on trouve des méthodes reposant sur des modèles hybrides tels que les automates hybrides à temps discret [Bhowal *et al.*, 2007], les bond graphs [Tagina *et al.*, 1995], les RdP hybrides [Gomaa et Gentil, 1996].

3.2.2 Méthodes à base de modèles quantitatifs

Les méthodes à base de modèles quantitatifs s'appuient sur l'estimation et le suivi de paramètres représentatifs de l'état du système. Les principales méthodes s'appuient sur la représentation d'états, ou sur l'espace de parité [Gertler et Singer, 1990] qui sont des modèles mathématiques décrivant le comportement du système [Brunet *et al.*, 1990], [Zwingelstein, 1995]. Si l'écart entre les valeurs des paramètres obtenus à l'aide de ces modèles et celles mesurées sur le procédé dépasse un certain seuil, une défaillance est alors détectée. Cet écart représente ainsi un résidu qui sera fourni au diagnostic et comparé avec toutes les signatures des défauts connues, afin d'isoler et d'identifier la défaillance avant de prendre une décision ([Cocquempot *et al.*, 2004], [Gentil, 2007] et [Samantary et Ould-Bouamama, 2008]).

Ces méthodes à base de modèle quantitatifs souffrent de plusieurs problèmes, présentés dans les travaux de ([Staroswiecki, 1994], [Staroswiecki et Comtet-Varga, 2001]) par les problèmes relatifs à la surveillabilité. D'une part, le résultat de diagnostic dépend fortement de l'instrumentation utilisée et de la qualité des mesures que peuvent entacher de bruits et conduire à des résidus différents de zéro, provoquant ainsi des fausses détections. D'autre part, la difficulté majeure de ces méthodes réside dans la difficulté de positionnement de capteurs physiques pour mesurer toutes les grandeurs nécessaires pour le diagnostic [Zwingelstein, 1995].

3.3 Méthodes indirectes

Une autre classification des méthodes de diagnostic est possible selon les points d'observabilités définis dans le système à surveiller. Dans ce cadre, ces méthodes peuvent être classées en deux catégories : directes ou indirectes. Concernant les méthodes directes, nous nous contentons de la première classification pour éviter toute confusion. Nous exposons dans cette partie une synthèse de la littérature des méthodes indirectes basées sur des analyses quantitatives ou qualitatives de la production.

3.3.1 Approches quantitatives

Les approches quantitatives de surveillance indirectes ont été principalement abordées dans les travaux de [Ly, 1999] et [Toguyeni et Korbaa, 2005]. Le principe de ces approches se base sur le fait d'établir un lien de cause à effet entre les perturbations de flux de produits fabriqués et l'état dégradé des équipements. Ces approches partent de l'hypothèse que la cause d'une dégradation est liée à l'état défaillant d'une ressource ayant entraîné une baisse de ses performances, et donc un retard dans l'accomplissement de sa fonction de production par rapport au comportement nominal. Le principe général de ces approches est de détecter et de diagnostiquer une dérive de flux, et ensuite d'utiliser le pronostic pour estimer son impact temporel et quantitatif sur le système. La détection des dégradations progressives est assurée par l'exploitation d'un modèle qui définit des taux de dérives des ratios de production en différents points mesurables du système. La connaissance préalable des gammes opératoires et des ressources utilisées avec leurs ratios de production prévisionnels, permet donc de localiser et identifier les ressources défaillantes et de déterminer l'amplitude des dégradations qui affectent leur comportement sur un horizon donné.

3.3.2 Approches qualitatives

Les approches qualitatives de surveillance indirectes ont été principalement développées dans les travaux de [Nabli, 2000] et [Telmoudi *et al.*, 2008]. Ces approches reposent sur l'identification des principaux paramètres de mesure de la qualité de la production d'un système en différentes étapes de la production. Ensuite des indicateurs de qualités sont déduits à partir de ces paramètres. Ces indicateurs sont comparés à des seuils critiques afin de détecter les dégradations de la qualité du produit. Certains approches qualités partent de l'hypothèse que l'état défaillant d'une ressource peut dans certain cas causer une baisse de la qualité du produit voire une baisse de ses performances. Les travaux de [Telmoudi, 2011] combinent les deux aspects qualitatif et quantitatif. L'architecture proposée dans ces travaux est composée

de deux modèles : un modèle de commande qui s'appuie sur la modélisation de la robustesse⁶ du système de point de vue qualité et flux, et un modèle de surveillance qui permet de prévoir les évènements qui peuvent aboutir à un fonctionnement anormal du système.

La complexité de résolution et l'explosion combinatoire sont d'autres problématiques posées par les méthodes de diagnostic [Cassez, 2012]. Les approches modulaires [Su, 2004] peuvent être dans certains cas une piste pour résoudre le problème d'explosion combinatoire. Cependant dans d'autres cas le coût de calcul du diagnostic augmente si chaque partie de système est diagnostiquée séparément [Mhamdi *et al.*, 2013]. Dans le cadre du diagnostic en ligne où de nouvelles données sont continuellement fournies aux systèmes, le diagnostic incrémental peut être un axe de recherche intéressant.

3.4 Méthodes incrémentales

Le principe du diagnostic incrémental est de mettre à jour les résultats de résolution lorsque de nouvelles observations sont disponibles. Dans ce contexte, les travaux de [Grastien *et al.*, 2012] ont utilisé une approche de diagnostic qui reprend la résolution à chaque nouveau symptôme détecté, en calculant le diagnostic global pour toutes les observations. L'inconvénient de cette approche est l'augmentation continue du temps de résolution à chaque ajout d'observation [Grastien et Anbulagan, 2010].

Dans le même contexte, les travaux de [Pencole *et al.*, 2000] proposent d'établir un nouveau diagnostic lorsqu'une nouvelle observation est disponible, et ensuite concaténer les nouveaux résultats obtenus avec les précédents. Les chercheurs dans cette étude ont utilisé une décomposition temporelle et spatiale pour simplifier le problème du diagnostic. L'aspect spatial est réalisé par la mise en œuvre d'un diagnostic local à chaque composant du système. Selon ces auteurs, les résultats du diagnostic sont corrects pour chaque composant. Cependant l'algorithme proposé ne garantit pas l'exhaustivité [Grastien et Anbulagan, 2009] ou la consistance en général, c.à.d. le diagnostic ne prend pas en considération toutes les anciennes observations. En effet, il peut causer un problème de consistance entre les résultats du diagnostic initial et celui obtenu par les nouvelles observations. En effet, les observations du système sont partitionnées en composants (appelés dans [Grastien et Anbulagan, 2009] par les questions de diagnostic) qui sont diagnostiqués séparément.

6. La robustesse d'un système manufacturier peut être définie comme sa capacité à conserver ses propriétés spécifiées (temporelles, stabilité, rentabilité...) face à des perturbations prévues ou imprévues.

4 Positionnement des travaux dans le contexte scientifique

Après avoir présenté une étude des différentes méthodes de diagnostic, nous exposons une synthèse de ces méthodes tout en dégagant la problématique. Ce qui permet de positionner nos travaux dans le contexte scientifique.

4.1 Synthèse des méthodes de diagnostic

Nous avons présenté dans ce chapitre les différents critères de classification des méthodes de diagnostic issues de la littérature. Ce recueil permet de justifier nos choix dans l'approche de diagnostic adaptée au cadre de notre étude. Nos travaux sont orientés plus particulièrement vers les systèmes flexibles de production manufacturière qui représentent une catégorie importante des systèmes à événements discrets. Ces systèmes peuvent être représentés par un modèle qui décrit le comportement normal du procédé. Ce critère justifie le choix des méthodes à base de modèles comme une première orientation. En effet, ces méthodes permettent une représentation du procédé et de la commande à travers différents types de modèles. En se positionnant dans le contexte des systèmes à événements discrets, nous récapitulons dans le tableau I.1 une liste non exhaustive des références de la littérature des méthodes de diagnostic à base de modèles. Ces méthodes sont répertoriées selon la nature de l'information disponible (qualitative ou quantitative) et la disponibilité des points d'observabilités (directes ou indirectes).

	Méthodes directes	Méthodes indirectes
Méthodes à base de modèles quantitatifs	[Brunet <i>et al.</i> , 1990], [Zwingelstein, 1995], [Cocquempot <i>et al.</i> , 2004], [Gentil, 2007], [Samantary et Ould-Bouamama, 2008], ...	[Ly, 1999], [Toguyeni et Korbaa, 2005], ...
Méthodes à base de modèles qualitatifs	[Tagina <i>et al.</i> , 1995], [Bhowal <i>et al.</i> , 2007], ...	[Nabli, 2000], [Telmoudi <i>et al.</i> , 2008], ...

Tableau I.1 – Classement des méthodes de diagnostic à base de modèles des SED

4.2 Problématique et orientation

Les méthodes directes de diagnostic sont plus réactives, cependant elles présentent les inconvénients suivants :

- L'utilisation d'un grand nombre de capteurs de surveillance supplémentaires peut diminuer notablement la disponibilité du système.
- Les difficultés techniques de mise en place des capteurs dans certains milieux délicats peuvent augmenter les coûts d'utilisation. En effet, ces méthodes nécessitent souvent une instrumentation spécifique dont on ne dispose pas toujours quand la surveillance est mise en place a posteriori.
- La vulnérabilité des capteurs dans certains milieux industriels incite à filtrer l'information donnée par les capteurs.
- Afin d'augmenter la disponibilité du système, il est indispensable d'utiliser des capteurs à faible taux de défaillance.
- La défaillance d'un capteur (fausse alarme ou alarme manquante) peut amener à une mauvaise action de la partie commande (PC), ce qui peut causer la défaillance d'un autre composant.

Les méthodes indirectes de diagnostic qui se basent sur une analyse des flux de production permettent d'éviter les désavantages des capteurs de surveillance cités précédemment. Cependant elles sont moins réactives car elles nécessitent que la défaillance influe de manière importante sur le flux ou la qualité de la production. Par conséquent les méthodes indirectes ne sont pas toujours capables de localiser voire identifier les fautes du procédé.

L'objectif de cette thèse est de réussir à diagnostiquer les fautes d'un système industriel avec le minimum d'instrumentation. Dans ce cadre, on s'intéresse à la capacité à diagnostiquer l'état d'un système à partir de l'analyse quantitative de la production. Le principe de diagnostic indirect choisi consiste à comparer le flux de production réel par rapport au flux prévisionnel. Avec les méthodes classiques cette comparaison est effectuée à la fin de la période de production. Cela fait constater a posteriori les écarts, ce qui est contraignant pour le diagnostic en ligne où la rapidité du résultat est une exigence. L'objectif est de diagnostiquer le plus tôt possible en s'appuyant sur une méthode de diagnostic indirecte.

Afin de palier la limite de la non réactivité des méthodes indirectes, nous avons opté pour une commande déterministe et cyclique. En effet les ordonnancements cycliques donnent des points de références intermédiaires à la fin de chaque cycle de production, permettant donc d'avoir des points d'observation du système supplémentaires. Etant donnée que l'ordonnement 1-cyclique garantie un temps de cycle optimal [Korbaa *et al.*, 2002], alors ce dernier permet de simplifier la surveillance car toute variation dans les sorties des pièces sera directement détectée grâce à la saturation de la machine menante. De plus, les dates de sorties des pièces seront définies modulo le temps de cycle. Donc la minimisation de ce dernier permet d'observer rapidement la sortie des pièces et par conséquent d'augmenter la réactivité de la méthode de diagnostic. Nous détaillerons tout au long du chapitre suivant l'intérêt de l'ordonnement cyclique pour le diagnostic indirect.

5 Conclusion

Nous avons présenté dans ce chapitre la problématique générale du diagnostic des défauts dans les SFPM qui devient de plus en plus crucial, afin d'améliorer leurs performances et disponibilités. La première partie de ce chapitre a été dédiée à la présentation de la terminologie utilisée dans ce mémoire. Quant à la deuxième partie, elle a permis de décrire le contexte de l'étude. Ensuite, nous avons proposé une classification non exhaustive de différentes méthodes de diagnostic. A travers cette étude, nous avons souligné l'importance de minimiser l'utilisation des capteurs dans la procédure de surveillance tout en gardant une bonne qualité de la fonction de diagnostic en ligne. Afin d'assurer un bon fonctionnement d'un SFPM, il est indispensable de trouver un compromis entre ces deux exigences. Ce chapitre nous a permis d'orienter notre choix vers la conception d'une approche de diagnostic indirect pour les systèmes flexibles de production manufacturière sous ordonnancement 1-cyclique.

Pour répondre à la problématique définie dans ce premier chapitre, nous proposerons dans les deux chapitres suivants deux méthodes de construction d'un diagnostic indirect et en ligne pour les systèmes flexibles de production manufacturière soumis à une commande cyclique. Cette démarche commencera, dans le deuxième chapitre, par la proposition d'une approche indirecte par analyse quantitative des flux de production basée sur la génération d'arbres de résolution. Par la suite (dans le 3ème chapitre), nous mettrons l'accent sur l'intérêt d'une approche incrémentale pour résoudre le problème de convergence de la résolution. La deuxième méthode de diagnostic, définie dans le troisième chapitre, reposera sur la conception d'un modèle

mathématique pour la programmation par contraintes permettant de répondre aux exigences du diagnostic indirect, en ligne et incrémental.

Chapitre II

Méthode de diagnostic indirect par arbre de résolution basé sur la technique du chaînage double

Sommaire

1	Hypothèses de diagnostic	30
2	Modélisation de l'aspect cyclique de l'ordonnancement	31
2.1	Choix conceptuel de la modélisation cyclique	32
2.2	Positionnement de l'approche dans le contexte de l'étude	33
3	Démarche de résolution	34
3.1	Construction de PERT cyclique	35
3.2	Génération des hypothèses par chaînage arrière	36
3.3	Validation des hypothèses par chaînage avant	41
3.4	Modélisation de l'induction du retard	42
4	Diagnostic par chaînage double	44
4.1	Tests et résultats du diagnostic	44
4.2	Analyse du principe de résolution	46
5	Efficacité du diagnostic	47
6	Conclusion	48

L'objectif de notre travail proposé dans ce chapitre, est de développer une approche de diagnostic en ligne basée sur l'analyse des flux de production d'un Système Flexible de Production Manufacturière (SFPM) adaptée à la conduite cyclique. Le choix de l'ordonnement cyclique lors de la surveillance des défaillances permet de définir des cycles au niveau du système de production et donc d'avoir des références régulières en termes d'observation et de contrôle de la production en sortie. Cela permet d'avoir une réactivité accrue face aux perturbations. En analysant les flux de production en sortie d'un SFPM soumis à une commande cyclique, nous visons à déterminer l'opération défaillante du système qui a engendré la diminution des performances du procédé et le retard d'une pièce observé à la sortie du système. Cette approche doit tenir compte des marges qui existent entre les opérations d'une gamme opératoire ou se succédant sur une machine, et ainsi du retard propagé par induction de gamme ou de machine.

Ce chapitre se compose de cinq parties. Dans la première partie, nous introduisons les hypothèses de base de notre travail. Ensuite nous justifions le choix conceptuel de la modélisation cyclique adoptée dans ce travail. Dans la troisième partie de ce chapitre, nous présentons la démarche de résolution que nous proposons pour répondre aux besoins du diagnostic indirect et en ligne. Ensuite nous appliquons le diagnostic proposé à un cas d'étude tout en exécutant un scénario de défaillance. Une analyse des résultats et une critique de l'efficacité de la méthode de résolution sont détaillées dans la dernière partie de ce chapitre.

1 Hypothèses de diagnostic

Avant de développer notre approche de diagnostic, nous fixons au préalable les hypothèses de base de notre travail.

- **A1 : Toute diminution de la productivité est due à une augmentation des durées opératoires.** Cette augmentation peut résulter de défaillances par dégradation de ressources du système.
- **A2 : Les défaillances sont permanentes et cohérentes.** C'est-à-dire qu'une défaillance produit les mêmes effets dans le temps jusqu'à la réparation de la ressource défaillante. La cohérence signifie qu'une défaillance ne peut que s'amplifier au fil du temps.
- **A3 : Chaque retard de pièce en sortie du système est la conséquence d'une défaillance dans le système de production.** Par conséquent, pour le diagnostic, nous nous positionnons à la sortie du système pour observer le flux de production. C'est

une contrainte opératoire qui permet de s'affranchir de la nécessité de rendre observable en interne les défaillances par ajout de capteurs de surveillance. Donc cela signifie que nous n'observons pas le comportement interne du système. Seules les dates de sortie (ou de non sortie) des pièces sont « observables ».

- **A4 : Le diagnostic est effectué dans le contexte de défaillance unique.** Cela signifie que nous supposons que le système de diagnostic est suffisamment rapide et efficace pour détecter, isoler et identifier une défaillance avant que n'en survienne une autre. C'est une hypothèse qui permet de simplifier le diagnostic notamment dans le cadre d'une approche incrémentale. Cette hypothèse nous semble appropriée aux ordonnancements cycliques qui ne sont pertinents que dans l'hypothèse de pannes rares [Korbaa, 1998].
- **A5 : Une seule fonction d'une ressource peut être défaillante.** Cela signifie que pour une machine multi-opérations, un seul type d'opération à la fois peut être la source première de défaillance durant un cycle de production.
- **A6 : Le diagnostic est lancé à la première détection du retard d'une pièce.** Nous considérons que toutes les observations antérieures ont été réalisées correctement sans retard, sauf la dernière, celle du cycle courant.
- **A7 : Sur l'horizon de production considéré, une défaillance entraîne un retard propre constant de l'opération de la ressource affectée par la défaillance.** Cela signifie que si une opération d'une machine est défaillante alors les mêmes opérations de la même machine qui la succèdent ont le même retard propre que la première. L'hypothèse A7 est une conséquence des hypothèses A2 et A4.

Dans la suite, une approche sera proposée pour la résolution du problème de diagnostic indirect d'un SFPM sous ordonnancement cyclique.

2 Modélisation de l'aspect cyclique de l'ordonnancement

Dans un SFPM sous ordonnancement cyclique, l'aspect cyclique peut être représenté du point de vue des machines ou du point de vue des pièces, comme le montre le diagramme de Gantt dual (voir exemple figure I.5). Une commande 1-cyclique par rapport aux machines est définie par le fait que chaque machine effectue la même séquence d'opérations à chaque cycle : c'est-à-dire elle reprend exactement le même état et le même fonctionnement à chaque cycle. Le procédé de fabrication des pièces suit également le même aspect cyclique. Nous devons alors faire un choix de l'aspect cyclique à suivre durant le diagnostic.

2.1 Choix conceptuel de la modélisation cyclique

Dans cette partie, nous justifions le choix de la modélisation cyclique adopté dans la suite de ce mémoire.

En considérant un aspect cyclique par rapport aux pièces, le processus de modélisation d'un SFPM doit suivre deux étapes. Dans une première étape, les opérations des pièces d'un cycle et les contraintes d'induction de gamme sont modélisées. Dans une deuxième étape, il faut regrouper ces opérations par ordre de leur traitement par les machines afin de modéliser les contraintes d'induction des machines. Cependant en construisant le chaînage entre les opérations par induction de machines, il peut manquer des opérations qui appartiennent à d'autres pièces et qui ne sont pas représentées dans le cycle courant. Par conséquent les contraintes d'induction des machines seront violées. Considérons comme exemple l'ordonnancement de la figure I.5. Avec une commande cyclique par rapport aux pièces et en modélisant les produits dans le cycle courant, nous remarquons que l'opération op_{11}^{β} ne sera pas modélisée puisqu'elle appartient à la pièce P_{11} qui sera produite dans le cycle suivant. En faisant la projection de cette opération sur la série des opérations de la machine M_1 , on remarque que cette opération se trouve entre deux opérations modélisées dans le cycle étudié. Le fait de ne pas considérer cette opération dans la représentation du cycle mène à enfreindre les contraintes d'induction des machines.

En outre, dans le cas d'un ordonnancement cyclique avec plusieurs en-cours, les périodicités nécessaires pour modéliser une pièce à plusieurs en-cours sont relativement importantes. C'est à dire si l'ordonnancement établit qu'il faut avoir n en-cours associés dans le système pour avoir une pièce finie sortie par cycle, alors modéliser une pièce revient à voir n cycle avant. En effet, dans chaque cycle on voit entrer une pièce et sortir une pièce, mais réellement il ne s'agit pas de la même pièce. Dans le cycle i on voit sortir la pièce lancée dans le cycle $i-n+1$, où n représente le nombre d'en-cours pour la gamme considérée. Ainsi la modélisation d'une gamme entière à n en-cours peut être coûteuse du point de vue mémoire puisqu'il faut modéliser les $n-1$ cycles antérieurs au cycle courant. Considérons comme exemple l'ordonnancement de la figure I.6 où la production est 4-cyclique par rapport aux produits, cela signifie que pour fabriquer complètement une pièce il faut mettre en œuvre 4 cycles par rapport aux produits.

De plus, par défaut chaque ordonnancement est 1-cyclique par rapport aux machines. Sauf que comme nous utilisons une approche indirecte, ce sont les produits que nous observons et

non pas les machines. Si chaque produit possède un seul en-cours donc la périodicité des en-cours est égale à celle des machines. Mais dans le cas général un produit peut posséder plusieurs en-cours. Dans ce cas, la production d'une pièce nécessite plusieurs cycles. Par conséquent, les opérations d'une pièce donnée peuvent appartenir à des cycles différents ce qui rend difficile le diagnostic. La modélisation doit donc prendre en considération le décalage de cycle. C'est la difficulté majeure de cette modélisation.

Nous proposons alors dans ce cadre une solution où l'aspect cyclique est représenté du point de vue des machines et les opérations sont organisées selon leur ordre de séquençement dans les gammes de production. De ce fait, nous gardons les deux types d'inductions : induction de machine et induction de gamme. En examinant chaque cycle nous remarquons que les enchaînements entre les différentes opérations sont cycliques et ce dans le cas d'un ou plusieurs en-cours.

Afin d'assurer une modélisation cyclique d'un SFPM tout en respectant toutes les contraintes de l'ordonnancement, nous proposons de modéliser les opérations d'un cycle machine, ensuite de construire les inductions des gammes et des machines et finalement de modéliser les enchaînements entre les cycles. Le choix de la modélisation cyclique par rapport aux machines permet de garder le temps de cycle optimal. En effet, par définition de l'ordonnancement cyclique, une commande 1-cyclique par rapport aux machines permet de garantir les performances optimales [Korbaa *et al.*, 2002] du système de production. Par conséquent, cette modélisation permet de minimiser la période d'observation, ce qui permet d'augmenter la réactivité du diagnostic.

2.2 Positionnement de l'approche dans le contexte de l'étude

Dans les travaux de [Ly, 1999], [Ly *et al.*, 2000], une approche de surveillance indirecte dans le contexte d'un ordonnancement classique avec pilotage réactif d'un SFPM a été proposée. Cette approche est basée sur l'analyse des flux de production en se plaçant dans le cadre d'un ordonnancement par rapport aux produits. Bien qu'innovante sur plusieurs aspects, nous constatons la limite de la réactivité de cette approche. En effet, les périodicités nécessaires pour obtenir un comportement cyclique sont relativement importantes. En conséquences, d'un point de vue temporel, nous avons moins de points d'observations.

Pour un diagnostic indirect et dans le contexte d'un ordonnancement cyclique par rapport aux machines, les auteurs dans [Toguyeni *et Korbaa*, 2005] ont proposé une approche de diagnostic qui se base sur un chaînage mixte. Cette approche alterne entre la phase de génération

des hypothèses par chaînage arrière et la phase de validation des hypothèses par chaînage avant. L'idée proposée dans ces travaux consiste à valider chaque hypothèse générée avant de passer à une autre. Cela implique que le chaînage avant est appliqué à chaque nœud de l'arbre des hypothèses avant de terminer le chaînage arrière des nœuds, d'où le nom du chaînage mixte. Cette approche a permis d'améliorer l'observabilité du système par rapport aux travaux de [Ly *et al.*, 2000]. Cette amélioration résulte d'une augmentation des dates d'observation du système, rendant la méthode plus réactive. Néanmoins, le chaînage mixte proposé demeure une proposition théorique dans les travaux de [Toguyeni et Korbaa, 2005].

En se basant sur les travaux de [Toguyeni et Korbaa, 2005], nous proposons dans ce travail une autre approche de diagnostic indirect par analyse des flux de production. Afin d'envisager ce type de diagnostic, nous nous situons dans le cadre d'une conduite cyclique par rapport aux machines essentiellement pour améliorer la réactivité de la méthode de diagnostic. La méthode de diagnostic que nous proposons sera détaillée dans la section suivante.

3 Démarche de résolution

Le raisonnement de diagnostic que nous proposons dans cette première partie se base principalement sur l'analyse de l'écart du comportement défaillant par rapport au comportement normal du système. Le comportement normal est défini a priori par l'ordonnancement cyclique prévisionnel du système. Tandis que le comportement défaillant est perçu par le biais de la perturbation du flux de production. Il se manifeste donc par l'ensemble des pièces en retard par rapport à leurs dates de sorties théoriques spécifiées par l'ordonnancement prévisionnel. Le premier retard ainsi détecté est un événement que nous qualifions de « **symptôme initiateur** ». A partir du symptôme initiateur, le principe du diagnostic proposé consiste à générer toutes les hypothèses possibles a priori cohérentes avec ce symptôme. Ces hypothèses devront par la suite être confirmées (respectivement réfutées) par l'observation (respectivement la non observation) des retards d'autres pièces en sortie, au cours du cycle courant (cycle de détection du premier symptôme) ou de cycles suivants (dans le contexte d'un ordonnancement 1-cyclique par rapport aux machines). Le processus de résolution est décrit par la figure II.1¹.

Nous proposons dans la suite une approche de diagnostic indirect basée sur les étapes suivantes :

1. Modèle SADT : Structured Analysis and Design Technics

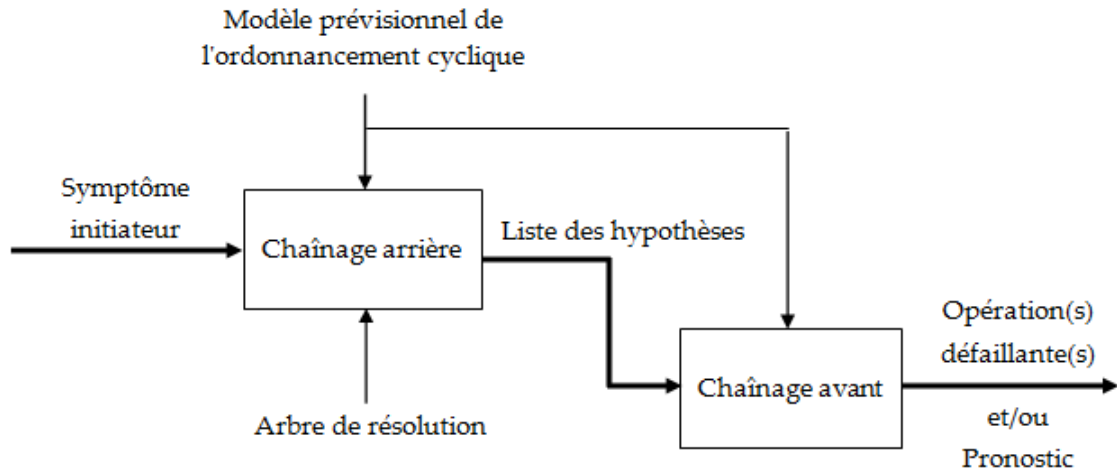


Figure II.1 – Processus de diagnostic indirect par analyse des flux

- **Étape 1** : Construire le graphe potentiel étape (PERT) d'un cycle de l'ordonnancement décrit dans le diagramme de Gantt.
- **Étape 2** : Générer l'arbre des hypothèses obtenues par un chaînage arrière sur le PERT déclenché par la détection d'un symptôme initiateur. (Algorithme 2)
- **Étape 3** : Parcourir l'arbre des hypothèses généré dans l'étape précédente et valider ces hypothèses par chaînage avant sur le PERT (Algorithme 3) :
 - ▷ Réfuter les hypothèses fausses : toutes les opérations observables², non défaillantes et situées à une date antérieure à celle du symptôme initiateur.
 - ▷ Garder les hypothèses possibles : toutes opérations observables et situées à une date postérieure à celle du symptôme initiateur.
 - ▷ Les hypothèses restantes peuvent être des sources de défaillance : toutes opérations non observables.
- **Étape 4** : Vérifier l'impact temporel des hypothèses restantes dans les cycles suivants et continuer le chaînage.

3.1 Construction de PERT cyclique

Afin de construire l'arbre des hypothèses, nous avons besoin d'un modèle qui décrit l'enchaînement des opérations dans un SFPM, tout en respectant les données de l'ordonnancement

2. Une opération est dite observable si elle est issue d'un événement observable. C'est une opération effectuée par une ressource observée par l'environnement du système. Voir définition de l'observabilité dans le chapitre 1

telles que les dates d'entrées et les durées des opérations. Ce modèle doit aussi conserver les contraintes de précédences entre les opérations sur les gammes et les machines. Nous proposons ainsi de construire le PERT à partir du diagramme de Gantt. Le choix du PERT est motivé par sa simplicité, tout en possédant les informations du Gantt (à savoir les dates de début de chaque opération ainsi que les marges). Ce graphe est composé de séquences d'opérations de l'ordonnancement cyclique ainsi que des contraintes de précédences entre les pièces et les machines dans un seul cycle. La figure II.2 résume à la fois les dépendances entre les différentes opérations ainsi que leurs dates de début au plus tôt et leurs dates de début au plus tard (ce qui fait représenter les marges entre des opérations successives). La construction du graphe PERT se base sur le diagramme de Gantt décrit par la figure I.5. Il s'agit de considérer les séquences d'opérations de chaque machine et de les représenter avec PERT tout en notant les dates de début au plus tôt de chaque opération. Les dates de début au plus tard sont calculées à partir des dates de début au plus tôt des deux opérations en aval ainsi que la durée de l'opération étudiée. Par exemple, l'observation des marges réelles des différentes opérations nous indique que la machine M_1 ne possède aucune marge. En effet, pour toutes les opérations réalisées par la machine M_1 , nous constatons que les dates de début au plus tôt sont égales à celles au plus tard. Ceci est justifié par le fait que M_1 est une machine menante.

En analysant ce graphe (voir figure II.2) nous remarquons que chaque opération possède deux opérations directement en aval (un successeur de gamme et un successeur de machine) et deux opérations en amont (un prédécesseur de gamme et un prédécesseur de machine). En effet, chaque opération est suivie par une autre opération dans la même gamme de fabrication. Cette même opération est également suivie par une autre opération sur la même machine. Même la dernière opération est suivie par la première étant donné l'aspect cyclique du problème. Il en est de même pour les opérations en amont.

Chaque nœud du PERT contient les informations nécessaires à la construction du modèle de connaissance de l'outil de diagnostic. La structure d'un nœud de PERT est décrite par le tableau II.1.

3.2 Génération des hypothèses par chaînage arrière

Un modèle de connaissance décrivant le comportement anormal du système est nécessaire pour énumérer toutes les causes possibles de défaillances. A partir de l'ordonnancement cyclique qui modélise le comportement normal du système, nous proposons la construction de l'arbre

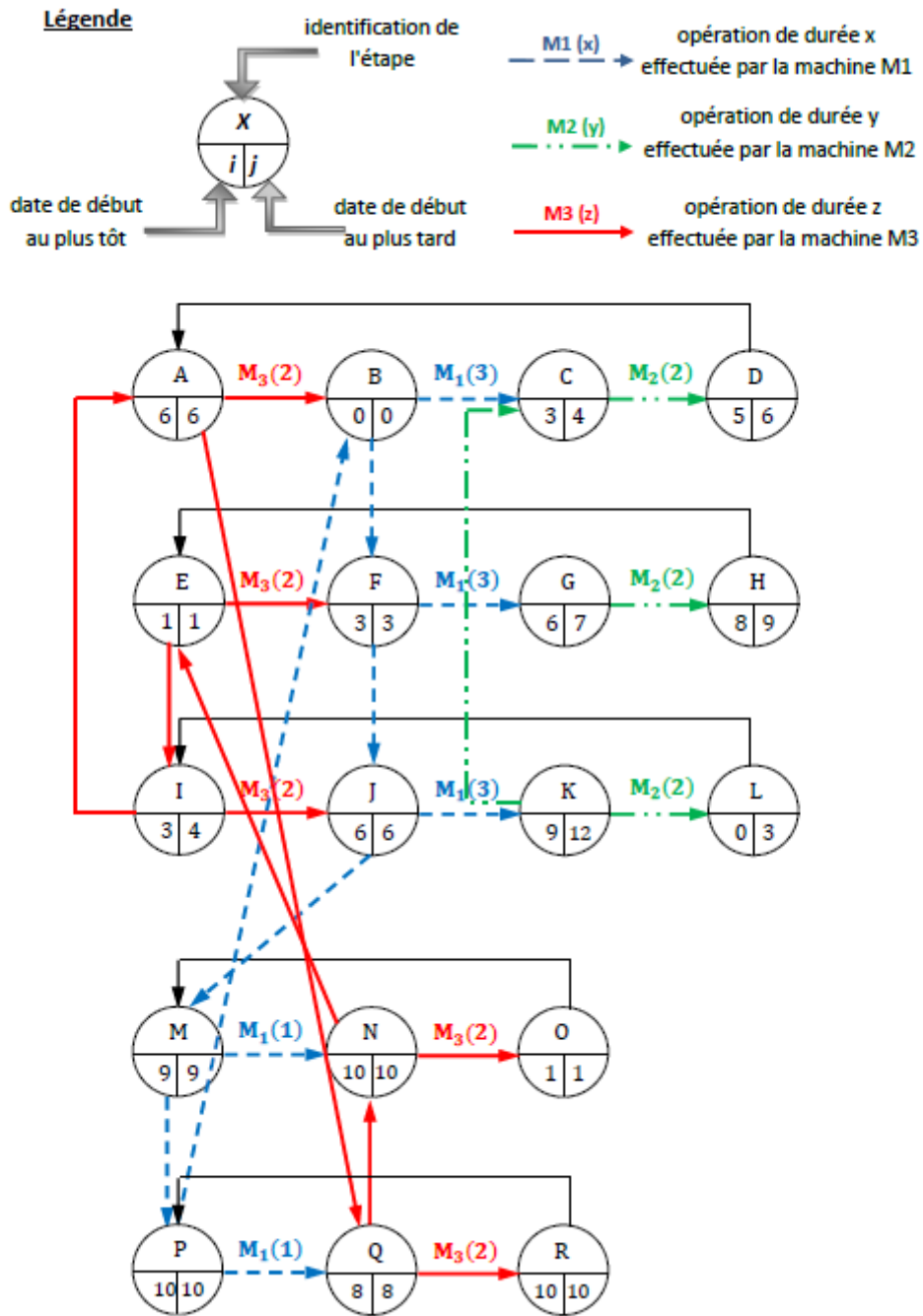


Figure II.2 – Graphe potentiel étape PERT cyclique

des hypothèses. La génération de cet arbre est lancée dès la première détection d'un symptôme initiateur par l'observation d'un retard à la sortie d'une pièce. A partir de ce symptôme initiateur on détermine toutes les hypothèses possibles par chaînage arrière sur le PERT en se basant sur les contraintes de précédences entre les circuits définies par l'ordonnancement.

Paramètre	Explication
P_i	Gamme
M_i	Machine
op_i	Opération
Gm_{sv}	Opération suivante par induction de gamme
Rs_{sv}	Opération suivante par induction de machine
Gm_{pr}	Opération précédente par induction de gamme
Rs_{pr}	Opération précédente par induction de machine
$Ep(op_i)$	Date de début de l'opération
$D(op_i)$	Durée de l'opération

Tableau II.1 – Les paramètres d'un nœud du PERT

Nous distinguons trois types d'hypothèses relatives à la violation de l'ordonnancement prévisionnel :

- **Défaillance propre** : Ce type d'hypothèse indique que l'opération considérée peut être la cause première de la défaillance détectée en sortie du système.
- **Défaillance par induction** : Ce type d'hypothèse indique que l'opération considérée peut être décalée dans le temps par rapport à l'ordonnancement prévisionnel. Ce décalage est dû à l'appartenance de l'opération à un circuit contenant une opération défaillante située en amont de l'opération considérée. Dans ce contexte, le retard d'une opération est la conséquence d'un retard propre ou d'un retard induit d'une opération en amont et appartenant à un circuit critique. Nous distinguons deux sous types de défaillance par induction.
 - ▷ **Induction de gamme** : perturbation de la séquence d'opérations planifiée par rapport à un type de pièce.
 - ▷ **Induction de machine** : perturbation de la séquence d'opérations planifiée par rapport à une machine.

Le principe de construction de l'arbre des hypothèses est basé sur le fait de parcourir le PERT à partir du nœud correspondant au symptôme initiateur et à mémoriser chaque nœud qui lui est lié par induction de gamme ou induction de machine. Les deux conditions d'arrêt de parcours de ce graphe sont les suivantes :

- Détecter une fausse hypothèse. Il s'agit de détecter une opération qui ne peut pas être une source de défaillance car elle a été observée avant le symptôme initiateur sans retard.
- Atteindre la fin du cycle d'ordonnement.

Pour faciliter la compréhension de la construction de l'arbre des hypothèses, nous proposons une représentation graphique avec la sémantique des nœuds et des arcs (voir figure II.3). Chaque nœud de l'arbre modélise une hypothèse de défaillance d'une opération op_k réalisée par une machine M_j sur une pièce P_i . La durée de retard qui caractérise la défaillance de cette opération est donnée entre parenthèses (ε). Nous modélisons ainsi dans la figure II.4 la propagation des contraintes temporelles relatives au phénomène d'induction du retard pour une succession d'opération. Cette induction est justifiée par l'hypothèse A2, qui suppose que les défaillances sont cohérentes et récurrentes.

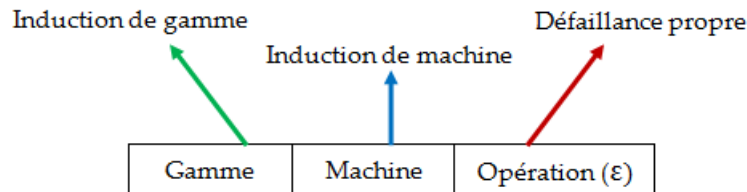


Figure II.3 – La sémantique des nœuds et des arcs dans un arbre de résolution

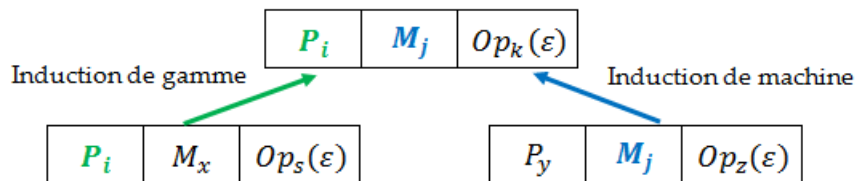


Figure II.4 – Propagation des contraintes temporelles entre des nœuds adjacents

Soit les paramètres suivants utilisés dans l'algorithme de chaînage arrière :

- NodeTree : un pointeur sur un nœud de l'arbre des hypothèses initialisé sur la racine de l'arbre qui représente le symptôme initiateur.
- NodePert : un pointeur sur un nœud du PERT. Ce pointeur est initialisé sur le nœud qui correspond au symptôme initiateur.
- Successeur_Gamme (nœud) : une fonction qui retourne le successeur de nœud par induction de gamme.

- Prédécesseur_Gamme (nœud) : une fonction qui retourne le prédécesseur de nœud par induction de gamme.
- Successeur_Machine (nœud) : une fonction qui retourne le successeur de nœud par induction de machine.
- Prédécesseur_Machine (nœud) : une fonction qui retourne le prédécesseur de nœud par induction de machine.
- Défaillance_Possible (nœud) : une fonction qui vérifie si le nœud peut être une source de défaillance. Cette fonction est décrite comme suit :

```
Fonction Défaillance_Possible( nodePert : NodePert) : Boolean  
  result : Boolean  
  Si (non-vide (nodePert) ET non-observable (nodePert) ET date-sortie (nodePert)  
  < date-sortie (Symptôme-Initiateur)) Alors  
    | result ← True;  
  Sinon  
    | result ← False;  
  Fin Si  
  Retourner result;  
Fin
```

Algorithme 1 – Fonction défaillance possible

Chaînage_Arrière (NodeTree, NodePert) : est la fonction qui permet de construire l'arbre des hypothèses à partir du symptôme initiateur représenté par la racine de l'arbre. Cette fonction est récursive et définie par l'Algorithme 2.

```
Procédure Chaînage_Arrière( nodeTree : NodeTree  nodePert : NodePert)
|
| Si (Défaillance_Possible(nodePert)) Alors
| |
| | Si (Défaillance_Possible (Successeur_Gamme( nodePert))) Alors
| | |
| | | 1. Successeur_Gamme(nodePert) ← Créer_NodeTree (nodePert)
| | | 2. Chaînage_Arrière (Successeur_Gamme (nodeTree), Prédéces-
| | | seur_Gamme (nodePert))
| | | Fin Si
| |
| | Si (Défaillance_Possible (Successeur_Machine(nodePert))) Alors
| | |
| | | 1. Successeur_Machine(nodePert) ← Créer_NodeTree(nodePert)
| | | 2. Chaînage_Arrière (Successeur_Machine (nodeTree), Prédéces-
| | | seur_Machine (nodePert))
| | | Fin Si
| |
| Fin Si
|
Fin
```

Algorithme 2 – Chaînage arrière

3.3 Validation des hypothèses par chaînage avant

Afin d'identifier la cause première du retard détecté à la sortie de système de production, nous avons besoin de valider les hypothèses générées dans l'étape précédente. Nous proposons ainsi de parcourir l'arbre des hypothèses à partir des feuilles et de prédire leurs conséquences afin de les valider par chaînage avant sur PERT. La validation de ces hypothèses se base sur les règles suivantes :

- Réfuter les hypothèses qui mènent à détecter, lors du chaînage, au moins une opération observable, non défaillante et située à une date antérieure à celle du symptôme initiateur. (Hypothèse non valide)
- Confirmer les hypothèses qui mènent à l'observation de retards d'autres pièces déjà détectées défaillantes. (Hypothèse possible)
- Les hypothèses situées à une date postérieure à celle du symptôme initiateur restent à vérifier dans les cycles suivants. (Hypothèse possible)

Le principe du chaînage avant est décrit par l'Algorithme 3.

```

Procédure Chaînage_Avant( nodeTree : NodeTree)
  Si (non-vide (nodeTree)) Alors
    Si (observable (nodeTree) ET date-sortie (nodeTree) < date-sortie (Symptôme-
    Initiateur)) Alors
      1.Hypothèse non valide
      2. Chaînage_Avant ← Faux
    Sinon
      Si (observable (nodeTree) ET date-sortie (nodeTree) > date-sortie
      (Symptôme-Initiateur)) Alors
        1.Hypothèse-à-valider (nodeTree, Symptôme_Initiateur)
        2. Chaînage_Avant ← Vrai
      Sinon
        Si (Chaînage_Avant(Successeur_Gamme(nodeTree)) est Faux
        OU Chaînage_Avant(Successeur_Machine(nodeTree)) est Faux))
        Alors
          1.Hypothèse non valide
          2. Chaînage_Avant ← Faux
        Sinon
          Chaînage_Avant ← Vrai
        Fin Si
      Fin Si
    Fin Si
  Fin Si
Fin

```

Algorithme 3 – Chaînage avant

Afin de parcourir l'arbre des hypothèses, nous choisissons un parcours en profondeur post-fixé. Pour chaque nœud, il s'agit d'appliquer l'algorithme de chaînage avant proposé.

3.4 Modélisation de l'induction du retard

Nous modélisons dans cette partie les contraintes d'induction de gamme et de machine relatives à l'ordonnancement cyclique.

3.4.1 Calcul de la marge

Considérons les deux opérations op_i et op_j liées par induction de gamme ou induction de machine. La marge entre ces deux opérations est calculée comme suit dans les deux cas d'induction :

$$marge_{i,j} = Ep(op_j) - (Ep(op_i) + D(op_i)) \quad (\text{II.1})$$

3.4.2 Calcul du retard

Le retard peut être propagé de deux manières définies par la conception d'un système de production : soit l'induction de gamme ou l'induction de machine.

▷ Induction de gamme

Soit op_i et op_j deux opérations successives liées par induction de gamme. Le retard provoqué par l'opération op_i sera totalement appliqué à l'opération op_j en absence de marge entre ces deux opérations, sinon une partie ou la totalité du retard sera consommée par la marge.

Soit la fonction δ définie comme suit : pour tout $x \in \mathbb{R}$, $\delta(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$

$$retard(op_j) = \delta(retard(op_i) - marge_{i,j}) \quad (\text{II.2})$$

▷ Induction de machine

Nous rappelons qu'une opération correspond à l'exécution d'une fonction faite par une machine sur une pièce. Une machine peut effectuer plusieurs types d'opération. Ainsi, le principe de l'induction du retard dans une machine dépend de type de l'opération traitée. S'il s'agit de deux opérations de même type, alors la deuxième opération a le même retard propre que la première, en plus du retard induit propagé s'il existe.

$$retard(op_j) = retard(op_i) + \delta(retard(op_i) - marge_{i,j}) \quad (\text{II.3})$$

S'il s'agit de deux opérations de types différents, alors seul le retard induit est propagé.

$$retard(op_j) = \delta(retard(op_i) - marge_{i,j}) \quad (\text{II.4})$$

4 Diagnostic par chaînage double

Afin d'illustrer l'approche de diagnostic par chaînage double, nous exécutons un scénario de défaillance dont l'objectif est l'identification de la cause première du symptôme de défaillance détecté.

4.1 Tests et résultats du diagnostic

La validation expérimentale de l'approche proposée concerne un SFPM sous ordonnancement cyclique à plusieurs types de pièce, comme illustré dans le chapitre I section 2.4.1.

Nous considérons le scénario suivant : la pièce P_{13} a un retard par rapport à sa date de sortie prévisionnelle donnée par l'ordonnancement cyclique. Ce retard détecté est de 3 unités de temps ($\varepsilon=3$ u.t.). Conformément au principe de diagnostic décrit précédemment, trois hypothèses de défaillance sont générées :

- H1 (Cas d'une défaillance propre) : La dernière opération de la pièce P_{13} , notée op_{33} est défaillante. Il s'agit d'une défaillance propre, notée : $\langle P_{13}|M_2|op_{33}(\varepsilon_1)\rangle$.
- H2 (Cas d'une induction de gamme) : La deuxième opération de la pièce P_{13} , notée op_{32} est défaillante. Alors que l'opération op_{33} subit un retard induit par l'opération précédente op_{32} de la même pièce P_{13} .
- H3 (Cas d'une induction de machine) : La dernière opération de la pièce P_{12} , notée op_{23} est défaillante. Alors que l'opération op_{33} subit un retard induit par l'opération précédente op_{23} de la même machine M_2 . Ces deux opérations sont de même types et sont séparées d'une marge de 1 u.t..

Le principe de diagnostic et la propagation des contraintes temporelles entre les nœuds de l'arbre de résolution sont illustrés par la figure II.5. Cette figure représente l'arbre des hypothèses obtenues par chaînage arrière lancé à partir du symptôme initiateur $\langle P_{13}|M_2|op_{33}(\varepsilon_1)\rangle$. Chaque nœud de l'arbre de résolution modélise une hypothèse de défaillance propre qui doit être confirmée ou réfutée par la suite. La direction des arcs reliant ces nœuds indique le sens de propagation des retards par induction de gamme ou de machine. En analysant la structure de l'arbre, on constate que le nœud racine $\langle P_{13}|M_2|op_{33}(\varepsilon_1)\rangle$ (hypothèse H1) peut être procuré par le nœud $\langle P_{13}|M_1|op_{32}(\varepsilon_2)\rangle$ (hypothèse H2) par induction de gamme ou par le nœud $\langle P_{12}|M_2|op_{23}(\varepsilon_3)\rangle$ (hypothèse H3) par induction de machine.

Concernant le calcul du retard, si nous considérons par exemple que la cause du symptôme initiateur est l'hypothèse H2, alors dans ce cas nous avons uniquement une induction de gamme liée à l'absence de marge entre les opérations op_{32} et op_{33} réalisées sur la pièce P_{13} . Par conséquent les retards de ces deux opérations sont égaux : $\varepsilon_1 = \varepsilon_2$. Par contre si nous considérons l'hypothèse H3, nous devons tenir compte à la fois de la défaillance propre de op_{33} puisqu'il s'agit d'un même type d'opérations et de l'induction du retard dans la machine M_2 . Etant donné la marge de 1 u.t. qui existe entre ces deux opérations dans le circuit relatif à la machine M_2 , il ne peut y avoir une induction du retard que si la marge entre les deux opérations est consommée. D'après l'équation II.3, pour avoir le retard détecté du symptôme initiateur ($\varepsilon_1 = 3$), l'opération op_{23} de l'hypothèse H3 $\langle P_{12}|M_2|op_{23}(\varepsilon_3) \rangle$ devrait avoir un retard égale à 2 u.t. soit $\varepsilon_3 = 2$.

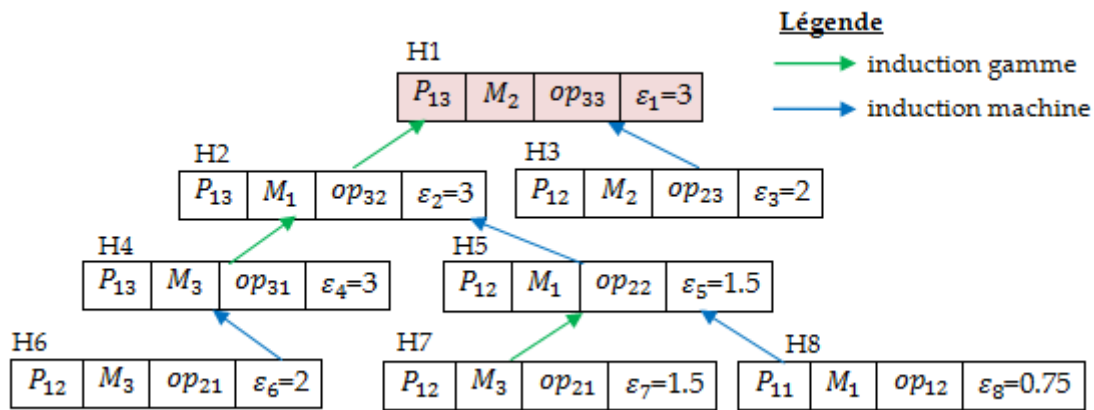


Figure II.5 – Arbre d'hypothèses généré par chaînage arrière

Le développement de l'arbre des hypothèses par chaînage arrière se poursuit jusqu'à ce que l'on arrive à des hypothèses qui peuvent être réfutées à la date courante (opération observée et non défaillante), ou bien à ce que tous les nœuds du cycle courant soient traités. On remarque que la génération de l'arbre des hypothèses est s'arrêtée au niveau de la feuille H3 : $\langle P_{12}|M_2|op_{23}(\varepsilon_3) \rangle$. Cette hypothèse H3 est tout de suite réfutée. En effet, si elle était vérifiée, un retard aurait dû être détecté pour la pièce P_{12} à une date antérieure à celle du symptôme initiateur du scénario étudié.

Concernant l'hypothèse H2 : $\langle P_{13}|M_1|op_{32}(\varepsilon_2) \rangle$, nous ne pouvons pas la valider directement car l'opération est supposée non observable. Par contre, si nous admettons cette hypothèse (défaillance propre de l'opération op_{32}), nous pouvons prédire toutes ses conséquences par chaînage avant. Nous constatons que l'hypothèse H2 aurait également pour conséquence l'observation

d'un retard relatif à la sortie de la pièce P_{21} . En effet la date de sortie de cette pièce est postérieure à celle de détection du symptôme initiateur. Nous pouvons juste prédire que si l'hypothèse H2 est vraie alors elle sera confirmée par l'occurrence d'un nouveau symptôme relatif à P_{21} dans le cycle suivant. La validation de l'hypothèse H2 par chaînage avant génère l'ensemble des hypothèses de la figure II.6.

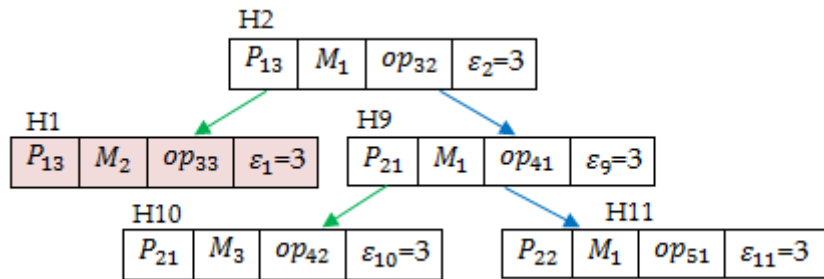


Figure II.6 – Validation de l'hypothèse H2 par chaînage avant

L'analyse des conséquences des hypothèses de l'arbre de résolution se poursuit par chaînage avant. Les hypothèses restantes seront réfutées par l'absence d'observation du retard à une date antérieure de celle du symptôme initiateur. Par exemple l'hypothèse H4 $\langle P_{13}|M_3|op_{31}(\varepsilon_4) \rangle$ implique que l'on devrait avoir eu un symptôme relatif à un retard à la sortie de la pièce P_{22} . Cette hypothèse est donc réfutée puisque la date de sortie de cette pièce P_{22} est égale à celle du symptôme initiateur.

L'exécution de l'approche de diagnostic indirect proposée donne le résultat de deux hypothèses possibles H1 ou H2. En conséquence, il nous faudra attendre la date de sortie de P_{21} (dans le cycle suivant) pour conclure entre ces deux hypothèses.

4.2 Analyse du principe de résolution

L'évaluation de l'approche proposée permet de tirer un certain nombre d'enseignements concernant son efficacité. D'une part, la capacité de détection des défauts est fortement dépendante de l'existence de circuits critiques dans l'ordonnancement. Ce problème peut être partiellement résolu, puisque les caractéristiques intrinsèques des ordonnancements cycliques font qu'il existe toujours au moins un circuit de ce type : celui relatif à la machine menante. D'autre part, la capacité de la localisation et l'identification des défauts est fortement dépendante des points d'observabilités définis pour le système : ceux relatives aux dates de sorties des pièces. Une autre conclusion de l'approche concerne la taille importante des arbres de ré-

solution. En effet nous admettons qu'il faut mémoriser les nœuds pour continuer la résolution durant les cycles suivants. De plus, l'arbre des hypothèses peut contenir des nœuds redondants. Ceci s'explique par le fait qu'un même nœud peut propager le retard par induction de gamme et induction de machine. Nous remarquons dans la figure II.5 que les hypothèses H6 et H7 correspondent à la même hypothèse avec des retards différents. En effet, l'hypothèse H4 peut être causée par l'hypothèse H6 par induction de machine. De même, l'hypothèse H5 peut être causée par l'hypothèse H7 par induction de gamme.

Dans la partie suivante nous analysons l'efficacité du diagnostic par chaînage double, et la capacité de cette approche pour répondre aux exigences du diagnostic en ligne.

5 Efficacité du diagnostic

L'évaluation de l'approche proposée dans la section précédente et présentée dans [Fakhfakh *et al.*, 2012] a montré la capacité à diagnostiquer un SFPM en se limitant à l'observation des sorties des pièces. Toutefois, le diagnostic peut ne pas converger vers une solution unique. En effet, nous constatons à travers l'exécution de scénario étudié que les observations d'un seul cycle ne sont pas toujours suffisantes pour identifier la cause première d'une défaillance (le diagnostic a donné deux solutions possibles).

Il est alors nécessaire de poursuivre le diagnostic dans les cycles suivants de l'ordonnancement cyclique afin d'espérer raffiner l'ensemble des solutions. En se basant sur le principe de l'approche proposée, il faut reprendre la construction des arbres de résolution à chaque nouvelle observation afin de construire le chaînage des nœuds (par induction de gamme et de machine) entre les cycles. Ensuite il s'agit d'appliquer le chaînage avant sur le nouvel arbre de résolution construit. Le fait de reprendre le diagnostic à chaque nouvelle observation peut être coûteux en termes de mémoire et de temps d'exécution.

Afin de pallier les insuffisances de l'approche par chaînage double, il est nécessaire d'envisager une stratégie incrémentale de diagnostic en ligne. L'objectif de la section suivante est de concevoir un diagnostic incrémental ayant la capacité de mettre à jour les résultats du diagnostic à chaque nouvelle observation afin de raffiner les solutions et aboutir à la cause première de défaillance. La difficulté intrinsèque d'une approche incrémentale réside dans la conception des liaisons entre les cycles. En effet, il ne suffit pas de dupliquer les cycles mais il faut créer automatiquement les liens entre les cycles faute de quoi ils deviennent indépendants et par

conséquent le diagnostic devient caduc.

Afin de répondre aux exigences de l'approche incrémentale et de s'adapter à l'ajout des contraintes en cours d'exécution de diagnostic, nous adoptons la technique de programmation par contraintes. Le principe de la programmation par contraintes est décrit dans le chapitre suivant.

6 Conclusion

Nous avons présenté au cours de ce chapitre notre approche de diagnostic indirect par analyse quantitative de flux de production basée sur un chaînage double. Cette approche repose sur la génération d'hypothèses par chaînage arrière suivi d'une étape de validation par chaînage avant. L'exécution de certains scénarios a montré la nécessité de poursuivre le diagnostic dans les cycles suivant afin de raffiner les hypothèses et identifier la source de défaillance. Une solution de diagnostic incrémental sera proposée dans le chapitre suivant.

Chapitre III

Approche par contraintes pour le diagnostic incrémental

Sommaire

1	Programmation par contraintes	51
2	Paramètres du diagnostiqueur	52
2.1	Opération	52
2.2	Données	53
2.3	Variables	53
3	Contraintes du diagnostiqueur	53
3.1	Types de retards	54
3.2	Dates de fin des opérations	56
3.3	Contraintes de précédences entre les opérations	56
3.4	Contraintes d'inductions de retards	58
4	Diagnostic incrémental	60
4.1	Mécanisme de résolution	61
4.2	Taille du problème	62
5	Application du diagnostic à un SFPM sous une commande cyclique .	63
5.1	Choix de l'outil de résolution	63
5.2	Description du problème pour le modèle proposé	64
5.3	Tests et résultats du diagnostic indirect et incrémental	64
5.4	Analyses qualitatives des résultats	70
6	Etude de performance du diagnostic	71
6.1	Evaluation du diagnostic d'un seul cycle	72
6.2	Evaluation du diagnostic incrémental complet	74
6.3	Stratégies de recherche	76

6.4	Comparaison avec la littérature	79
7	Conclusion	82

Nous présentons dans cette section notre nouvelle méthode de diagnostic incrémental d'un SFPF soumis à un ordonnancement cyclique et dans le contexte de défaillance unique. Cette approche se base sur la construction d'un modèle mathématique décrivant les paramètres d'un SFPF et modélisant les contraintes d'induction de gammes et de machines. Ces contraintes doivent tenir compte de l'enchaînement entre les cycles et le cas particulier de chevauchement des cycles tout en gardant l'aspect cyclique de l'ordonnancement. Finalement, en cas de détection d'un retard en sortie du SFPF, le diagnostic consistera en la résolution de ce modèle mathématiques de manière à évaluer le retard propre de chaque opération afin d'expliquer le retard du produit en sortie. La valeur du retard propre permettra d'identifier et isoler la source de défaut détecté.

Nous rappelons que toute dégradation du flux de production est synonyme de l'occurrence d'un défaut de fonctionnement au sein du système. Quelquefois ces défauts ne sont pas directement détectables à la sortie du système, à cause des marges temporelles qui existent entre certaines opérations. La difficulté majeure de ce problème c'est que nous n'avons aucune observation sur le comportement interne du système. Cela signifie que nous n'avons pas de connaissances sur le fonctionnement réel des machines.

Le principe de l'approche incrémentale sera détaillé tout au long de ce chapitre.

1 Programmation par contraintes

La programmation par contraintes est une technique de résolution des problèmes combinatoires issue de la programmation logique et de l'intelligence artificielle. Elle a été utilisée pour une très grande variété de domaines d'application tels que les problèmes d'ordonnancement [Korbaa *et al.*, 2000] et de diagnostic des défauts. La démarche contient deux étapes : la modélisation puis la résolution du problème. Dans la modélisation, un problème est représenté à l'aide de variables et de contraintes (et potentiellement une ou plusieurs fonction(s) d'objectif s'il s'agit de problème d'optimisation). Chaque variable est munie d'un domaine définissant l'ensemble des valeurs possibles pour cette variable. Une contrainte exprime une propriété qui doit être satisfaite par un ensemble de variables. Tandis que la résolution est caractérisée par l'utilisation des contraintes du problème d'une manière active afin de limiter la taille de l'espace de recherche (principe de la propagation de contraintes). La résolution est assurée, généralement, par la recherche arborescente avec retour arrière (backtracking).

Un solveur de contraintes calcule une solution en instanciant chacune des variables à une valeur satisfaisant simultanément toutes les contraintes. De nombreuses techniques issues de la recherche opérationnelle, de la programmation mathématique ou même de la recherche locale sont appliquées grâce à la séparation entre un langage de modélisation déclaratif et les algorithmes employés durant la résolution.

La programmation par contraintes bénéficie de l'avantage que le langage de modélisation adopté est un langage proche de l'utilisateur. En fait, la modélisation peut contenir des relations non linéaires. Cependant, lors de la résolution, l'ordre de prise en compte des contraintes du problème influe largement sur l'efficacité de la résolution.

2 Paramètres du diagnostiqueur

Dans cette partie, les paramètres du diagnostiqueur sont détaillés afin de concevoir plus tard le modèle de diagnostic.

2.1 Opération

Une opération correspond à l'exécution d'une fonction réalisée par une machine sur une pièce. Nous associons à chaque opération le numéro du cycle de production β dans lequel l'opération commence. Nous considérons que le premier cycle de production correspond à $\beta = 1$. Soit β le cycle courant qui contient une pièce en retard. Ayant la date de sortie absolue d'une pièce, nous pouvons calculer le numéro du cycle β dans lequel la pièce sort du système de production.

Soit $\beta = \left\lceil \frac{\text{Date de sortie absolue}}{CT} \right\rceil + 1$ (partie entière de la division).

Nous rappelons que CT correspond au temps du cycle de production fixé par l'ordonnancement cyclique. Chaque cycle commence à 0 (module CT).

La $i^{\text{ème}}$ opération de la gamme α est notée par : $op_{\alpha i}^{\beta}$.

Les paramètres du modèle définissent toutes les informations nécessaires à la modélisation du système de production. Nous spécifions dans la suite les données et les variables du modèle.

2.2 Données

Les données d'entrées du modèle sont fournies par le schéma prévisionnel du système défini par l'ordonnancement cyclique. Ces données représentent le comportement normal du système. Nous connaissons les dates d'entrées et de sorties prévisionnelles des opérations.

$Ep(op_{\alpha i}^{\beta})$, $Sp(op_{\alpha i}^{\beta})$ et $D(op_{\alpha i}^{\beta})$ sont les données prévisionnelles du système, données par l'ordonnancement cyclique établi hors-ligne. Les données et leurs domaines de définition sont représentés dans le tableau III.1. Il est à noter que la date de sortie de la dernière opération de chaque pièce est considérée comme étant une donnée du problème puisqu'il s'agit d'une observation en sortie du système.

Données	Explication	Domaine de définition
$Ep(op_{\alpha i}^{\beta})$	Date de début prévisionnelle de l'opération $op_{\alpha i}^{\beta}$	$[0, CT[(\text{modulo } CT)$
$Sp(op_{\alpha i}^{\beta})$	Date de fin prévisionnelle de l'opération $op_{\alpha i}^{\beta}$	$]0, CT](\text{modulo } CT)$
$D(op_{\alpha i}^{\beta})$	Durée prévisionnelle de l'opération $op_{\alpha i}^{\beta}$	$]0, CT](\text{modulo } CT)$

Tableau III.1 – Les données du modèle relatives à une opérations donnée

2.3 Variables

Les dates et durées réelles de chaque opération représentent les variables du modèle. L'écart entre les données prévisionnelles et les dates réelles correspond à un type de retard.

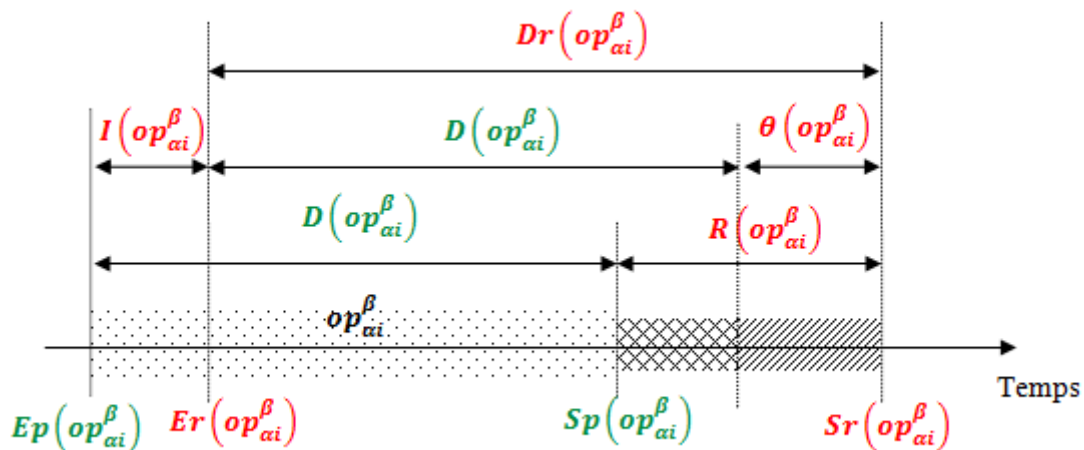
$Er(op_{\alpha i}^{\beta})$, $Sr(op_{\alpha i}^{\beta})$, $Dr(op_{\alpha i}^{\beta})$, $\theta(op_{\alpha i}^{\beta})$, $I(op_{\alpha i}^{\beta})$ et $R(op_{\alpha i}^{\beta})$ sont les variables du modèle. Ces variables et leurs domaines de définition sont présentées dans le tableau III.2.

Les relations entre les différents paramètres du modèle sont décrites par la figure III.1 et expliquées dans la suite.

3 Contraintes du diagnostiqueur

Dans cette partie, nous définissons toutes les contraintes caractérisant les différents types de retards ainsi que les relations temporelles entre les opérations reliées par induction de gammes

Variables	Explication	Domaine de définition
$Er(op_{\alpha i}^{\beta})$	Date de début réelle de l'opération $op_{\alpha i}^{\beta}$	$[0, CT[(\text{modulo } CT)$
$Sr(op_{\alpha i}^{\beta})$	Date de fin réelle de l'opération $op_{\alpha i}^{\beta}$	$]0, CT](\text{modulo } CT)$
$Dr(op_{\alpha i}^{\beta})$	Durée réelle de l'opération $op_{\alpha i}^{\beta}$	\mathbb{R}_+^*
$\theta(op_{\alpha i}^{\beta})$	Retard propre de l'opération $op_{\alpha i}^{\beta}$	$[0, CT]$
$I(op_{\alpha i}^{\beta})$	Retard induit de l'opération $op_{\alpha i}^{\beta}$	\mathbb{R}_+
$R(op_{\alpha i}^{\beta})$	Retard total de l'opération $op_{\alpha i}^{\beta}$	\mathbb{R}_+

Tableau III.2 – Les variables du modèle relatives à une opérations donnée

Figure III.1 – Les caractéristiques des dates et des durées d'une opération

et induction de machines.

3.1 Types de retards

A partir des définitions précédentes, nous considérons qu'un symptôme de défaut est détecté par l'observation d'un écart entre la date de sortie réelle d'une pièce et celle prévisionnelle. Cet écart représente le retard de la pièce qui résulte de l'augmentation de la durée opératoire d'une opération de la gamme de la pièce défailante. Nous distinguons deux types de retards : des

retards propres et des retards induits.

3.1.1 Retard propre

Le retard propre d'une opération caractérise une opération dont la durée opératoire a augmentée. Il indique donc si l'opération considérée est la cause première du symptôme de défaut détecté. Le retard propre d'une opération est normalement égal à zéro en cas d'absence de défaillance. En cas de défaillance du système le retard propre devient supérieur à zéro. Au fil du temps, le retard ne peut que croître. Le retard propre est défini dans l'équation III.1 par la différence entre la durée réelle et celle prévisionnelle d'une opération.

$$\theta(op_{\alpha i}^{\beta}) = Dr(op_{\alpha i}^{\beta}) - D(op_{\alpha i}^{\beta}) \quad (\text{III.1})$$

3.1.2 Retard induit

Le retard induit indique que l'opération traitée a été décalée dans le temps par rapport à sa date prévisionnelle de début. Il est causé par l'appartenance de l'opération à un circuit soumis à une défaillance en amont de l'opération considérée. Il est défini dans l'équation III.2 par la différence entre la date réelle de début de l'opération et sa date prévisionnelle de début.

$$I(op_{\alpha i}^{\beta}) = Er(op_{\alpha i}^{\beta}) - Ep(op_{\alpha i}^{\beta}) \quad (\text{III.2})$$

3.1.3 Retard total

Le retard total peut être défini de deux manières. D'une part, il est défini dans l'équation III.3 par la différence entre la date réelle de fin de l'opération et sa date prévisionnelle de fin.

$$R(op_{\alpha i}^{\beta}) = Sr(op_{\alpha i}^{\beta}) - Sp(op_{\alpha i}^{\beta}) \quad (\text{III.3})$$

Cette définition est utile pour détecter la défaillance d'une pièce en sortie par le calcul du retard total de sa dernière opération, puisque la date réelle de fin de la dernière opération est connue du fait de l'observation du système. En effet, si le retard total $R(op_{\alpha i}^{\beta})$ est strictement supérieur à zéro, cela permet de détecter un comportement défaillant du système. Le retard total est défini aussi dans l'équation III.4 par la somme du retard propre et du retard induit d'une opération.

$$R(op_{\alpha i}^{\beta}) = \theta(op_{\alpha i}^{\beta}) + I(op_{\alpha i}^{\beta}) \quad (\text{III.4})$$

3.2 Dates de fin des opérations

Une défaillance dans le système peut causer une différence entre les dates réelles et celles prévisionnelles. La durée opératoire prévisionnelle est définie dans l'équation III.5 par la différence entre la date de fin prévisionnelle et la date de début prévisionnelle. De la même façon nous définissons dans l'équation III.6 la durée opératoire réelle qui est la différence entre la date de fin réelle et la date de début réelle. Nous exprimons la durée opératoire réelle d'une opération par la somme de sa durée opératoire prévisionnelle et de son retard propre. Cela signifie que la durée opératoire réelle est forcément supérieure ou égale à la durée opératoire prévisionnelle.

$$Sp(op_{\alpha i}^{\beta}) = Ep(op_{\alpha i}^{\beta}) + D(op_{\alpha i}^{\beta}) \quad (\text{III.5})$$

$$Sr(op_{\alpha i}^{\beta}) = Er(op_{\alpha i}^{\beta}) + D(op_{\alpha i}^{\beta}) + \theta(op_{\alpha i}^{\beta}) \quad (\text{III.6})$$

3.3 Contraintes de précédences entre les opérations

Soit $op_{\alpha+i+}^{\beta+}$ l'opération qui succède à l'opération $op_{\alpha i}^{\beta}$ sur la même gamme α ou la même machine. Les deux paramètres β^+ et α^+ sont définis de la manière suivante :

- Si les deux opérations $op_{\alpha+i+}^{\beta+}$ et $op_{\alpha i}^{\beta}$ commencent dans le même cycle alors $\beta^+ = \beta$.
- Sinon, si l'opération $op_{\alpha+i+}^{\beta+}$ commence dans le cycle suivant alors $\beta^+ = \beta + 1$.
- Dans le cas où les deux opérations appartiennent à la même gamme, on a alors $\alpha^+ = \alpha$.

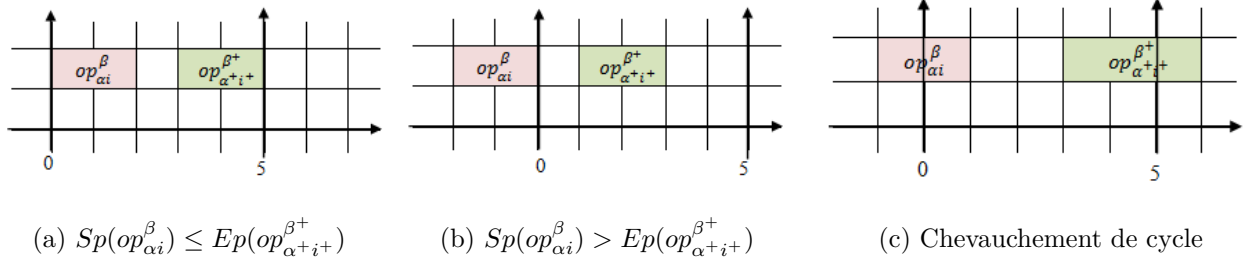
Nous considérons que $op_{\alpha+i+}^{\beta+}$ et $op_{\alpha i}^{\beta}$ deux opérations successives d'une même pièce. Sur une échelle de temps absolue, il est évident que la date de fin de l'opération $op_{\alpha i}^{\beta}$ précède la date de début de l'opération $op_{\alpha+i+}^{\beta+}$. Puisque les paramètres du diagnostiqueur sont définis modulo CT, il est important de tenir compte de tous les cas particuliers, notamment le cas de chevauchement de cycles.

Soit la variable Δ définie comme suit : $\Delta = \begin{cases} 0, & \beta^+ = \beta \\ 1, & \beta^+ = \beta + 1 \end{cases}$

Dans le cas général, la contrainte de précédence entre deux opérations successives est définie comme suit dans l'équation III.7 :

$$Sr(op_{\alpha i}^{\beta}) \leq Er(op_{\alpha+i+}^{\beta+}) + \Delta \times CT \quad (\text{III.7})$$

L'exemple de la figure III.2(c) illustre le cas de chevauchement de cycle de l'opération $op_{\alpha i}^{\beta}$. En effet, cette dernière commence dans le cycle β et se termine dans le cycle $\beta + 1$.


Figure III.2 – Exemple de contraintes de précédences

Considérons comme exemple les ordonnancements de la figure III.2 dans lesquels le temps de cycle est fixé à 5 u.t.

- Si $op_{\alpha+i+}^{\beta+}$ et $op_{\alpha i}^{\beta}$ commencent dans le même cycle ; sachant que le temps de cycle est fixé à 5 ; alors $Sp(op_{\alpha i}^{\beta}) \leq Ep(op_{\alpha+i+}^{\beta+})$ (comme le montre la Figure III.2(a)).
- Sinon si $op_{\alpha+i+}^{\beta+}$ commence dans le cycle suivant, alors $Sp(op_{\alpha i}^{\beta}) \leq Ep(op_{\alpha+i+}^{\beta+}) + CT$ (comme le montre les figures III.2(b) et III.2(c)). Ceci s'explique par le fait que les domaines de définition de ces paramètres sont donnés modulo CT, i.e que les dates sont relatives à un cycle.

L'équation III.7 est valide quelque soit le type d'ordonnement cyclique : avec ou sans chevauchement de cycle (des opérations de gamme sur plusieurs cycles).

Dans le cas où l'ordonnement ne contient pas d'opérations avec chevauchement de cycle, l'équation III.7 peut être représentée par l'équation III.8.

- Si $op_{\alpha+i+}^{\beta+}$ et $op_{\alpha i}^{\beta}$ commencent dans le même cycle (comme le montre la Figure III.2(a)) alors $Sp(op_{\alpha i}^{\beta}) \leq Ep(op_{\alpha+i+}^{\beta+})$.
- Sinon si $op_{\alpha+i+}^{\beta+}$ commence dans le cycle suivant (comme le montre la figure III.2(b)) alors $Sp(op_{\alpha i}^{\beta}) > Ep(op_{\alpha+i+}^{\beta+})$.

Ces deux cas sont pris en compte par l'équation III.8.

$$Sr(op_{\alpha i}^{\beta}) \leq \begin{cases} Er(op_{\alpha+i+}^{\beta+}) & \text{si } Ep(op_{\alpha+i+}^{\beta+}) \geq Sp(op_{\alpha i}^{\beta}) \\ Er(op_{\alpha+i+}^{\beta+}) + CT & \text{si } Ep(op_{\alpha+i+}^{\beta+}) < Sp(op_{\alpha i}^{\beta}) \end{cases} \quad (\text{III.8})$$

Dans le cas d'absence de marge entre les deux opérations $op_{\alpha+i+}^{\beta+}$ et $op_{\alpha i}^{\beta}$ alors l'équation III.8 reste toujours valide. En effet :

- Si les deux opérations commencent dans le même cycle alors la date de fin de la première opération et la date de début de deuxième coïncident : $Ep\left(op_{\alpha+i+}^{\beta+}\right) = Sp\left(op_{\alpha i}^{\beta}\right)$,
- Sinon si la deuxième opération commence dans le cycle suivant alors : $Ep\left(op_{\alpha+i+}^{\beta+}\right) < Sp\left(op_{\alpha i}^{\beta}\right)$, puisque $Sp\left(op_{\alpha i}^{\beta}\right) = CT$ et $Ep\left(op_{\alpha+i+}^{\beta+}\right) = 0$.

Par définition de l'ordonnancement cyclique, on ne peut pas avoir une marge supérieure ou égale à la durée d'un cycle, sinon le cycle serait vide. Ainsi, cette inégalité est toujours valide : $Ep\left(op_{\alpha+i+}^{\beta+}\right) - Sp\left(op_{\alpha i}^{\beta}\right) < CT$.

En effet, si $Ep\left(op_{\alpha+i+}^{\beta+}\right) - Sp\left(op_{\alpha i}^{\beta}\right) > 0$, alors : $0 < \frac{Er\left(op_{\alpha+i+}^{\beta+}\right) - Sr\left(op_{\alpha i}^{\beta}\right)}{CT} < 1$

$$\left[\frac{Er\left(op_{\alpha+i+}^{\beta+}\right) - Sr\left(op_{\alpha i}^{\beta}\right)}{CT} \right] = \begin{cases} 0 & \text{si } Ep\left(op_{\alpha+i+}^{\beta+}\right) \geq Sp\left(op_{\alpha i}^{\beta}\right) \\ -1 & \text{si } Ep\left(op_{\alpha+i+}^{\beta+}\right) < Sp\left(op_{\alpha i}^{\beta}\right) \end{cases}$$

Nous généralisons alors ces deux cas par l'équation III.9.

$$Sr\left(op_{\alpha i}^{\beta}\right) \leq Er\left(op_{\alpha+i+}^{\beta+}\right) + CT \times \left[\frac{Er\left(op_{\alpha+i+}^{\beta+}\right) - Sr\left(op_{\alpha i}^{\beta}\right)}{CT} \right] \quad (\text{III.9})$$

3.4 Contraintes d'inductions de retards

Nous détaillons dans la suite toutes les contraintes d'inductions de différents types de retards. Cette partie est importante pour garantir la propagation des contraintes temporelles relatives au phénomène d'induction du retard pour une succession d'opérations.

3.4.1 Contrainte liée au retard propre

Soit $op_{\alpha^*i^*}^{\beta^*}$ l'opération qui succède à l'opération $op_{\alpha i}^{\beta}$ sur la même machine. Supposons que ces deux opérations soient de même type.

$$Si \ \theta\left(op_{\alpha i}^{\beta}\right) > 0 \ \text{alors} \ \theta\left(op_{\alpha^*i^*}^{\beta^*}\right) = \theta\left(op_{\alpha i}^{\beta}\right) \quad (\text{III.10})$$

Selon les hypothèses A2 et A7, nous supposons que le retard propre d'une opération demeure constant après son apparition. Cela signifie que si une opération d'une machine est défaillante

alors les mêmes opérations de la même machine qui succèdent cette opération ont le même retard propre que la première. Cette propriété est formulée par l'équation III.10. Dans la réalité, dans le cas d'une défaillance progressive, le retard propre d'une opération ne peut que s'accroître au fil du temps. C'est ce qui explique d'ailleurs qu'à un moment, il devient perceptible car ayant atteint une valeur qui perturbe l'ordonnancement prévisionnel en consommant les marges résiduelles entre opérations. Mais notre hypothèse d'un retard constant a du sens dans la mesure où sur quelques cycles de fonctionnement d'un ordonnancement cyclique, la croissance de ce retard est en général peu sensible.

3.4.2 Contrainte liée au retard induit

Le retard induit peut être causé par une induction de gamme ou une induction de machine. Soit $op_{\alpha-i}^{\beta-}$ l'opération qui précède l'opération $op_{\alpha i}^{\beta}$ sur une gamme ou une machine. Une portion du retard de l'opération $op_{\alpha-i}^{\beta-}$ est consommée par la marge qui peut exister entre les deux opérations. Par conséquent, si le retard induit est inférieur ou égale à la marge alors il n'y aura aucune induction du retard sur l'opération $op_{\alpha i}^{\beta}$ (voir figure III.3(a)). Sinon, le retard induit est défini par la différence entre le retard de l'opération antérieure et la marge (voir figure III.3(b)).

Soit la fonction δ définie comme suit : pour tout $x \in \mathbb{R}$, $\delta(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$

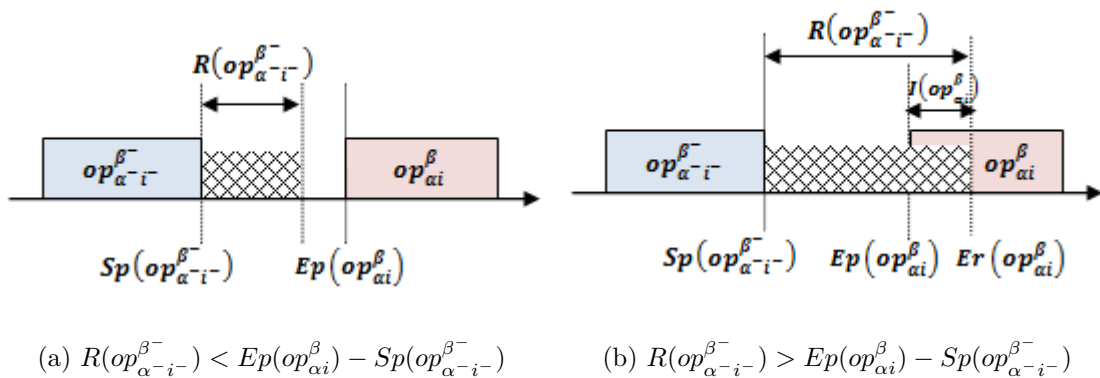


Figure III.3 – Contraintes liées au retard induit

▷ Induction de gamme

L'induction de gamme est une conséquence de la contrainte de séquençement des opérations dans une gamme. Cette contrainte exige le respect de l'ordre d'exécution des opérations dans la production d'une pièce dans les systèmes de type job shop. Ce type de retard est dû à l'appartenance de l'opération considérée à une gamme n'ayant pas assez de marge pour consommer le retard produit par une opération antérieure. Soit la variable $I_g(op_{\alpha i}^{\beta})$ qui définit le retard induit par induction de gamme, elle est définie par l'équation III.11.

$$I_g(op_{\alpha i}^{\beta}) = \delta \left(R(op_{\alpha^- i^-}^{\beta^-}) - Ep(op_{\alpha i}^{\beta}) + Sp(op_{\alpha^- i^-}^{\beta^-}) \right) \quad (\text{III.11})$$

▷ Induction de machine

De même, l'induction de machine est définie par la contrainte de séquençement des opérations affectées à une machine donnée. En effet, une machine ne peut commencer une opération seulement après avoir terminé l'exécution de l'opération précédente. Soit la variable $I_m(op_{\alpha i}^{\beta})$ qui définit le retard induit par induction de machine.

$$I_m(op_{\alpha i}^{\beta}) = \delta \left(R(op_{\alpha^- i^-}^{\beta^-}) - Ep(op_{\alpha i}^{\beta}) + Sp(op_{\alpha^- i^-}^{\beta^-}) \right) \quad (\text{III.12})$$

Si une opération est soumise à un premier retard par induction de gamme et un second retard par induction de machine, alors le retard résultat est le maximum des deux retards (cf. équation III.13).

$$I(op_{\alpha i}^{\beta}) = \text{Max} \left(I_g(op_{\alpha i}^{\beta}), I_m(op_{\alpha i}^{\beta}) \right) \quad (\text{III.13})$$

3.4.3 Contrainte de défaillance unique

Notre modèle de diagnostic se base sur l'hypothèse de défaillance unique. Soit $\overline{op_{\alpha i}^{\beta}}$ l'opération qui succède l'opération $op_{\alpha i}^{\beta}$ et qui est de type différent. Selon l'hypothèse A4, une seule opération peut correspondre à la source de défaillance.

$$\text{Si } \theta(op_{\alpha i}^{\beta}) > 0 \quad \text{Alors } \theta(\overline{op_{\alpha i}^{\beta}}) = 0 \quad (\text{III.14})$$

4 Diagnostic incrémental

Dans le contexte du diagnostic indirect où seules les sorties des pièces sont observées, les données d'un seul cycle de production peuvent ne pas être suffisantes pour que le diagnostic

converge vers une solution unique. L'objectif de cette section est de concevoir une méthode qui permet d'ajouter régulièrement au diagnostic chaque nouvelle observation obtenue au cours de la production.

4.1 Mécanisme de résolution

Le principe de diagnostic incrémental que nous proposons dans l'algorithme 4 consiste à compter le nombre de solutions obtenues après chaque résolution du modèle pour un cycle. Si la résolution aboutie à une solution unique alors le diagnostic est considéré complet avec l'identification de l'opération source de la défaillance. Sinon, un nouveau cycle est créé et ajouté au modèle du diagnostic en cours. Dans l'enchaînement entre les cycles, il ne suffit pas de dupliquer les cycles mais nous créons automatiquement les liens entre les cycles en nous basant sur la modélisation inter-cycle proposée dans la section précédente. A ce niveau, le système de diagnostic contient le modèle (variables, domaine de définition et contraintes) du nouveau cycle créé ainsi que les domaines de définition des variables du cycle précédent obtenus après la première résolution et les contraintes d'induction entre les deux cycles. En effet, après le diagnostic du cycle précédent, on ne mémorise que les instances de variables qui représente une solution potentielle de diagnostic. De ce fait, il ne s'agit pas de reprendre la résolution dès le premier cycle du diagnostic, mais de raffiner l'ensemble des solutions générées par le diagnostic du cycle précédent. Par la suite, le diagnostic permet de poursuivre la résolution jusqu'à que le système converge vers une solution unique (puisque nous nous situons dans le contexte de défaillance unique).

La difficulté du diagnostic incrémental a été résolue par la définition d'une fenêtre temporelle glissante du cycle modélisant l'ordonnancement qui représente fidèlement le diagramme de Gantt. Cette fenêtre est définie par les opérations des pièces produites dans un cycle d'ordonnancement. A chaque nouvelle observation, la fenêtre temporelle avance pour tenir compte de la nouvelle pièce produite par le système. L'avantage de cette modélisation réside dans le fait qu'elle est indépendante de la représentation temporelle du cycle.

Début

1. Créer les pièces du cycle contenant le symptôme initiateur.
 2. Initialiser le nombre de solution à 0
 3. Lancer la résolution
 4. **Si** (le nombre de solution = 1) **Alors**
 - 4.1. Source de défaillance identifiée
 - 4.1. Arrêter le diagnostic**Fin Si**
 5. **Si** (le nombre de solution > 1) **Alors**

Raffiner l'ensemble des solutions :

 - 5.1. Ajouter une nouvelle observation (date de sortie d'une pièce)
 - 5.2. Poursuivre la résolution
 - 5.3. Aller en 4.**Fin Si**
 6. **Si** (le nombre de solution < 1) **Alors**

//opération défaillante dans le cycle précédent

 - 6.1. Créer les pièces du cycle précédent
 - 6.2. Ajouter les observations correspondantes
 - 6.3. Aller en 4.**Fin Si**
- Fin.**

Algorithme 4 – Diagnostic incrémental

4.2 Taille du problème

La complexité du problème étudié se manifeste à deux niveaux, le grand nombre des variables de décision ainsi que le nombre important de contraintes. Le nombre de variables et le nombre de contraintes du modèle dépendent du nombre d'opérations par cycle ainsi que du nombre de cycles pris en compte pour aboutir à la solution. Dans les systèmes manufacturiers de production, chaque pièce est composée de plusieurs opérations. Par conséquent, pour chaque cycle de production, le nombre total d'opérations correspond à la somme des différentes opérations de toutes les pièces considérées.

Comme le montre le Tableau III.1, pour chaque opération il y a 8 variables de décisions correspondantes. Le diagnostic étant basé sur une approche cyclique, à chaque cycle, le modèle développé doit satisfaire les 13 contraintes caractéristiques d'un cycle.

Considérons comme exemple le cas d'étude proposé dans la figure I.5. L'ordonnancement cyclique de cet SFPM donne qu'il existe 13 opérations par cycle (soit 3 pièces composées de 3 opérations et 2 pièces composées de 2 opérations). Par conséquent, pour chaque cycle on trouve 104 variables de décisions (8 variables \times 13 opérations) et 169 contraintes (13 contraintes \times 13 opérations).

5 Application du diagnostic à un SFPM sous une commande cyclique

Dans la section précédente, nous avons présenté un modèle mathématique afin de résoudre le problème du diagnostic indirect et incrémental d'un SFPM sous ordonnancement cyclique. Nous limitons notre observation à la mesure de retards des pièces à la sortie du système de production. Nous supposons que cette observabilité est suffisante pour nous permettre de détecter toute diminution de performance d'un SFPM suite à la défaillance de ses ressources de production. Cette hypothèse revient à supposer que le système est diagnosticable. Ce concept sera étudié dans la partie diagnosticabilité du chapitre suivant.

L'objectif de cette section est d'appliquer le diagnostic proposé à un cas d'étude et résoudre le modèle mathématique afin d'identifier la cause de défaillance qui a perturbé le flux de production par rapport à l'ordonnancement planifié. La résolution du modèle nécessite un solveur adapté aux contraintes du problème. Dans la suite, nous justifierons le choix du solveur utilisé pour la résolution du problème. Ensuite, différents scénarios seront illustrés afin de valider les capacités du modèle pour la mise en œuvre d'un diagnostic indirect et incrémental.

5.1 Choix de l'outil de résolution

Le modèle mathématique de diagnostic proposé dans ce mémoire contient des contraintes non linéaires comme les équations III.8 et III.9 et des contraintes conditionnelles comme les équations III.10, III.11, III.12 et III.14. Aussi, la programmation classique comme la programmation linéaire (PL) et la programmation linéaire en nombres entiers (PLNE) nécessitent la linéarisation des contraintes qui n'est pas toujours simple ou possible.

Afin de résoudre notre modèle nous avons donc besoin d'un outil de résolution de contraintes non linéaires. D'autre part, pour chaque observation, nous avons besoin de générer toutes les

solutions possibles. L’outil de résolution à choisir doit donc nous permettre d’énumérer l’ensemble des solutions valides pour un ensemble d’observations et cohérentes avec l’ensemble des contraintes du modèle.

Nous avons ainsi choisi le solveur « ILOG Solver [Manual, 2010] » qui est une bibliothèque C++ utilisant la programmation par contraintes pour résoudre des problèmes combinatoires complexes. L’un des avantages de ce type de programmation utilisé dans ce solveur réside dans la dissociation entre la représentation du problème, nommé modèle, et sa résolution qui utilise des algorithmes de recherche prédéfinies. Le choix de l’outil est guidé par sa capacité de résolution des contraintes et d’énumération des solutions.

5.2 Description du problème pour le modèle proposé

La production cyclique proposée pour le SFPM présenté dans le premier chapitre I consiste à produire cinq pièces par cycle. Ces pièces sont composées de trois pièces de type P_1 notées $\{P_{11}, P_{12}, P_{13}\}$ et de deux pièces de type P_2 notées $\{P_{21}, P_{22}\}$.

Supposons que le symptôme initiateur qui déclenche le diagnostic est détecté à la sortie de la pièce P_{12} dans le cycle β . Par définition, les opérations liées à cette pièce sont notées par : $P_{12} = \{op_{21}^\beta \mid op_{22}^\beta \mid op_{23}^\beta\}$.

Nous représentons dans le tableau III.3 les cinq prochaines observations après la détection du retard. Nous rappelons que chaque opération qui commence dans le cycle β est nommée avec ce paramètre. Le tableau III.3 montre que les premières données ajoutées au modèle de diagnostic sont les dates de sortie des pièces P_{13} et P_{22} qui sont attendus dans le même cycle que la pièce défectueuse P_{12} .

5.3 Tests et résultats du diagnostic indirect et incrémental

Afin de tester la validité du modèle de diagnostic proposé, nous avons besoin d’identifier la cause première de chaque symptôme de défaillance détecté. Dans la suite, nous allons exécuter deux scénarios de défaillance d’un SFPM défini dans le cas d’étude ci-dessus.

Type de pièce	Pièce	Date prévisionnelle de sortie	Opérations		
P_1	P_{11}	5 u.t. modulo CT	op_{11}^β	$op_{12}^{\beta+1}$	$op_{13}^{\beta+1}$
	P_{12}	8 u.t. modulo CT	$op_{21}^{\beta+1}$	$op_{22}^{\beta+1}$	$op_{23}^{\beta+1}$
	P_{13}	11 u.t. modulo CT	op_{31}^β	op_{32}^β	op_{33}^β
P_2	P_{21}	1 u.t. modulo CT	op_{41}^β	op_{42}^β	
	P_{22}	11 u.t. modulo CT	$op_{51}^{\beta-1}$	op_{52}^β	

Tableau III.3 – Ordre partiel d'exécution des opérations des produits attendus en sortie

5.3.1 Scénario 1

Nous considérons dans ce premier scénario que la pièce P_{12} sort du système de production avec un retard de 2 u.t.. En effet, on observe la date de sortie de la pièce P_{12} à la date $\beta \times CT + 10$. Le diagnostic a été lancé à cette date et toutes les observations antérieures ne présentent aucun symptôme de défaillance. Cela veut dire que la pièce P_{11} du cycle β est sortie du système à sa date prévisionnelle sans retard tandis que nous n'avons pas encore connaissance des dates de sortie des pièces P_{13} , P_{21} et P_{22} . En se basant sur ces observations, le solveur génère trois hypothèses (voir tableau III.4).

- **Hypothèse 1** : Il fait l'hypothèse que la défaillance serait due à la dernière opération op_{23}^β de la pièce P_{12} avec un retard propre de 2 u.t., $\theta(op_{23}^\beta) = 2$. Dans cette hypothèse, la dernière opération op_{33}^β de la pièce P_{13} serait également défaillante avec un retard propre de 2 u.t., $\theta(op_{33}^\beta) = 2$, puisqu'il s'agit de la même fonction de transformation de la machine M_2 . Ces deux opérations sont exécutées par la machine M_2 pour laquelle l'ordonnancement a défini une marge de 1 u.t. entre les deux. Cette marge consommerait une partie du retard induit par l'opération op_{23}^β . Dans ce cadre, le retard induit par induction de machine de l'opération op_{33}^β serait égal à 1, soit $I_m(op_{33}^\beta) = 1$. Dans l'hypothèse de la défaillance première de l'opération op_{23}^β , le diagnostic pronostique donc une sortie de la pièce P_{13} avec un retard global de 3 u.t. qui correspond à la somme du retard propre $\theta(op_{33}^\beta) = 2$ et du retard induit dû à l'induction de machine $I_m(op_{33}^\beta) = 1$.
- **Hypothèse 2** : Il fait l'hypothèse que la défaillance serait due à la deuxième opération op_{22}^β de la pièce P_{12} qui serait défaillante avec un retard propre de 2 u.t., $\theta(op_{22}^\beta) = 2$. Dans

cette hypothèse, la deuxième opération op_{32}^β de la pièce P_{13} serait également défaillante avec un retard propre de 2 u.t., $\theta(op_{32}^\beta) = 2$. Ces deux opérations sont exécutées par la machine M_1 qui est la machine menante. Cela entraînerait une induction de machine totale entre op_{22}^β et op_{32}^β . D'où le retard induit par induction de machine de l'opération op_{32}^β serait égal à 2 u.t., soit $I_m(op_{32}^\beta) = 2$. Dans l'hypothèse de la défaillance première de l'opération op_{22}^β , le diagnostic pronostique donc une sortie de la pièce P_{13} avec un retard global de 4 u.t. qui correspond à la somme du retard propre $\theta(op_{32}^\beta) = 2$ et du retard induit dû à l'induction de machine $I_m(op_{32}^\beta) = 2$.

- **Hypothèse 3 :** Il fait l'hypothèse que la défaillance serait due à la première opération op_{21}^β de la pièce P_{12} qui serait défaillante avec un retard propre de 2 u.t., $\theta(op_{21}^\beta) = 2$. Dans cette hypothèse, la première opération op_{31}^β de la pièce P_{13} serait également défaillante avec un retard propre de 2 u.t., $\theta(op_{31}^\beta) = 2$. Ces deux opérations sont exécutées par la machine M_3 avec une marge de 1 u.t. entre les deux. Cette marge consomme une partie du retard induit par l'opération op_{21}^β . Nous constatons que le retard induit par induction de machine de l'opération op_{31}^β est égal à 1, soit $I_m(op_{31}^\beta) = 1$. Dans l'hypothèse de la défaillance première de l'opération op_{21}^β , le diagnostic pronostique donc une sortie de la pièce P_{13} avec un retard global de 3 u.t. qui correspondrait à la somme du retard propre $\theta(op_{31}^\beta) = 2$ et du retard induit dû à l'induction de machine $I_m(op_{31}^\beta) = 1$.

Nous constatons à travers ce scénario que les observations d'un seul cycle sont insuffisantes pour identifier la source de la défaillance. D'où le besoin d'ajouter en ligne d'autres observations et de mettre à jour le résultat du diagnostic. En ajoutant la date de sortie de la pièce P_{13} avec un retard de 4 u.t., le diagnostiqueur raffine les 3 hypothèses du scénario 1 et donne une unique solution qui correspond à la solution 2. L'opération op_{22}^β est alors la source première de la défaillance ayant eu pour symptôme initiateur la sortie de la pièce P_{12} avec un retard de 2 u.t.. Le tableau III.5 donne les retards pronostiqués par le solveur pour les différentes pièces à sortir ensuite.

Le tableau III.6 illustre l'aspect incrémental du diagnostic du scénario 1. En effet, En ajoutant une seconde observation à la date $(\beta + 1) * CT + 4$, la résolution du modèle aboutit à une unique solution correspondant à la source première de défaillance. De plus, on constate que le temps de prise en compte de la seconde observation est plus faible que celui de la première. Le fait que la durée de résolution après une seconde observation diminue confirme que le diagnostic incrémental proposé ne reprend pas la résolution à chaque nouveau symptôme détecté.

Hypothèse 1						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{12}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	0	0	0	0	0
	op_{23}^{β}	2	0	0	0	2
P_{13}	op_{31}^{β}	0	0	0	0	0
	op_{32}^{β}	0	0	0	0	0
	op_{33}^{β}	2	1	1	0	3
Hypothèse 2						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{12}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	2	0	0	0	2
	op_{23}^{β}	0	2	0	2	2
P_{13}	op_{31}^{β}	0	0	0	0	0
	op_{32}^{β}	2	2	2	0	4
	op_{33}^{β}	0	4	1	4	4
Hypothèse 3						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{12}	op_{21}^{β}	2	0	0	0	2
	op_{22}^{β}	0	2	0	2	2
	op_{23}^{β}	0	2	0	2	2
P_{13}	op_{31}^{β}	2	1	1	0	3
	op_{32}^{β}	0	3	2	3	3
	op_{33}^{β}	0	3	1	3	3

Tableau III.4 – Résultats d'exécution du scénario 1

5.3.2 Scénario 2

Dans ce second scénario, nous considérons que la pièce P_{21} sort du système de production avec un retard de 1 u.t.. En effet, on observe la date de sortie de la pièce P_{21} à la date $(\beta + 1) \times CT + 2$. Ce scénario permet de tester le cas de chevauchement de cycle puisque la

Solution 2						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{12}	$op_{21}^{\beta+1}$	0	4	4	0	4
	$op_{22}^{\beta+1}$	2	6	6	4	8
	$op_{23}^{\beta+1}$	0	8	6	8	8
P_{13}	$op_{31}^{\beta+1}$	0	3	3	0	3
	$op_{32}^{\beta+1}$	2	8	8	3	10
	$op_{33}^{\beta+1}$	0	10	6	10	10

Tableau III.5 – Retards pronostiqués du scénario 1

Date réelle de sortie	Cycle	Observation (Pièce, Retard)	Nombre de solutions	Durée de résolution
$\beta * CT + 10$	β	$(P_{12}, 2 \text{ t.u.})$	3	0.023 sec
$(\beta + 1) * CT + 4$	$\beta + 1$	$(P_{13}, 4 \text{ t.u.})$	1	0.015 sec

Tableau III.6 – Aspect incrémental de résolution

dernière opération de cette pièce commence dans le cycle β et se termine dans le cycle $\beta + 1$. Les hypothèses générées par le solveur sont données dans le tableau III.7.

- **Hypothèse 1** : Il fait l'hypothèse que la défaillance serait due à la dernière opération op_{42}^{β} de la pièce P_{21} avec un retard propre de 1 u.t., $\theta(op_{42}^{\beta}) = 1$.
- **Hypothèse 2** : Il fait l'hypothèse que la défaillance serait due à la première opération op_{41}^{β} de la pièce P_{21} avec un retard propre de 1 u.t., $\theta(op_{41}^{\beta}) = 1$. Dans cette hypothèse, la première opération op_{51}^{β} de la pièce P_{22} serait également défaillante avec un retard propre de 1 u.t., $\theta(op_{51}^{\beta}) = 1$, puisqu'il s'agit de la même fonction de transformation de la machine menante M_1 . Cela entrainerait aussi une induction de machine entre op_{41}^{β} et op_{51}^{β} . Dans ce cadre, le retard induit par induction de machine de l'opération op_{51}^{β} serait égal à 1, soit $I_m(op_{51}^{\beta}) = 1$.

Hypothèse 1						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{11}	op_{11}^{β}	0	0	0	0	0
	op_{12}^{β}	0	0	0	0	0
	op_{13}^{β}	0	0	0	0	0
P_{12}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	0	0	0	0	0
	op_{23}^{β}	0	0	0	0	0
P_{13}	op_{31}^{β}	0	0	0	0	0
	op_{32}^{β}	0	0	0	0	0
	op_{33}^{β}	0	0	0	0	0
P_{21}	op_{41}^{β}	0	0	0	0	0
	op_{42}^{β}	1	0	0	0	1
P_{22}	op_{51}^{β}	0	0	0	0	0
	op_{52}^{β}	0	0	0	0	0
Hypothèse 2						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{11}	op_{11}^{β}	0	0	0	0	0
	op_{12}^{β}	0	0	0	0	0
	op_{13}^{β}	0	0	0	0	0
P_{12}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	0	0	0	0	0
	op_{23}^{β}	0	0	0	0	0
P_{13}	op_{31}^{β}	0	0	0	0	0
	op_{32}^{β}	0	0	0	0	0
	op_{33}^{β}	0	0	0	0	0
P_{21}	op_{41}^{β}	1	0	0	0	1
	op_{42}^{β}	0	0	0	1	1
P_{22}	op_{51}^{β}	1	0	1	0	2
	op_{52}^{β}	0	0	0	0	0

Tableau III.7 – Résultats d'exécution du scénario 2

Afin de raffiner l'ensemble des hypothèses possibles, le diagnostiqueur continue la résolution du modèle en ajoutant une autre observation. La première observation ajoutée au solveur correspond à la pièce P_{11} . Dans le cycle $(\beta + 1)$ on observe la sortie de la pièce P_{11} à sa date prévisionnelle de sortie $(\beta + 1) \times CT + 5$. A travers cette observation le solveur donne une unique solution qui correspond à l'hypothèse 1. L'opération op_{42}^{β} est alors la source première de la défaillance ayant eu pour symptôme initiateur la sortie de la pièce P_{21} avec un retard de 1 u.t.. Le tableau III.8 donne les retards pronostiqués par le solveur pour les différentes pièces à sortir ensuite.

Solution 1						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{11}	$op_{11}^{\beta+1}$	0	0	0	0	0
	$op_{12}^{\beta+1}$	0	0	0	0	0
	$op_{13}^{\beta+1}$	0	0	0	0	0
P_{12}	$op_{21}^{\beta+1}$	1	1	1	0	2
	$op_{22}^{\beta+1}$	0	2	0	2	2
	$op_{23}^{\beta+1}$	0	2	0	2	2
P_{13}	$op_{31}^{\beta+1}$	1	1	1	0	2
	$op_{32}^{\beta+1}$	0	2	2	2	2
	$op_{33}^{\beta+1}$	0	2	1	2	2
P_{21}	$op_{41}^{\beta+1}$	0	2	2	0	2
	$op_{42}^{\beta+1}$	1	4	4	2	5
P_{22}	$op_{51}^{\beta+1}$	1	3	3	0	4
	$op_{52}^{\beta+1}$	0	4	2	4	4

Tableau III.8 – Retards pronostiqués du scénario 2

5.4 Analyses qualitatives des résultats

L'exécution des différents scénarios mène à plusieurs conclusions sur l'efficacité du modèle mathématique proposé [Fakhfakh *et al.*, 2013] pour le diagnostic. D'une part, nous illustrons à travers le scénario 1 que si nous avons suffisamment d'observations, notre modèle de diagnostic est capable d'identifier la cause première d'une défaillance. Sinon il énumère les différentes hypothèses possibles. En outre, notre modèle a la capacité de prédire les dates de sorties des

pièces dans le futur à partir d'un symptôme de défaillance, ceci est illustré par la deuxième étape du scénario 1. La résolution du modèle dans le scénario 1 a pronostiqué le retard des pièces P_{13} et P_{22} avec une estimation de la valeur des retards respectifs. Dans la deuxième étape du scénario 1, nous ajoutons une nouvelle observation en cours d'exécution. Elle permet de raffiner en ligne l'ensemble des solutions générées par le diagnostic. La résolution est arrêtée à ce niveau puisque le diagnostic converge vers une solution unique ce qui correspond à notre critère d'arrêt. Le temps de résolution pour ces scénarios sera évalué dans la partie suivante.

6 Etude de performance du diagnostic

Bien que l'approche incrémentale ait une meilleure qualité de solution, nous avons besoin d'évaluer le temps de résolution du diagnostic incrémental et le comparer avec d'autres approches. D'autant plus qu'il existe plusieurs algorithmes de résolution au sein de l'approche incrémentale (stratégies de résolution du solveur).

Dans ce cadre, cette partie vise à évaluer l'approche de diagnostic incrémental proposée dans ce chapitre. Cette évaluation concerne la mesure du temps de résolution de différents scénarios, ensuite il s'agit de comparer ce temps avec celui de l'approche indirecte [Fakhfakh *et al.*, 2012] proposée dans le chapitre II ainsi qu'avec une autre approche de la littérature [Ly *et al.*, 2000].

Pour cette raison, quatre comparaisons seront effectuées :

- Comparer l'approche par arbre de résolution avec l'approche incrémentale sur un seul cycle (sur le même espace de recherche). Cette comparaison permet de comparer expérimentalement la qualité du diagnostic (la capacité à faire le bon diagnostic) ainsi que le temps de calcul.
- Comparer l'approche incrémentale sur un seul cycle avec l'approche incrémentale complète. Cette comparaison permet de prouver l'intérêt de l'aspect incrémental en termes de qualité de solution.
- Comparer les différentes stratégies de résolution proposées par le solveur. Cette comparaison permet de déterminer expérimentalement la stratégie de résolution la mieux adaptée à la problématique étudiée dans ce travail.
- Comparer l'approche incrémentale avec la littérature. Une fois qu'on a prouvé l'intérêt de l'aspect incrémental de diagnostic et déterminer la meilleur stratégie de résolution, nous procédons avec une comparaison avec une méthode de la littérature qui se place dans le même contexte que notre travail. Nous avons opté pour une comparaison avec les travaux

de Ly [Ly *et al.*, 2000] qui considèrent une approche de surveillance indirecte basée sur une analyse quantitative de la production.

Dans la suite, nous considérons les notations suivantes :

- *Approche 1* : l'approche de diagnostic indirect par arbre de résolution basée sur la technique de chaînage double. Cette approche ne fonctionne que sur le cycle de l'ordonnement dans lequel a été détecté le symptôme initiateur.
- *Approche 2* : l'approche de diagnostic incrémental basée sur la programmation par contraintes. Cette approche a la capacité de fonctionner sur plusieurs cycles. Le diagnostic est arrêté quand la résolution converge vers une seule solution.

Des tableaux comparatifs regroupant une variété de scénarios seront présentés dans la suite. Chaque ligne de ces tableaux correspond à un scénario de diagnostic. On donne pour chaque scénario l'éventuel retard de chaque pièce. Si une pièce sort du système de production sans retard, alors la valeur du retard correspondante est notée nulle. Sinon, s'il y a une déviation entre la date de sortie prévisionnelle et celle réelle, alors on note la différence entre ces deux dates comme une valeur de retard. Dans le cas où le diagnostic est lancé avant la date de sortie prévisionnelle d'une autre pièce, la valeur de retard de cette dernière reste indéterminée. Par exemple dans le deuxième scénario du tableau III.9, la pièce P_{11} est sortie à sa date prévisionnelle sans retard $R(P_{13}) = 0$, tandis que la pièce P_{12} accuse un retard de 1 u.t.. A la date considérée, nous n'avons pas encore observé les sorties des autres pièces. Pour chaque scénario, on note le temps de résolution en seconde ainsi que le nombre de solutions données par le solveur qui correspondent aux nombres d'hypothèses de défaillances possibles.

6.1 Evaluation du diagnostic d'un seul cycle

Nous rappelons que l'*Approche 1* proposée au chapitre II exécute le diagnostic sur un seul cycle de l'ordonnement cyclique. Afin de pouvoir comparer les deux approches, nous considérons alors dans cette partie l'*Approche 2* bridée à un seul cycle. Dans la suite on note *Mono-cycle-Approche 2*, l'*Approche 2* bridée à un seul cycle qui correspond à l'exécution du diagnostic sur un seul cycle d'ordonnement.

Scénario	Retard des pièces(u.t.)					Approche 1		Mono-cycle	Approche 2 ¹	
	P_{11}	P_{12}	P_{13}	P_{21}	P_{22}	Temps ²	Solutions ³	Temps	Solutions	
- 1 -	1	∅	∅	∅	∅	0.01 s	2	0.0312 s	2	
- 2 -	0	1	∅	∅	∅	0.019 s	3	0.0624 s	3	
- 3 -	0	0	1	∅	0	0.026 s	2	0.0156 s	2	
- 4 -	0	0	∅	∅	1	0.01 s	2	0.0468 s	1	
- 5 -	0	0	0	1	0	0.027 s	2	0.0468 s	2	
- 6 -	2	∅	∅	∅	∅	0.01 s	2	0.0321 s	2	
- 7 -	0	2	∅	∅	∅	0.01 s	3	0.0468 s	3	
- 8 -	0	0	2	∅	0	0.03 s	2	0.0642 s	2	
- 9 -	0	0	∅	∅	2	0.04 s	2	0.0312 s	2	
- 10 -	0	0	0	2	0	0.01 s	2	0.0468 s	2	
Moyenne de temps de résolution						0.02 s		0.0421 s		

¹ Approche 2 appliquée à un seul cycle.

² Temps de résolution de l'approche.

³ Nombre de solutions.

Tableau III.9 – Comparaison entre Approche 1 et Mono-cycle Approche 2

Dans la majorité des cas, les deux approches donnent le même nombre de solutions. Mais l'inconvénient dans ces cas c'est le fait que le diagnostic d'un seul cycle n'est pas assez précis pour identifier la source de défaillance. Comme le montre le tableau III.9, le nombre de solutions dans tous les cas est supérieur à 1. Pour cette raison, l'Approche 2, qui a la capacité de continuer le diagnostic dans les cycles suivants, est beaucoup plus intéressante même si elle est un peu plus lente en temps de calcul. Ceci est expliqué par le fait que l'Approche 2 est plus exhaustive en termes d'instanciation. En effet, l'Approche 2 instancie toutes les valeurs possibles puis elle fera l'élimination des fausses hypothèses plus tard avec la propagation des contraintes. Comme cette approche est incrémentale elle a besoin de plus de données pour raffiner l'ensemble de solutions. Puisque la première approche se limite à un seul cycle, elle arrête la construction de l'arbre de résolution dès qu'elle détecte une hypothèse qui peut être réfutée à la date courante (par exemple une opération est non défaillante et ne peut donc pas être une source de défaillance). Une évaluation du diagnostic complet sera détaillée dans la partie suivante.

6.2 Evaluation du diagnostic incrémental complet

Dans cette partie, nous considérons *Mono-cycle Approche 2* qui correspond à un test sur un cycle seulement (c.f. Tableau III.9 précédent) et l'*Approche 2* complète qui continue la résolution jusqu'à ce qu'elle converge. La colonne du temps de résolution du dernier cycle donne le temps de résolution à partir du dernier symptôme observé jusqu'à l'identification de la source de défaillance.

La comparaison entre *Mono-cycle Approche 2* et *Approche 2* complète dans le Tableau III.10 montre, d'une part, que le nombre de solutions converge dans tous les cas vers une unique solution. Ce qui montre l'intérêt de l'aspect incrémental de l'*Approche 2* complète qui a la capacité d'identifier la cause première de la défaillance.

D'autre part, l'évaluation de temps de résolution démontre que le fait d'aller vers le deuxième cycle de production ne multiplie pas par deux le temps de calcul. Ceci est important car cette valeur prouve que le diagnostic incrémental ne reprend pas la résolution dès le premier cycle de diagnostic, mais en poursuivant la résolution à chaque nouvelle observation ajoutée. En effet le temps de résolution donné correspond à celui de la totalité de l'application. Cependant ayant déjà commencé le diagnostic pour un cycle et après avoir eu les nouveaux symptômes, le temps de résolution nécessaire pour converger vers une solution est $1/3$ le temps total (c.à.d. de l'ordre de $2/100$ seconde). De plus, le temps de calcul du premier cycle est déjà inclus dans la durée du cycle de production, puisque le calcul de diagnostic est effectué en parallèle avec la production (diagnostic en ligne).

Retard des pièces (u.t.)		Mono-cycle				Approche 2		Approche 2 complète ¹			
Scénario	Cycle	P_{11}	P_{12}	P_{13}	P_{21}	P_{22}	Temps ²	Solutions ³	Temps ²	$Temps_{derniercycle}$ ⁴	Solutions ³
-1-	-1-	1	∅	∅	∅	∅	0.0312 s	2	0.0468 s	0.0156 s	1
	-2-	1	∅	∅	∅	∅					
-2-	-1-	0	1	∅	∅	∅	0.0624 s	3	0.0936 s	0.0312 s	1
	-2-	1	∅	∅	∅	∅					
-3-	-1-	0	0	1	∅	0	0.0156 s	2	0.0312 s	0.0156 s	1
	-2-	4	∅	∅	∅	∅					
-4-	-1-	0	0	∅	∅	3	0.0468 s	2	0.0780 s	0.0312 s	1
	-2-	3	∅	∅	∅	∅					
-5-	-1-	0	0	0	1	0	0.0468 s	2	0.0780 s	0.0312 s	1
	-2-	0	2	∅	∅	∅					
-6-	-1-	2	∅	∅	∅	∅	0.0312 s	2	0.0468 s	0.0156 s	1
	-2-	3	∅	∅	∅	∅					
-7-	-1-	0	2	∅	∅	∅	0.0468 s	3	0.0624 s	0.0156 s	1
	-2-	7	11	∅	∅	∅					
-8-	-1-	0	0	2	∅	0	0.0642 s	2	0.0780 s	0.0156 s	1
	-2-	8	∅	∅	∅	∅					
-9-	-1-	0	0	∅	∅	2	0.0312 s	2	0.0468 s	0.0156 s	1
	-2-	2	4	4	∅	∅					
-10-	-1-	0	0	0	2	0	0.0468 s	2	0.0624 s	0.0156 s	1
	-2-	1	∅	∅	∅	∅					
Moyenne de temps de résolution							0.0421 s		0.0624 s	0.0205 s	

¹ Approche de diagnostic incrémental

² Temps total de résolution

³ Nombre de solutions

⁴ Temps de résolution du dernier cycle de diagnostic

Tableau III.10 – Evaluation de l'approche incrémentale

En outre, la constante de temps des problèmes manufacturiers n'est pas de l'ordre de 10^{-2} seconde. Ce qui prouve que le temps de calcul de l'*Approche 2* complète est très raisonnable même s'il est légèrement supérieur à celui de l'*Approche 1*. De plus, le temps de résolution du dernier cycle de l'*Approche 2* complète montre que la plus grande partie du temps de résolution est incluse dans le premier cycle. Ainsi dans le cycle suivant, dès qu'on a un nouveau symptôme le résultat est presque immédiat.

Finalement, le temps de résolution reste négligeable par rapport à la durée du cycle d'ordonnement. Cette comparaison démontre la double efficacité de l'approche incrémentale, en termes de convergence et temps de résolution. Cette évaluation reste expérimentale faute de disposer de données de la complexité algorithmique des méthodes de résolution du solveur. Néanmoins nous pouvons confirmer que notre approche de diagnostic incrémental ne change pas la classe de complexité de la méthode de résolution adoptée. Ceci est expliqué par le fait que notre approche de diagnostic exécute la méthode de solveur n fois au pire des cas, où n correspond au nombre d'observations dans le système.

6.3 Stratégies de recherche

Dans cette section, nous proposons une comparaison expérimentale entre les différentes stratégies de résolution du solveur, puisque nous n'avons pas de connaissance concernant la complexité de ces stratégies.

Le principe de la programmation par contraintes se base sur deux phases pour résoudre un problème : la modélisation et la résolution. Dans la phase de modélisation, la programmation par contraintes permet de réduire l'espace de recherche par la réduction des domaines des variables et l'ajout dynamique des contraintes. Tandis que dans la phase de résolution, certaines stratégies de recherche permettent de trouver les solutions avec un parcours efficace de l'arbre de recherche tout en éliminant les nœuds non pertinents.

Après avoir modélisé le problème de diagnostic indirect et incrémental, la résolution du problème constitue une phase critique dans le diagnostic qui a une influence directe sur non seulement la qualité de solution mais aussi le temps de résolution. En effet, le temps nécessaire pour trouver une solution varie considérablement en fonction des stratégies de recherches et de l'espace de recherche.

Dans la suite, nous effectuons une comparaison entre les différentes stratégies de recherche données par le solveur adopté *IlogSolver* [Manual, 2010]. Ces stratégies de recherche sont basées sur la taille de domaines de définition de variables de décision.

- **FirstUnbound** : choisir la première variable donnée non traitée, c'est la stratégie par défaut utilisée par le solveur.
- **MinSize** : choisir la variable ayant le plus petit domaine de définition
- **MaxSize** : choisir la variable ayant le plus grand domaine de définition
- **MinMin** : choisir la variable ayant la plus petite borne inférieure
- **MaxMin** : choisir la variable ayant la plus grande borne inférieure
- **MinMax** : choisir la variable ayant la plus petite borne supérieure
- **MaxMax** : choisir la variable ayant la plus grande borne supérieure

Temps de résolution de la Stratégie de recherche									
Scénario	FirstUnbound	MinSize	MinMax	MinMin	MaxSize	MaxMin	MaxMax		
-1-	0,0312 s	0,0624 s	0,0312 s	0,0156 s	11,8561 s	≈ 1h	≈ 1h		
-2-	0,0312 s	0,0312 s	0,0312 s	0,078 s	14,5549 s	≈ 1h	≈ 1h		
-3-	0,0312 s	0,0624 s	0,0468 s	0,0468 s	0,0624 s	≈ 1h	≈ 1h		
-4-	0,0624 s	0,0156 s	0,1716 s	0,0468 s	80,4809 s	≈ 1h	≈ 1h		
-5-	0,0624 s	0,0312 s	0,0312 s	0,0624 s	0,4992 s	≈ 1h	≈ 1h		
-6-	0,0156 s	0,0312 s	0,0468 s	0,0159 s	491,169 s	≈ 1h	≈ 1h		
-7-	0,0312 s	0,078 s	0,0156 s	0,078 s	2,9952 s	≈ 1h	≈ 1h		
-8-	0,0156 s	0,0156 s	0,0312 s	0,078 s	0,0156 s	≈ 1h	≈ 1h		
-9-	0,0312 s	0,0156 s	0,0624 s	0,0312 s	2,0592 s	≈ 1h	≈ 1h		
-10-	0,0468 s	0,078 s	0,0156 s	0,0468 s	31,7306 s	3088,52	≈ 1h		
Moyenne de temps	0,03588 s	0,04212 s	0,04836 s	0,04995 s	63,54231 s	≈ 1h	≈ 1h		

Tableau III.11 – Stratégies de recherche

Les résultats donnés par le tableau III.11 montrent l'influence du choix de la stratégie de recherche sur le temps de résolution. En premier lieu, on constate à travers cette comparaison que les variables ayant la plus grande borne inférieure ou supérieure ont un temps de résolution très important qui dépasse une heure. En deuxième lieu, les variables ayant le plus grand domaine de définition ont un temps de résolution de l'ordre d'une minute. En effet, les valeurs d'un intervalle sont énumérés de la borne inférieure à la borne supérieure. Commencer par la plus grande valeur mène à instancier plus de valeurs, ce qui fait augmenter le temps de résolution. En effet, le diagnostic se place dans le cadre de défaillance unique, donc la valeur de retard propre est non nulle pour une seule variable.

6.4 Comparaison avec la littérature

Afin d'évaluer les performances de la méthode de diagnostic proposée dans ce travail, nous effectuons dans la suite une comparaison de notre méthode de diagnostic incrémental avec un travail de la littérature proposé par [Ly *et al.*, 2000] et situé dans le même contexte.

Comme indiqué dans la section 3.3.1, les travaux de [Ly *et al.*, 2000] présentent une approche de surveillance indirecte dans le contexte d'un ordonnancement classique avec pilotage réactif d'un SFPM. Cette approche [Ly *et al.*, 2000] se base sur une analyse quantitative de flux de production à la fin de la période de production. Cela veut dire que les résultats de diagnostic sont donnés qu'après la production de toutes les pièces par le SFPM.

Pour que la comparaison soit effectuée dans le même contexte, nous appliquons notre méthode de diagnostic sur le même système étudié par [Ly *et al.*, 2000]. Le système étudié, illustré par la figure III.4, est un SFPM composé de 5 machines $M_i/i \in \{1, \dots, 5\}$ pour produire 3 types de pièces (A, B et C). Les machines M_1 , M_3 et M_5 sont mono-opérations. Tandis que les machines M_2 et M_4 sont multi-opérations. Nous rappelons que $op_{\alpha i}^\beta$ présente la $i^{\text{ème}}$ opération de la gamme α qui commence dans le cycle β .

Le système est décrit comme suit :

- $A : op_{\alpha 1}^\beta(M_2, 15 \text{ u.t.}), \quad op_{\alpha 2}^\beta(M_1, 9 \text{ u.t.}), \quad op_{\alpha 3}^\beta(M_3, 27 \text{ u.t.})$
- $B : op_{\alpha 1}^\beta(M_4, 15 \text{ u.t.}), \quad op_{\alpha 2}^\beta(M_2, 15 \text{ u.t.}), \quad op_{\alpha 3}^\beta(M_1, 9 \text{ u.t.})$
- $C : op_{\alpha 1}^\beta(M_1, 9 \text{ u.t.}), \quad op_{\alpha 2}^\beta(M_5, 27 \text{ u.t.}), \quad op_{\alpha 3}^\beta(M_4, 15 \text{ u.t.})$

Notre travail se place dans le cadre du diagnostic en ligne qui considère des SFPM sous

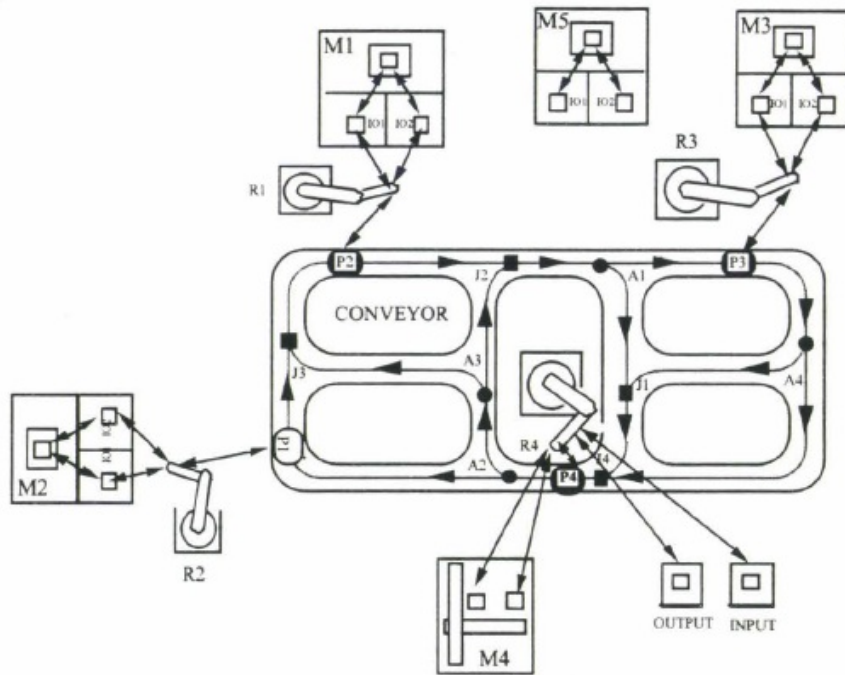


Figure III.4 – Système Flexible de Production Manufacturière [Ly *et al.*, 2000]

ordonnancement cyclique permettant d’avoir des points de références intermédiaires à la fin de chaque cycle de production. Comme première étape, nous proposons alors un ordonnancement cyclique au SFPM étudié par [Ly *et al.*, 2000].

La production cyclique consiste à réaliser une seule pièce pour chaque type par cycle, soit 3 pièces par cycle. L’ordonnancement cyclique fixe le temps de cycle optimal à 30 secondes. Les machines menantes, celles qui travaillent à 100% du temps, sont M_2 et M_4 . L’ordonnancement donne qu’il faut avoir 6 en-cours associés dans le système pour avoir 3 pièces finies en sortie du système par cycle. C’est à dire qu’il existe 2 en-cours par type de pièce.

L’ordonnancement cyclique que nous proposons au système de production étudié par [Ly *et al.*, 2000] est représenté par la figure III.5.

L’exécution des différents scénarios de diagnostic de défaut démontre l’efficacité de notre approche. L’expérimentation sur le système étudié confirme cette efficacité avec une performance de l’ordre de 7/100 secondes. Même si on a besoin d’ajouter les observations du cycle suivant afin de poursuivre le diagnostic incrémental et réduire le nombre de solutions possibles, on aura besoin d’attendre uniquement 30 secondes. Cette durée correspond au temps de cycle.

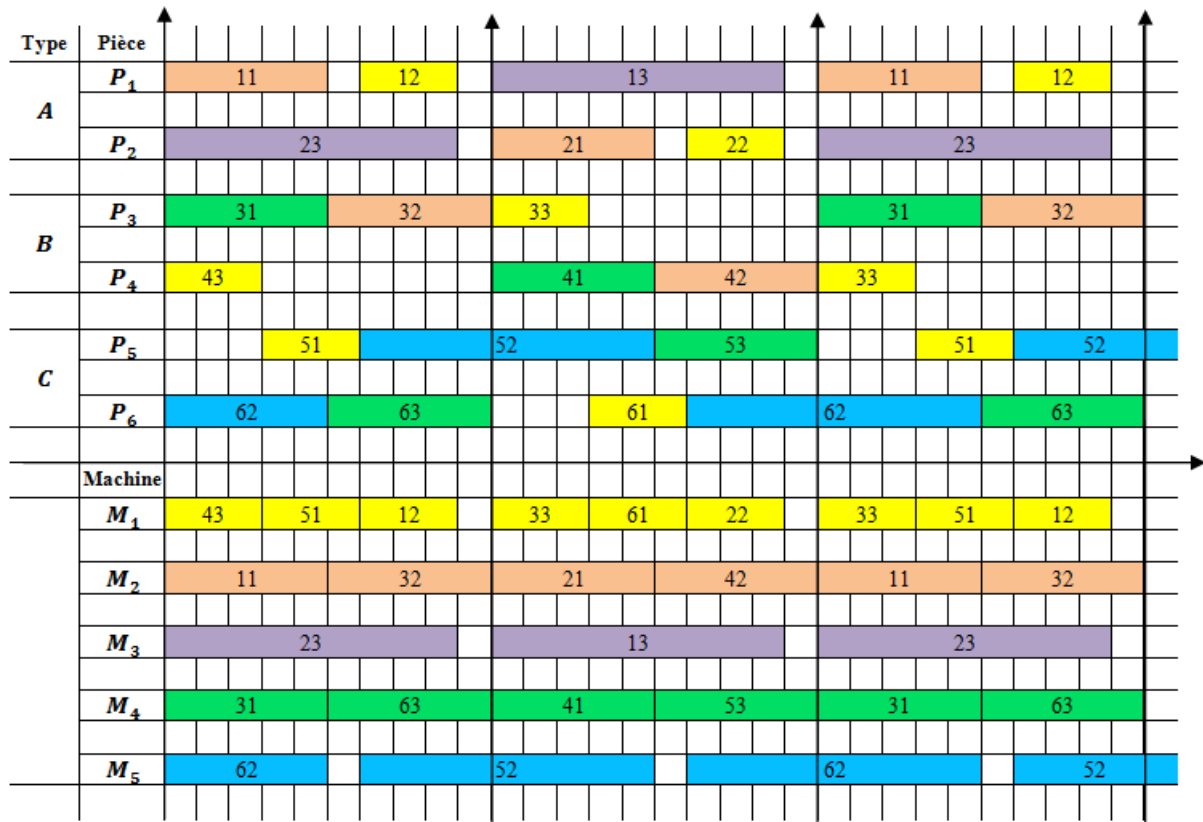


Figure III.5 – Ordonnancement cyclique d'un SFPM

Le tableau III.12 illustre les détails de résolution d'un scénario de défaillance. Nous considérons comme scénario de défaillance que la pièce P_6 sort du système de production avec un retard de 2 u.t.. En effet, on observe la date de sortie de la pièce P_6 à la date $\beta \times CT + 23$. Le solveur génère 2 hypothèses possibles dans une durée de 0.78 secondes. La seconde observation correspond à la sortie de la pièce P_3 à la date $(\beta + 1) \times CT + 10$, c'est à dire avec un retard de 1 u.t.. A travers cette observation le solveur donne une unique solution qui correspond à la source de défaillance. Le temps de prise en compte de la seconde observation est 0.0312 secondes.

Comparée aux résultats du diagnostic obtenu par l'approche proposée par [Ly et al., 2000], notre approche de diagnostic prouve une meilleure performance. En effet, les auteurs dans [Ly et al., 2000] considèrent un intervalle élémentaire de production est égale à 8370 secondes à la fin duquel 270 pièces de chaque type sont fabriquées. Après cette période de production, la méthode de diagnostic consiste à comparer le flux de production réel avec celui prévisionnel. Cette comparaison permet d'analyser le taux de dérive de chaque ressource ainsi que le taux

Date réelle de sortie	Cycle	Observation (Pièce, Retard)	Nombre de solution	Durée de résolution
$\beta \times CT + 32$	β	$(P_6, 2 \text{ t.u.})$	2	0.780sec
$(\beta + 1) \times CT + 10$	$\beta + 1$	$(P_3, 1 \text{ t.u.})$	1	0.0312 sec

Tableau III.12 – Aspect incrémental de résolution

de dérive de chaque gamme de production. Avec cette technique, les résultats du diagnostic de défauts ne peuvent pas être donnés avant environ 2 heures et 20 minutes.

7 Conclusion

Nous avons présenté au cours de ce chapitre le principe de notre solution de diagnostic indirect des SFPM sous une commande cyclique. Nous avons proposé une approche incrémentale de diagnostic indirect. Cette approche [Fakhfakh *et al.*, 2013] repose sur la construction d'un modèle mathématique basé sur la programmation par contraintes. La validation de l'approche conduit à plusieurs conclusions sur l'efficacité et la faisabilité du diagnostic en ligne, indirect et incrémental.

- **Diagnostic indirect** : Capacité du modèle à diagnostiquer un SFPM sous ordonnancement cyclique en se limitant à l'observation des sorties des pièces. Si nous avons suffisamment d'observations, notre modèle de diagnostic est capable d'identifier la cause première de la défaillance. Sinon, il suffit d'attendre les nouvelles observations données par les cycles suivants de l'ordonnancement.
- **Diagnostic incrémental** : Capacité du modèle à mettre à jour régulièrement le diagnostic à chaque nouvelle observation. Il ne s'agit pas de reprendre la résolution dès le premier cycle du diagnostic, mais de raffiner l'ensemble des solutions générées par le diagnostic du cycle précédent. Dans ce contexte, nous assurons une gestion dynamique des pièces créées afin de réduire l'utilisation de la mémoire et de pouvoir adapter à priori l'approche à la résolution de problèmes de grandes tailles.
- **Pronostic** : Capacité du modèle à prédire dans le futur les dates de sortie des pièces suite à une observation. Etant donné un symptôme initiateur, la fonction de pronostic intégrée dans le modèle de diagnostic permet de prévoir les défaillances qui auront lieu dans les cycles suivants en fonction des différentes hypothèses générées.

Avec l'aspect incrémental du diagnostic, la question qui peut se poser est : **Quand doit-on arrêter le diagnostic s'il ne converge pas vers une solution unique ?** Cette question a imposé d'effectuer une étude de diagnosticabilité de la commande cyclique qui sera détaillée dans le chapitre suivant.

Chapitre IV

Etude de la diagnosticabilité d'une commande cyclique

Sommaire

1	Etude de la diagnosticabilité	85
1.1	Principe et objectifs	85
1.2	Diagnosticabilité des SED	86
2	Proposition d'une nouvelle méthode de diagnosticabilité d'un ordonnancement cyclique	91
2.1	Définition de la diagnosticabilité d'un SFPM	91
2.2	Vérification de la détectabilité	92
3	Application de la diagnosticabilité sur des cas d'étude	94
3.1	Etude de la détectabilité d'un SFPM à plusieurs types de pièce	94
3.2	Etude de détectabilité pour un SFPM à plusieurs en-cours par produit	96
4	Vers un ordonnancement diagnosticable	99
4.1	Solutions de diagnosticabilité	99
4.2	Propriétés d'un ordonnancement diagnosticable	100
5	Conclusion	101

Afin de garantir la qualité du diagnostic et de prévoir la capacité d'une approche à diagnostiquer un système, il est intéressant d'étudier à l'avance la diagnosticabilité. Cette étude permet de répondre à l'interrogation suivante : Compte-tenu de l'observabilité dont on dispose sur le système et d'un ordonnancement cyclique donné, est-il possible d'identifier en un nombre fini de cycles la cause première de toute défaillance affectant le système ?

Afin de répondre à cette question, nous allons étudier la notion de diagnosticabilité des SFPM. Dans un premier lieu, nous allons proposer une synthèse des techniques proposées dans la littérature pour tester hors ligne la diagnosticabilité de SED. L'objectif de cette synthèse est d'identifier une technique qui soit adaptable à notre problématique. En second lieu, nous allons proposer une méthode d'étude de la diagnosticabilité qui constituera la base de notre contribution.

1 Etude de la diagnosticabilité

Cette partie permet d'introduire le principe de la diagnosticabilité des SED en général. Ensuite, il s'agit de représenter l'état de l'art relatif aux différentes techniques de diagnosticabilité.

1.1 Principe et objectifs

La diagnosticabilité permet de savoir si les observabilités prévus pour le système sont suffisantes pour localiser et identifier au bout d'un temps fini les causes premières de défaillance. Cette étude aide à vérifier si toute défaillance est diagnosticable dans un délai borné à partir d'un modèle du procédé et d'un ensemble d'événements observables. Une défaillance est diagnosticable si le diagnostiqueur est capable de l'identifier sans ambiguïté par rapport aux autres défaillances. Nous nous limitons dans ce qui suit aux travaux qui ont traité le problème de diagnosticabilité dans les systèmes à événements discrets.

La définition de la diagnosticabilité pour les SED a été initialement formalisée dans les travaux de [Lin, 1994]. Plusieurs extensions de cette définition ont été ensuite proposées, dans différents contextes [Sampath *et al.*, 1995], [Debouk *et al.*, 2000], [Jiang *et al.*, 2001], [Tripakis, 2002], [Yoo et Lafortune, 2002] et [Grastien et Anbulagan, 2010].

1.2 Diagnosticabilité des SED

Selon la dynamique du système (continue, discret, hybride), la structure de diagnostic (centralisée, décentralisée, distribuée), le modèle de défaillance (à base d'événements, à base d'états), et l'outil de représentation (automates à états, automates temporisées, réseaux de Petri) plusieurs formes de diagnosticabilité peuvent être définies.

1.2.1 Classification selon la modélisation des défaillances

La notion de diagnosticabilité s'applique autant pour la modélisation des défaillances à bases d'événements que celle à base d'états.

(a) Diagnosticabilité à base d'événements

Les SED peuvent être modélisés par des modèles à base d'événements. Il s'agit de systèmes dont le comportement est vu comme des transitions possibles entre différents états suite à l'occurrence d'événements. La notion de diagnosticabilité à base d'événements a été introduite et définie par [Sampath *et al.*, 1995] de la manière suivante :

«Chaque défaillance de l'ensemble des défaillances doit avoir une signature distincte et observable pour inférer l'occurrence de cette défaillance et déterminer son type».

Après la construction du diagnostiqueur et pour assurer la diagnosticabilité, il suffit que ce diagnostiqueur satisfasse les deux conditions suivantes :

- Il existe au moins un état du diagnostiqueur pour lequel le diagnostiqueur décide avec certitude l'occurrence d'un défaut appartenant à une partition des défauts.
- Il ne doit pas y avoir de cycles dits «indéterminés» pour lesquels le diagnostiqueur est incapable de décider avec certitude l'occurrence d'un défaut appartenant à une partition des défauts.

D'après [Pencole, 2004] une défaillance est le résultat de motifs d'événements. Dans ce cadre la diagnosticabilité à base d'événements peut être définie informellement de la manière suivante : *« L'occurrence d'un événement de panne F est diagnosticable s'il est toujours possible de diagnostiquer l'occurrence de F sans ambiguïté après un nombre fini d'observations qui suivent cette occurrence».* Autrement dit, si F est diagnosticable :

- Il est toujours possible de conclure sur la présence certaine de F ;
- Cette conclusion est établie en attendant un nombre fini d'observations.

(b) Diagnosticabilité à base d'états

La notion de diagnosticabilité pour les modèles à base d'état a été introduite et définie dans [Lin, 1994] de la manière suivante : Si un état du diagnostiqueur a été atteint suite à l'occurrence d'une séquence d'évènements comprenant une défaillance, alors toute séquence d'évènements ayant un comportement observable équivalent au premier, doit faire évoluer le modèle vers un autre état du diagnostiqueur. D'après [Sayed-Mouchaweh *et al.*, 2005] cette approche nécessite l'identification de l'état courant après chaque réception d'évènements produit par le procédé.

1.2.2 Classification selon l'outil de représentation pour le diagnostic

La notion de diagnosticabilité est adaptée aux différents outils de modélisation utilisés pour le diagnostic des SED.

(a) Automates à états

Les automates à états sont un outil formel de modélisation basé sur la théorie des langages et adapté pour la description du comportement événementiel d'un procédé. Il existe des travaux sur la diagnosticabilité des SED qui utilisent les automates finis simples comme modèles de représentation [Sampath *et al.*, 1995]. Dans le cadre des automates à états finis, on peut s'intéresser au délai maximal nécessaire pour diagnostiquer toute défaillance dans le système. Si cela nécessite l'occurrence d'au plus K événements observables après l'occurrence, on dit que le système est K -diagnosticable [Tripakis, 2002].

Afin de tenir compte de l'information temporelle sur la réception d'évènements (délais entre l'observation de deux événements ou leurs instants d'occurrence), ces automates sont étendus vers des automates temporisés.

(b) Automates temporisées

Dans [Tripakis, 2002], le diagnostic et la diagnosticabilité sont analysés dans le cadre des automates temporisés en suivant l'approche des « twin plants » proposée par [Jiang *et al.*, 2001]. Plusieurs autres travaux ont étudié la diagnosticabilité en utilisant les automates temporisées : [Bouyer *et al.*, 2005] [Saddem, 2012].

L'auteur dans [Tripakis, 2002] propose une extension de la notion de diagnosticabilité, in-

tégrant le temps, dite la Δ - diagnosticabilité. En effet, un automate temporisé est dit Δ -diagnosticable, s'il est possible de détecter tout défaut non observable, au bout d'un délai de Δ u.t. après son occurrence.

(c) Réseaux de Petri

La diagnosticabilité des réseaux de Petri en général a été largement étudiée par [Genc et Lafortune, 2003] [Wen et Jeng, 2004]. Dans [Haar *et al.*, 2003], la diagnosticabilité des SED a été étudiée en utilisant la technique de dépliage des réseaux de Petri (Petri Net unfoldings). Les travaux de [Liu *et al.*, 2012], [Liu *et al.*, 2013] et [Liu, 2014] ont proposée une extension des RdP temporels pour laquelle à chaque transition est associé un événement qui peut être observable ou non. La diagnosticabilité des SED temporisés a été vérifiée en utilisant le réseau de Petri T-temporel et labellisé. L'analyse de la diagnosticabilité est effectuée sur la base de deux modèles nommés respectivement FM-graph et FM-set tree qui sont construits à la volée. Les algorithmes à la volée proposés dans les travaux de [Liu, 2014] ont l'avantage d'obtenir un résultat sans avoir nécessairement à explorer tout l'espace d'états.

1.2.3 Classification selon la structure de la méthode de diagnostic

La notion de diagnosticabilité s'applique aux différentes structures de diagnostic des SED.

(a) Approche centralisée

Dans l'approche centralisée, le système étudié est décrit par un seul modèle globale qui intègre l'intégralité de son comportement. La structure centralisée d'un diagnostiqueur consiste à représenter un seul modèle du procédé avec un seul module de diagnostic. Cependant, pour des raisons de complexité, il devient très coûteux voire impossible de représenter dans un modèle unique le fonctionnement de l'ensemble du système.

Dans le contexte de diagnosticabilité centralisée, on trouve dans la littérature des approches à base de diagnostiqueur [Sampath *et al.*, 1995] ou à base de vérificateur ([Yoo et Lafortune, 2002] et [Jiang *et al.*, 2001]). Le vérificateur est un outil basé sur la construction d'un pseudo-diagnostiqueur qui a été conçu pour la vérification de la diagnosticabilité d'un système discret modélisé par un automate à états finis. L'un des inconvénients majeur de la proposition centra-

lisée de Sampath est le risque d'explosion combinatoire engendré pour des systèmes complexes. Ces travaux initiaux ont donc fait l'objet de nombreuses extensions. Ces extensions sont liées soit à une réduction du problème initial en sous-problèmes plus simples à résoudre, soit à un changement de formalisme.

(b) Approche décentralisée/ distribuée

Les systèmes complexes peuvent être considérés comme un ensemble de sous-systèmes qui coopèrent à travers des modes de communications appropriés. Le principe de la diagnosticabilité distribuée consiste à vérifier la diagnosticabilité globale d'un système en se fondant des modèles locaux des sous-systèmes qui le constituent et de la manière de communication entre ses sous-systèmes sans être obligé de calculer le modèle global du système. Il s'agit donc de concevoir des algorithmes qui permettent de faire émerger une décision globale sur la diagnosticabilité d'un système à partir de décisions locales sur la diagnosticabilité de ses différentes parties.

Au niveau de la réduction de la complexité par décomposition en sous-problèmes, les différentes extensions concernent la diagnosticabilité décentralisée [Debouk *et al.*, 2000] ou la diagnosticabilité modulaire [Debouk *et al.*, 2002], [Contant *et al.*, 2006]. Le travail décrit dans [Contant *et al.*, 2006] peut être vu comme une généralisation de l'algorithme initial de vérification de la diagnosticabilité proposé dans [Sampath *et al.*, 1995] au cas d'une architecture décentralisée modulaire. [Sengupta, 1998] propose une architecture de diagnostic pour les systèmes physiquement distribués qui se base sur l'échange direct d'information entre sites sans coordonnateur, ni fonction de fusion. Enfin, les travaux décrits dans [Pencole, 2004] et [Schumann et Pencolé, 2007] s'intéressent également au traitement de la diagnosticabilité de SED dans le cas distribué en adaptant l'algorithme centralisé proposé dans [Jiang *et al.*, 2001] au contexte distribué.

(c) Co-diagnosticabilité

La notion de co-diagnosticabilité doit permettre d'assurer le fait que toute défaillance doit être diagnostiquée dans un délai borné par au moins un diagnostiqueur local en utilisant ses propres observations.

Les travaux relatifs à la co-diagnosticabilité [Wang *et al.*, 2007] correspondent à une amélioration des travaux initiaux de [Debouk *et al.*, 2000] sur le diagnostic décentralisé. Ils ont pour

objectif de simplifier l'architecture proposée notamment en remplaçant le coordonnateur par un centre de fusion plus simple, ne nécessitant pas de capacité de mémorisation. Cela nécessite de revoir le fonctionnement des différents sites et notamment implique que les sites prennent des décisions locales. Ces décisions locales sont communiquées au centre de fusion qui agrège l'information pour prendre une décision globale. L'originalité de ce travail réside dans l'introduction d'une nouvelle forme de diagnostic : l'absence de faute. En effet, les sites ne se contentent pas de diagnostiquer s'il y a une faute. Ils peuvent également dire que jusqu'à une certaine évolution du système, il y a absence de faute dans le système.

D'autres travaux [Qiu et Kumar, 2004] ont développé la notion de co-diagnosticabilité pour garantir l'absence de défaut. On parle alors de co-diagnosticabilité forte (Strong diagnosability). La diagnosticabilité collaborative (Joint-Diagnosability) s'applique essentiellement aux structures distribuées. C'est une extension de la co-diagnosticabilité puisqu'elle se base sur les informations locales de chaque diagnostiqueur mais également sur les informations des diagnostiqueurs voisins. En effet, la structure distribuée permet d'avoir une communication entre les différents diagnostiqueurs. Cette communication représente une information supplémentaire au niveau de chaque diagnostiqueur qui est nécessaire pour la prise de décision finale. Cette communication est définie par un protocole de communication.

1.2.4 Autres approches formelles

La propriété de diagnosticabilité dans les SED peut être vue comme une propriété formelle que l'on veut vérifier qu'un modèle formel donné doit satisfaire. C'est pourquoi la vérification de la diagnosticabilité a attiré des chercheurs qui s'intéressent à ce type de problèmes, notamment ceux qui travaillent sur le model-checking. Les auteurs dans [Cimatti et Cavada., 2003] traitent la vérification formelle des systèmes de diagnostic. Le problème de diagnosticabilité est traduit en la recherche de paires de scénarios ayant les mêmes traces d'observables, mais conduisant à des situations différentes. Les conditions de diagnosticabilité sont formellement exprimées en logique temporelle linéaire (LTL). La vérification de diagnosticabilité est réalisée en résolvant un problème de model-checking.

D'autres travaux s'intéressent à la vérification de la diagnosticabilité des SED en basant sur des langages algébriques. Les travaux de [Console et al., 2000] proposent d'utiliser le langage algébrique PEPA (Performance Evaluation Process Algebra) pour la modélisation de systèmes physiques, la caractérisation de leur diagnostic et l'analyse de leur propriété de diagnosticabilité.

Dans le même contexte, [Rintanen et Grastien, 2007] cherche à vérifier des propriétés formelles en les modélisant dans un cadre logique à l'aide de formules SAT et en utilisant ensuite des solveurs adaptés pour la vérification. Dans ces travaux les systèmes à événements discrets sont modélisés par des variables d'états à valeurs booléennes. La technique de diagnostic proposée dans ce travail se base sur la notion de cohérence [Grastien et Anbulagan, 2010]. En effet, le système de diagnostic renvoie une hypothèse de diagnostic si celle-ci est considérée comme cohérente avec le modèle et les observations. Ensuite la vérification de la diagnosticabilité dans des ces systèmes est assurée par la satisfiabilité propositionnelle SAT.

2 Proposition d'une nouvelle méthode de diagnosticabilité d'un ordonnancement cyclique

Dans cette partie, nous définissons la diagnosticabilité dans le cadre d'étude de ce travail. Ensuite, nous détaillons le principe de l'étude de la détectabilité que nous proposons pour un SFPM sous une commande cyclique.

2.1 Définition de la diagnosticabilité d'un SFPM

L'étude de diagnosticabilité intervient principalement dans la phase qui précède le diagnostic, soit la phase de la vérification de la qualité du système ou la phase de conception. La diagnosticabilité est une capacité établie hors ligne et qui mesure l'aptitude d'une approche à diagnostiquer toute défaillance dans un délai fini. Il s'agit de s'assurer qu'en fonctionnement, le diagnostiqueur sera toujours capable de détecter une telle défaillance lorsqu'elle apparaît et à l'identifier sans ambiguïté par rapport aux autres défaillances. Le premier cas revient donc à s'assurer qu'une défaillance engendre toujours un comportement anormal observable, et le second revient à s'assurer que plusieurs défaillances distinctes n'engendrent pas un même comportement observable.

Un SFPM est diagnosticable si et seulement si toutes ses opérations sont diagnosticables. Une opération est diagnosticable s'il est toujours possible de diagnostiquer la défaillance de cette opération dans un temps fini. Ce qui revient à vérifier les deux conditions suivantes :

- La défaillance d'une opération provoque une perturbation sur le flux de production du SFPM. C'est à dire, la défaillance d'une opération induit certainement l'observation d'un

comportement défaillant dans un temps fini. Il s'agit de vérifier si la défaillance d'une opération est détectable.

- La défaillance d'une opération engendre un comportement défaillant différent des autres défaillances et qu'il est ainsi possible de les distinguer les uns des autres. C'est à dire, la défaillance de cette opération est discriminable de toutes autres défaillances. Il s'agit de vérifier si la défaillance d'une opération est identifiable.

2.2 Vérification de la détectabilité

L'étude de la diagnosticabilité d'un SFPM revient à étudier la diagnosticabilité de toutes ses opérations. Afin de vérifier la diagnosticabilité d'un SFPM on se limite à la vérification de la détectabilité de défaillances de toutes les opérations qui composent le système de production à diagnostiquer. La détectabilité d'un défaut signifie qu'il est possible de le détecter dans un temps fini, c'est à dire qu'il est possible d'affirmer que le comportement observable du système correspond à un comportement de défaut.

Dans cette partie, il s'agit de vérifier si la défaillance d'une opération engendre au moins une pièce sortie en retard par rapport à sa date prévisionnelle après un nombre fini de cycles. Autrement dit, la défaillance d'une opération est dite non détectable si son retard est totalement absorbé par les marges qui existent entre les opérations du système.

Le principe de vérification de la détectabilité d'une opération se base sur la démarche suivante : on suppose que le retard propre de cette opération soit strictement supérieur à zéro. Il est évident que tant qu'une défaillance à peu d'impact sur la durée de l'opération elle sera difficilement détectable. On s'intéresse donc à des défaillances dont l'ordre de grandeur est comparable à celui de la durée normale d'une opération. Aussi on suppose un retard propre minimum d'au moins 1 u.t.. Ensuite, il suffit de lancer le pronostic et vérifier l'impact temporel de cette défaillance sur les dates de sorties des pièces. Le pronostic est arrêté dès qu'on détecte un comportement défaillant : la première pièce sortant en retard. Dans ce cas, la défaillance de cette opération est considérée détectable. Si aucun comportement défaillant n'est observé après une durée maximale de pronostic, le pronostic sera arrêté. Ainsi, la défaillance de cette opération est considérée non détectable. Cette durée maximale de pronostic est définie a priori par un nombre maximum de cycles à créer pour observer le comportement du système. L'algorithme de vérification de la détectabilité d'une opération est décrit par l'Algorithme 5. D'après l'hypothèse A3, seules les dates de sortie des pièces sont observables. Par conséquent, la date réelle de fin de la dernière opération est connue du fait de l'observation du système.

Nous considérons par la suite comme observable la dernière opération de chaque pièce. Il est à noter que toute opération observable est directement détectable, donc cet algorithme ne traite que les opérations non observables.

Début

1. Initialiser le nombre de cycles créés : $\text{nbrCycle} = 0$
 2. Initialiser le nombre de défaillances détectées : $\text{nbrDéfaillance} = 0$
 3. Lire le nombre maximal de cycles à créer : maxCycle
 4. **Tant que** ($\text{nbrCycle} < \text{maxCycle}$ **ET** $\text{nbrDéfaillance} == 0$) **faire**
 - 4.1. $\text{nbrCycle} ++$
 - 4.2. Créer un nouveau cycle
 - Si** ($\text{nbrCycle} == 1$) **Alors**
 - | Ajouter la contrainte : Retard propre de l'opération = 1 u.t.
 - Sinon**
 - Faire le chaînage avec le cycle précédent
 - Ajouter les contraintes d'induction entre les cycles
 - Fin Si**
 - 4.3. Lancer le pronostic
 - Fait**
 5. **Si** ($\text{nbrDéfaillance} > 0$) **Alors**
 - | Retourner : Opération Détectable
 - Sinon**
 - | Retourner : Opération Non Détectable
 - Fin Si**
- Fin.**

Algorithme 5 – Vérification de la détectabilité d'une opération

Si après un nombre maximum de cycles créés, l'application ne détecte aucune défaillance alors la défaillance de l'opération est considérée non détectable et par suite non diagnosticable. Dans le cas contraire, si après un certain nombre de cycles, le système détecte au moins une défaillance dans les pièces en sortie alors la défaillance de l'opération est considérée détectable.

- Le nombre de cycles créés détermine le temps maximal pour qu'une défaillance soit détectable.
- Le nombre de pièces détectées en retard détermine le degré de perturbation de la production suite à la défaillance de cette opération.

En vérifiant la détectabilité de toutes les opérations d'un SFPM pour un cycle donné, nous pouvons vérifier la détectabilité d'un SFPM. Si le système à diagnostiquer contient au moins

une opération non détectable alors le système de production à diagnostiquer est considéré non détectable et par conséquent non diagnosticable.

3 Application de la diagnosticabilité sur des cas d'étude

Nous appliquons dans cette partie l'ensemble du cadre théorique de la diagnosticabilité présenté dans ce chapitre sur les cas d'étude introduits au chapitre (I), soit un SFPM à plusieurs types de pièce et un SFPM à plusieurs en-cours par produit.

3.1 Etude de la détectabilité d'un SFPM à plusieurs types de pièce

Cette partie concerne l'étude de la détectabilité du système de production présenté dans la figure I.5. Nous rappelons que le système est composé de trois machines (M_1, M_2 et M_3) utilisées pour réaliser deux types de pièces. La production cyclique dans ce système consiste à produire dans chaque cycle 3 pièces de type P_1 notées $\{P_{11}, P_{12}, P_{13}\}$ et 2 pièces de type P_2 notées $\{P_{21}, P_{22}\}$.

Le tableau IV.1 présente les résultats obtenus en vérifiant la détectabilité de l'opération op_{21}^β de la machine M_3 . Nous constatons à travers cette exécution que la défaillance de l'opération op_{21}^β entraîne la sortie en retard de 4 pièces $\{P_{12}, P_{13}, P_{21}$ et $P_{22}\}$ dans le cycle courant. Elle est alors considérée détectable en 1 cycle. La détectabilité de cette opération est assurée grâce à l'induction de gamme sans marge pour la pièce P_{12} .

Nous étudions dans la suite la détectabilité de l'opération op_{51}^β de la pièce P_{22} qui est la dernière opération de la machine M_1 . Le tableau IV.2 présente les détails d'évaluation de la détectabilité de cette opération.

Le pronostic d'un cycle montre que la défaillance de l'opération op_{51}^β n'engendre aucune pièce sortie en retard. Il faut donc poursuivre le pronostic dans le cycle suivant. Nous rappelons que les trois premières opérations de M_1 et les deux dernières sont de type différent. Les résultats du pronostic donnent cinq pièces défaillantes dans le cycle suivant. Nous constatons alors que l'opération op_{51}^β est détectable dans deux cycles.

Le tableau IV.3 résume le résultat de l'étude de la détectabilité de toutes les opérations du premier système de production étudié. Seule l'opération op_{51}^β de la machine M_1 est détectable en 2 cycles, les autres opérations sont détectables en 1 cycle.

Cycle n° : 1, Pièce(s) défaillante(s) : 4						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{11}	op_{21}^{β}	1	1	1	0	2
	op_{22}^{β}	0	0	0	0	0
	op_{23}^{β}	1	0	0	0	0
P_{12}	op_{21}^{β}	1	0	0	0	1
	op_{22}^{β}	0	1	0	1	1
	op_{23}^{β}	0	1	0	1	1
P_{13}	op_{31}^{β}	1	0	0	0	1
	op_{32}^{β}	0	1	0	1	1
	op_{33}^{β}	0	1	0	1	1
P_{21}	op_{41}^{β}	0	1	1	0	1
	op_{42}^{β}	1	3	3	1	4
P_{22}	op_{51}^{β}	0	3	1	3	3
	op_{52}^{β}	1	2	2	0	3

 Tableau IV.1 – Détectabilité de l'opération op_{21}^{β} de la machine M_3

Machines	Opérations	Détectabilité	Explications
M_1	$op_{12}^{\beta}, op_{22}^{\beta}, op_{32}^{\beta}, op_{41}^{\beta}$	1 cycle	Induction gamme sans marge
	op_{51}^{β}	2 cycle	Induction gamme avec marge
M_2	$op_{13}^{\beta}, op_{23}^{\beta}, op_{33}^{\beta}$	1 cycle	Opérations observables
M_2	$op_{21}^{\beta}, op_{31}^{\beta}$	1 cycle	Induction gamme sans marge
	$op_{52}^{\beta}, op_{42}^{\beta}$	1 cycle	Opérations observables
	op_{11}^{β}	1 cycle	Induction gamme sans marge

Tableau IV.3 – Etude de la détectabilité d'un SPFM à plusieurs types de pièce

Le système de production étudié est alors considéré détectable au pire des cas en 2 cycles.

Cycle n° : 1, Pièce(s) défaillante(s) : 4						
Pièce	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
P_{11}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	0	1	1	0	1
	op_{23}^{β}	0	1	0	1	1
P_{12}	op_{21}^{β}	0	0	0	0	0
	op_{22}^{β}	0	1	1	0	1
	op_{23}^{β}	0	1	0	1	1
P_{13}	op_{31}^{β}	0	0	0	0	0
	op_{32}^{β}	0	1	1	0	1
	op_{33}^{β}	0	1	1	0	1
P_{21}	op_{41}^{β}	1	1	1	0	2
	op_{42}^{β}	0	1	0	1	1
P_{22}	op_{51}^{β}	1	2	2	0	3
	op_{52}^{β}	0	0	0	0	0

Tableau IV.2 – Détectabilité de l'opération op_{51}^{β} de la machine M_1

3.2 Etude de détectabilité pour un SFPM à plusieurs en-cours par produit

Cette partie concerne l'étude de la détectabilité du système de production présenté dans la figure I.6. Le système est composé de 4 machines utilisées pour réaliser un type de pièce P. L'ordonnancement donne qu'il faut avoir 4 en-cours associés dans le système pour avoir une pièce finie par cycle.

Le tableau IV.4 présente les résultats obtenus en vérifiant la détectabilité de l'opération op_{11}^{β} de la machine M_1 . Le pronostic d'un cycle montre que la défaillance de l'opération op_{11}^{β} n'engendre aucune pièce sortie en retard. L'étude de la détectabilité se poursuit alors dans le cycle suivant, dans lequel on détecte une pièce défaillante. Nous rappelons que les deux opérations op_{11}^{β} et op_{14}^{β} de la machine M_1 sont de type différent.

Cycle n° : 1, Pièce(s) défaillante(s) : \emptyset						
Machine	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
M_1	op_{11}^{β}	1	0	0	0	1
	op_{14}^{β}	0	1	1	0	1
M_2	op_{12}^{β}	0	0	0	0	0
M_3	op_{31}^{β}	0	0	0	0	0
M_4	op_{15}^{β}	0	0	0	0	0
Cycle n° : 2, Pièce(s) défaillante(s) : 1						
Machine	$op_{\alpha i}^{\beta+1}$	$\theta(op_{\alpha i}^{\beta+1})$	$I(op_{\alpha i}^{\beta+1})$	$I_m(op_{\alpha i}^{\beta+1})$	$I_g(op_{\alpha i}^{\beta+1})$	$R(op_{\alpha i}^{\beta+1})$
M_1	$op_{11}^{\beta+1}$	1	1	1	0	2
	$op_{14}^{\beta+1}$	0	2	2	0	2
M_2	$op_{12}^{\beta+1}$	0	0	0	0	0
M_3	$op_{31}^{\beta+1}$	0	0	0	0	0
M_4	$op_{15}^{\beta+1}$	0	1	0	1	1

Tableau IV.4 – Détectabilité de l'opération op_{11}^{β} de la machine M_1

Nous étudions dans la suite la détectabilité de l'opération op_{12}^{β} de la machine M_2 . En analysant les résultats de la détectabilité de cette opération dans le tableau IV.5, nous constatons que la défaillance de l'opération op_{12}^{β} n'a pas d'effet sur le flux de production du système étudié puisqu'on ne détecte pas de retard dans la date de sortie de la pièce finie.

L'étude de la détectabilité se poursuit dans les cycles suivants jusqu'au dernier cycle d'étude (fixé par l'utilisateur à 10) et le résultat demeure sans détection de retard (voir tableau IV.6). Par conséquent, l'opération op_{12}^{β} est considérée non détectable. Ceci est expliqué par l'existence d'une marge de machine de 2 u.t. en aval de l'opération op_{12}^{β} et d'une marge de gamme de 2 u.t. en aval de la même opération.

Le tableau IV.7 résume les résultats d'étude de la détectabilité de toutes les opérations du système de production étudié (Figure I.6). Cette étude montre que les opérations op_{12}^{β} et op_{13}^{β} ne sont pas détectables pour la même cause qui est l'existence des marges supérieures à 1 u.t.

Cycle n° : 1, Pièce(s) défaillante(s) : \emptyset						
Machine	$op_{\alpha i}^{\beta}$	$\theta(op_{\alpha i}^{\beta})$	$I(op_{\alpha i}^{\beta})$	$I_m(op_{\alpha i}^{\beta})$	$I_g(op_{\alpha i}^{\beta})$	$R(op_{\alpha i}^{\beta})$
M_1	op_{11}^{β}	0	0	0	0	0
	op_{14}^{β}	0	0	0	0	0
M_2	op_{12}^{β}	1	0	0	0	1
M_3	op_{31}^{β}	0	1	0	1	1
M_4	op_{15}^{β}	0	0	0	0	0
Cycle n° : 2, Pièce(s) défaillante(s) : \emptyset						
Machine	$op_{\alpha i}^{\beta+1}$	$\theta(op_{\alpha i}^{\beta+1})$	$I(op_{\alpha i}^{\beta+1})$	$I_m(op_{\alpha i}^{\beta+1})$	$I_g(op_{\alpha i}^{\beta+1})$	$R(op_{\alpha i}^{\beta+1})$
M_1	$op_{11}^{\beta+1}$	0	0	0	0	0
	$op_{14}^{\beta+1}$	0	0	0	0	0
M_2	$op_{12}^{\beta+1}$	1	0	0	0	1
M_3	$op_{31}^{\beta+1}$	0	1	0	1	1
M_4	$op_{15}^{\beta+1}$	0	0	0	0	0

Tableau IV.5 – Détectabilité de l'opération op_{12}^{β} de la machine M_2

Cycle n° : 10, Pièce(s) défaillante(s) : \emptyset						
Machine	$op_{\alpha i}^{\beta+9}$	$\theta(op_{\alpha i}^{\beta+9})$	$I(op_{\alpha i}^{\beta+9})$	$I_m(op_{\alpha i}^{\beta+9})$	$I_g(op_{\alpha i}^{\beta+9})$	$R(op_{\alpha i}^{\beta+9})$
M_1	$op_{11}^{\beta+9}$	0	0	0	0	0
	$op_{14}^{\beta+9}$	0	0	0	0	0
M_2	$op_{12}^{\beta+9}$	1	0	0	0	1
M_3	$op_{31}^{\beta+9}$	0	1	0	1	1
M_4	$op_{15}^{\beta+9}$	0	0	0	0	0

Tableau IV.6 – Opération non détectable

qui absorbent le retard.

Machines	Opérations	DéTECTABILITÉ	Explications
M_1	op_{11}^β	2 cycles	Induction machine + Induction gamme
	op_{14}^β	2 cycle	Induction gamme
M_2	op_{12}^β	NON	Marge gamme = 2 u.t., Marge machine = 2 u.t.
M_3	op_{13}^β	NON	Marge gamme = 2 u.t., Marge machine = 2 u.t.
M_4	op_{15}^β	1 cycle	Opération observable

Tableau IV.7 – Etude de la détectabilité d'un SPFM à plusieurs en-cours

Le système de production étudié est considéré non détectable puisqu'il contient deux opérations non détectables, par conséquent, le système est non diagnosticable.

4 Vers un ordonnancement diagnosticable

Dans la première partie de ce chapitre, nous avons proposé une méthode qui permet de vérifier la détectabilité d'une défaillance dans un SFPM. Cependant la simple vérification de la détectabilité n'est plus suffisante dans certains cas à cause du coût élevé du diagnostic pour les systèmes complexes. Ceci justifie la nécessité à chercher à déterminer les causes qui rendent une défaillance non diagnosticable et à proposer des solutions qui permettent de les éliminer.

4.1 Solutions de diagnosticabilité

Afin de garantir la diagnosticabilité d'un système de production, plusieurs modifications sur le modèle d'ordonnancement peuvent être considérées selon la méthode de diagnostic adoptée. Dans la littérature, le principal type de modification consiste à améliorer l'observabilité de système en optimisant le positionnement des capteurs sur les composants de système et leur nombre [Debouk *et al.*, 1999] ou encore en adoptant des types précis de capteurs [Jiang *et al.*, 2003] selon la criticité de l'évènement (des capteurs d'événements locaux ou de communication, des capteurs intelligents pour l'acquisition active de l'information [Thorsley et Teneketzis, 2007]). Ces travaux présupposent que les capteurs ajoutés sont fiables et non bruités.

Dans le contexte de notre travail, nous envisageons deux types de solutions pour résoudre la problématique de diagnosticabilité. En se basant sur l'étude de détectabilité effectuée dans la première partie de ce chapitre, nous avons pu dégager une synthèse sur les opérations non détectables qui causent un problème de détectabilité. La première solution que nous proposons repose sur la modification de l'architecture de surveillance en ajoutant des capteurs surveillant ces opérations. Considérons comme exemple le système de production présenté dans la figure I.6, l'étude de diagnosticabilité a montré que l'opération op_{12}^β de la machine M_2 est non détectable. Une solution triviale est d'ajouter un capteur de surveillance supplémentaire pour cette machine.

Etant donné l'éventuel impact de l'ajout des capteurs dans un système, certains systèmes exigent que l'observabilité soit limitée à la sortie de système de production. Dans ce cas, nous proposons une seconde solution qui se base sur la modification du modèle de l'ordonnancement. Cette modification intervient au niveau de la phase de conception de système qui se traduit par un ajustement des marges des opérations. Dans la suite, nous introduisons le problème de la conception d'un ordonnancement cyclique afin d'améliorer son degré de diagnosticabilité. Cette conception s'appuie sur un problème d'optimisation des coûts d'ordonnancement et de diagnostic.

4.2 Propriétés d'un ordonnancement diagnosticable

Dans ce travail, l'objectif est d'établir un retour à l'ordonnancement sous forme de pré-requis afin de garantir la diagnosticabilité d'un SFPM avec une observabilité limitée à la sortie de système de production. Nous cherchons à déterminer les règles pour les concepteurs de l'ordonnancement cyclique afin d'améliorer et de garantir les objectifs de diagnosticabilité. Soient les fonctions suivantes :

- $\Delta_g(op_{\alpha i}^\beta)$: la fonction qui calcule la marge en aval de l'opération $op_{\alpha i}^\beta$ d'une gamme de pièce,
- $\Delta_m(op_{\alpha i}^\beta)$: la fonction qui calcule la marge en aval de l'opération $op_{\alpha i}^\beta$ sur une machine,
- $Observable(op_{\alpha i}^\beta)$: la fonction booléenne qui teste si une opération est observable,
- $Detectable(op_{\alpha i}^\beta)$: la fonction booléenne qui teste si la défaillance d'une opération est détectable,
- $R_{min}(op_{\alpha i}^\beta)$: le retard total minimal d'une opération.

$$NonDetectable (op_{\alpha i}^{\beta}) \text{ ssi } \begin{cases} NonObservable (op_{\alpha i}^{\beta}) \\ \mathbf{ET} \\ Max (\Delta_g (op_{\alpha i}^{\beta}), \Delta_m (op_{\alpha i}^{\beta})) > R_{min} (op_{\alpha i}^{\beta}) \end{cases} \quad (IV.1)$$

En effet, pour que la défaillance d'une opération soit détectable il faut que cette dernière soit observable ou son retard soit strictement supérieur à au moins une des deux marges en aval de l'opération considérée (gamme ou machine).

5 Conclusion

Nous avons présenté au cours de ce chapitre l'intérêt de mesurer l'aptitude d'une approche à diagnostiquer tout défaut dans un temps fini. Nous avons proposé une approche qui étudie la détectabilité d'un ordonnancement cyclique d'un SFPM. Cette étude permet de savoir si le modèle du système est suffisamment riche en information pour permettre l'identification de tous les défauts affectant son fonctionnement. Si l'ordonnancement étudié est estimé détectable, notre approche mesure le nombre de cycles maximal pour qu'une défaillance soit détectée et énumère l'ensemble de retards pronostiqués (prédire le nombre de pièces en retard) suite à la défaillance d'une opération.

Conclusion générale et perspectives

Dans cette thèse nous nous sommes intéressés à la problématique du diagnostic indirect des systèmes flexibles de production manufacturière (SFPM). La fonction de diagnostic consiste à détecter une défaillance, de localiser son origine et de déterminer ses causes. L'intérêt de diagnostic indirect est justifié par le besoin de réduire l'utilisation des capteurs de surveillance susceptibles à la détérioration et qui peuvent diminuer la disponibilité du système. Le principe de diagnostic indirect proposé dans ce travail consiste à comparer le flux de production réel par rapport au flux prévisionnel. Dans la littérature, cette comparaison est effectuée à la fin de la période de production ce qui entraîne un retard dans la détection des écarts. Afin d'améliorer la réactivité du diagnostic indirect nous avons opté pour une commande cyclique qui permet de définir des cycles au niveau du système de production et donc d'avoir des références régulières en termes d'observation et de contrôle de la production en sortie. Dans ce contexte, nous avons considéré un ordonnancement cyclique par rapport aux machines permettant de garantir le temps de cycle optimal, et par conséquent minimiser la période d'observation.

Pour aborder le problème du diagnostic indirect, nous avons proposé dans une première étape une approche par analyse de flux de production. Cette approche se base principalement sur la construction de l'arbre des hypothèses par chaînage arrière. La génération de cet arbre est lancée dès la première détection d'un symptôme initiateur par l'observation d'un retard à la sortie d'une pièce. Ensuite il s'agit de valider ces hypothèses en estimant leur l'impact quantitatif et temporel sur le système en utilisant le chaînage avant. La validation de cette approche a montré la capacité de diagnostiquer un SFPM en se limitant à l'observation des sorties des pièces. Toutefois, le diagnostic peut ne pas converger vers une solution unique. En effet, les observations d'un seul cycle ne sont pas toujours suffisantes pour identifier la cause première d'une défaillance. Il est alors nécessaire de poursuivre le diagnostic dans les cycles suivants de l'ordonnancement afin d'espérer raffiner l'ensemble des solutions. D'où le besoin d'envisager

une stratégie incrémentale de diagnostic en ligne pour résoudre le problème de convergence.

Afin de pallier les insuffisances de la première contribution, nous avons adopté dans une deuxième étape une approche incrémentale de diagnostic. Pour répondre aux exigences de cette approche et pour s'adapter à l'ajout des contraintes en cours d'exécution de diagnostic, nous avons opté à une méthode basée sur la programmation par contraintes. Cette approche se base sur la construction d'un modèle mathématique décrivant les paramètres d'un SFPM et modélisant les contraintes d'induction du retard et les contraintes de précedence entre les opérations tout en respectant le séquençement des opérations dans une gamme ou une machine. Afin de résoudre ce modèle nous avons choisis un solveur dédié à la programmation par contraintes qui permet d'énumérer l'ensemble des solutions valides pour un ensemble d'observations et cohérentes avec l'ensemble des contraintes du modèle. Le principe de diagnostic incrémental proposé dans la deuxième approche consiste à ajouter régulièrement chaque observation obtenue au cours de la production et poursuivre la résolution jusqu'à qu'elle converge vers une solution unique. Cette approche incrémental a la capacité de mettre à jour les résultats de résolution lorsque de nouvelles observations sont disponibles sans reprendre la résolution dès le premier cycle de diagnostic. Ce qui a permis de raffiner les solutions et aboutir à la cause première de défaillance. L'exécution des différents scénarios de défaillances a démontré l'efficacité du diagnostic incrémental par rapport à l'approche par arbre de résolution. L'étude de performance de diagnostic a montré que le temps de résolution reste négligeable par rapport à la constante de temps des problèmes manufacturiers et à la durée du cycle d'ordonnancement. Cette évaluation démontre la double efficacité de l'approche incrémentale, en termes de convergence et temps de résolution.

Afin de prévoir la capacité de l'approche proposée à diagnostiquer une commande cyclique d'un SFPM, nous avons proposé une étude de diagnosticabilité. Cette étude permet de savoir si les observabilités prévus pour le système sont suffisantes pour localiser et identifier au bout d'un temps fini les causes premières de défaillance. Dans ce cadre, nous avons proposé une méthode permettant de vérifier la détectabilité d'un ordonnancement cyclique d'un SFPM. Une fois l'ordonnancement étudié est estimé détectable, la méthode proposée mesure alors le nombre de cycles maximal pour qu'une défaillance soit détectée et prédit le nombre de pièces en retard suite à la défaillance d'une opération.

Au terme de ces travaux de recherche que nous avons menés, plusieurs perspectives peuvent être envisagées, à court terme ou à long terme. Nous les avons regroupées selon quatre axes :

Dans un premier axe, nous visons à concevoir une méthode de résolution spécifique au problème étudié en se basant sur le modèle mathématique proposé. En effet, la méthode utilisée dans cette thèse est celle par défaut du solveur IlogSolver. Certes, ce solveur présente des performances de résolution intéressantes. Cependant le développement d'un algorithme de résolution spécifique se basant sur le modèle mathématique de diagnostic et de diagnosticabilité représente une alternative intéressante pour la résolution. Nous pouvons nous inspirer des approches proposées de la programmation par contraintes [Methlouthiand et Bouamama, 2011]. Nous proposons ainsi dans ce contexte d'étudier l'efficacité des algorithmes internes d'IlogSolver par rapport à notre problème.

Dans un deuxième axe, nous envisageons étudier la problématique inverse de conception d'ordonnancement permettant de garantir la diagnosticabilité d'un SFPM par approche indirecte compte tenu de taux de dérive progressive de la durée des opérations de transformation ou d'assemblage. Il s'agit de concevoir un ordonnancement diagnosticable qui devrait assurer les caractéristiques de l'ordonnancement cyclique (optimisation du temps de cycle et minimisation l'en-cours) pour optimiser la production et celles de diagnosticabilité (capacité de détectabilité et d'isolabilité des défauts).

Dans un troisième axe, nous visons à étendre notre démarche de diagnostic indirect dans le cadre des défaillances multiples. Ceci revient à relâcher ainsi l'hypothèse de défaillance unique et celle de défaillance d'une seule fonction par ressource. En effet, dans le cadre de notre étude, la limitation de défaillance unique a été basée sur le fait que l'ordonnancement cyclique n'est pertinent que dans l'hypothèse de pannes rares. Cependant, certains systèmes complexes sont souvent sujets à des dysfonctionnements.

Dans un dernier axe et pour le long terme, nous suggérons quelques pistes exploratoires. Nous proposons d'élargir notre approche de diagnostic indirect et incrémental à d'autres formes d'ordonnancement que les ordonnancements 1-cyclique et de l'adapter à tous les modes de marches d'un SFPM. Finalement, nous pourrions envisager une adaptation de l'approche à une décomposition d'un SFPM en ateliers, cellules, ressources ou sous-système. L'idée ici est de pouvoir descendre jusqu'à l'identification des sous-systèmes dans une ressource responsable de défaillances. Dans ce cadre, il impératif de relâcher l'hypothèse de défaillance unique par ressource.

Bibliographie

- ABED, M., EZZEDDINE, B., ABDELLATIF, B. et GRABARA, J. (2013). An agent – based framework for supply chain cooperation. *Applied Mechanics and Materials*, 309:185–194. 13
- BARBIER, F. et JAULENT, P. (1992). *Les techniques orientées objet et CIM*. Editions Eyrolles. 14
- BELKAHLA, O., YIM, P., KORBAA, O. et GHEDIRA, K. (2007). A distributed transient inter-production scheduling for flexible manufacturing systems. *Journal Européen des Systèmes Automatisés*, 41(1):101–123. 16
- BENAMAR, A. (2013). *Ordonnancement cyclique des systèmes contraints*. Thèse de doctorat, Université de Manouba, École Nationale des Sciences de l’Informatique. 16, 17
- BERRUET, P., TOGUYENI, A.-K.-A., ELKHATTABI, S. et CRAYE, E. (2000). Toward an implementation of recovery procedures for flexible manufacturing systems supervision. *Computers in Industry*, 43(3):227–236. 9
- BHOWAL, P., SARKAR, D., MUKHOPADHYAY, S. et BASU, A. (2007). Fault diagnosis in discrete time hybrid systems a case study. *Information Sciences : an International Journal*, 177(5): 1290–1308. 22, 25
- BOUFAIED, A. (2003). *Contribution à la surveillance distribuée des systèmes à événements discrets complexes*. Thèse de doctorat, Université Paul Sabatier de Toulouse. 10
- BOUYER, P., CHEVALIER, F., KRICHEN, M. et TRIPAKIS, S. (2005). Observation partielle des systèmes temporisés. *Journal Européen des Systèmes Automatisés (JESA)*, Lavoisier, pages 381–393. 87
- BRUNET, J., LABARRERE, M., JAUME, D., RAULT, A. et VERGE, M. (1990). *Détection et diagnostic de pannes : approche par modélisation*. *Traité des Nouvelles Technologies, série Diagnostic et Maintenance*, Hermès. 9, 22, 25
- CAMUS, H., OHL, H., KORBAA, O. et GENTINA, J. (1996). Cyclic schedules in flexible manufacturing systems with flexibilities in operating sequences. *17th International conference on Application and Theory of Petri Nets, Osaka, Japon*, pages 97–116. 17

- CANDAN, G. et YAZGAN, H. (2015). Genetic algorithm parameter optimisation using taguchi method for a flexible manufacturing system scheduling problem. *International Journal of Production Research (IJPR)*, 53(33):897–915. 16
- CASSEZ, F. (2012). The complexity of codiagnosability for discrete event and timed systems. *IEEE Transactions on Automatic Control*, 57(7):1752–1764. 24
- CHIACCHIO, P. et TOMMASI, G. (2009). An efficient approach for online diagnosis of discrete event systems. *IEEE Transactions On Automatic Control*, 54(4). 20
- CHRETIENNE, P., COFFMAN, E., LENSTRAAND, J. et LIU., Z. (1997). *Scheduling Theory and its applications*. Wiley editorials. ISBN : 978-0-471-94059-3. 17
- CIMATTI, A., C. P. et CAVADA., R. (2003). Formal verification of diagnosability via symbolic model checking. *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, Acapulco, Mexico,, pages 363–369. 90
- COCQUEMPOT, V., MEZYANI, T. et STAROSWIECKIY, M. (2004). Fault detection and isolation for hybrid systems using structured parity residuals. *5th Asian Control Conference, Victoria, Australia (ASCC04)*, 2:1204–1212. 22, 25
- COMBACAU, M. (1991). *Commande et surveillance des systèmes à événements discrets complexes : applications aux ateliers flexibles*. Thèse de doctorat, Université Paul Sabatier de Toulouse. 9, 10
- COMBACAU, M., BERRUET, P., CHARBONNAUD, F. et KHATAB, A. (2000). Réflexions sur la terminologie : Surveillance – supervision. groupement pour la recherche en productique, systèmes de production sûrs de fonctionnement. <http://home-pages.laas.fr/combacau/SPSF/sursup.html>. 1, 6, 9, 11, 21
- CONSOLE, L., PICARDI, C. et RIBAUDO, M. (2000). Diagnosing and diagnosability analysis using pepa. *Proceedings of the 14th European Conference on Artificial Intelligence, (ECAI-00)*, Berlin, Germany, pages 131–135. 90
- CONTANT, O., LAFORTUNE, S. et TENEKETZIS, D. (2006). Diagnosability of discrete event systems with modular structure. *Discrete Event Dynamic Systems*, 16(1):9–37. 89
- DARKHOVSKI, B. et STAROSWIECKI, M. (2003). A game–theoretic approach to decision in FDI. *IEEE Transactions On Automatic Control*, 48(5):853–858. 10
- DEBOUK, R., LAFORTUNE, S. et TENEKETZIS, D. (1999). On an Optimization Problem in Sensor Selection for Failure Diagnosis. *38th Conference on Decision and Control, Phoenix, Arizona, USA*, pages 4990–4995. 99
- DEBOUK, R., LAFORTUNE, S. et TENEKETZIS, D. (2000). Coordinated decentralized protocols for failure diagnosis of discrete events systems. *Discrete Event Dynamic Systems : Theory and Application (Kluwer Academic Publishers, Boston. Manufactured in The Netherlands)*, 10:33–86. 85, 89
- DEBOUK, R., MALIK, R. et BRANDIN, B. (2002). A modular architecture for diagnosis of discrete event systems. *IEEE Conference on Decision and Control, Las Vegas USA*, 10:417–422. 89

- DING, S. (2008). *Model-Based Fault Diagnosis Techniques : Design Schemes, Algorithms, and Tools*. Springer-Verlag. 1, 6
- DOUSSON, C. (1994). *Suivi d'Évolutions et Reconnaissance de Chroniques*. Thèse de doctorat, Laboratoire d'Analyse et d'Architecture des Systèmes, Toulouse. 22
- DUBUISSON, B. (1990). *Diagnostic et reconnaissance des formes*. Traité des Nouvelles Technologies, série Diagnostic et Maintenance, Hermès Sciences. 21
- ERSCHLER, J., LEVEQUE, D. et ROUBELLAT, F. (1982). Periodic loading of flexible manufacturing systems. (*IFIP*) *Congress, APMS, Bordeaux, France*, pages 327–339. 17
- FAKHFAKH, O., KORBAA, O. et TOGUYENI, A.-A.-K. (2012). Double chaining approach for indirect monitoring of fms under cyclic scheduling. *Information Control Problems in Manufacturing, INCOM'12 Bucharest, Romania*, 14(1):151–157. 47, 71
- FAKHFAKH, O., TOGUYENI, A.-A.-K. et KORBAA, O. (2013). Toward incremental diagnosis of fms faults based on constraint programming. *In the first international conference on Reasoning and Optimization in Information Systems. ROIS'2013, Hammam- Sousse, Tunisia*, pages 85–92. 70, 82
- FILHO, M.-G., BARCO, C.-F. et TAVARES, R.-F. (2014). Using genetic algorithms to solve scheduling problems on flexible manufacturing systems (fms) : a literature survey, classification and analysis. *Flexible Services and Manufacturing Journal*, 26(3):408–431. 16
- FLISS, I. et TAGINA., M. (2013). Diagnosing multiple faults in dynamic hybrid systems. *Intelligent Informatics Advances in Intelligent Systems and Computing, (ISI'12) Chennai, India*, 182:129–139. 20
- GENC, S. et LAFORTUNE, S. (2003). Distributed diagnosis of discrete-event systems using petri nets. *24th International Conference on Applications and theory of Petri Nets ICATPN, Eindhoven, Netherlands, Lecture Notes in Computer Science LNCS 2679*, pages 316–336. 22, 88
- GENTIL, S. (2007). *Supervision des procédés complexes*. Traité IC2, Série systèmes automatisés, Hermes science publications / Lavoisier. 1, 22, 25
- GENTINA, J.-C., KORBAA, O. et CAMUS, H. (2001). *Problèmes d'ordonnancement cyclique*, chapitre 7 dans *Ordonnancement de la Production*, pages 197–223. 16
- GERTLER, J. et SINGER, D. (1990). A new structural framework for parity equation based failure detection and isolation. *Automatica (Journal of IFAC)*, 26(2):381–388. 22
- GOMAA, M. et GENTIL, S. (1996). Hybrid industrial dynamical system supervision via hybrid continuous causal petri nets. *Symposium on Discrete Events and Manufacturing Systems, Lille, France, Springer - Verlag, IEEE SMC CESA96 - IMACS*, pages 380–384. 22
- GRASTIEN, A. et ANBULAGAN, A. (2009). Incremental diagnosis of DES with a non-exhaustive diagnosis engine. *20th International Workshop on Principles of Diagnosis (DX-09) Stockholm*, pages 345–352. 24
- GRASTIEN, A. et ANBULAGAN, A. (2010). Diagnostic de Systèmes à Événements Discrets à base de cohérence par SAT. *Revue d'Intelligence Artificielle*, 24(6):757–786. 24, 85, 91

- GRASTIEN, A., HASLUM, P. et THIEBAUX, S. (2012). Conflict-based diagnosis of discrete event systems : theory and practice. *13th International Conference on the Principles of Knowledge Representation and Reasoning (KR-12), Rome, Italy*, pages 489–499. [24](#)
- HAAR, S., BENVENISTE, A., FABRE, E. et JARD, C. (2003). Partial order diagnosability of discrete event systems using petri net unfoldings. *Proceedings of the 42th IEEE Conference on Decision and Control, Maui, HI*, 4:3748–3753. [88](#)
- HANEN, C. (1995). *Problèmes d’ordonnancement cycliques*. Habilitation à diriger les recherches, l’IPT, institut blaise pascal, université paris vi, Université Paris VI. [16](#)
- HILLION, H.-P. (1989). *Modélisation et analyse des systèmes de production discrets par les réseaux de Petri temporisés*. Thèse de doctorat, Université Paris VI. [16](#), [17](#)
- ISERMANN, R. (1984). Process fault detection based on modeling and estimation methods – a survey. *Automatica (Journal of IFAC)*, 20(4):387–404. [10](#)
- JIANG, S., HIANG, Z., CHANDRA, V. et KUMAR, R. (2001). A polynomial algorithm for testing diagnosability of discrete event systems. *IEEE Transactions on Automatic Control*, pages 1318–1321. [85](#), [87](#), [88](#), [89](#)
- JIANG, S., KUMAR, R. et GARCIA, H.-E. (2003). Optimal sensor selection for discrete event systems under partial observation. *IEEE Transactions on Automatic Control*, 48(3):369–381. [99](#)
- KATS, V. et LEVNER, E. (2011). A faster algorithm for 2 – cyclic robotic scheduling with a fixed robot route and interval processing times. *European Journal of Operational Research*, 209(1):51–56. [17](#)
- KERMAD, L. (1996). *Contribution à la supervision et à la gestion des modes et des configurations des Systèmes Flexibles de Production Manufacturière*. Thèse de doctorat, Université des Sciences et Technologies de Lille 1. [viii](#), [14](#)
- KORBAA, O. (1998). *Commande cyclique des systèmes flexibles de production manufacturière à l’aide des réseaux de Petri : de la planification à l’ordonnancement des régimes transitoires*. Thèse de doctorat, Université de Lille 1. [15](#), [16](#), [17](#), [19](#), [31](#)
- KORBAA, O. (2003). *Contribution à la conception et l’optimisation des systèmes de transport et de production*. Habilitation à diriger les recherches, Université de Lille 1. [14](#)
- KORBAA, O., BENASSER, A. et YIM, P. (2003). Two FMS scheduling methods based on Petri nets : a global and a local approach. *International Journal of Production Research*, 41(7):1349–1371. [15](#)
- KORBAA, O., CAMUS, O. et GENTINA, J.-C. (2002). A new cyclic scheduling algorithm for flexible manufacturing systems. *International Journal of Flexible Manufacturing Systems*, 14(2):173–187. [17](#), [27](#), [33](#)
- KORBAA, O. et GENTINA, J.-C. (2003). *Optimisation de commandes cycliques par des réseaux de Petri*, chapitre Chapitre 10. Evaluation des performances des systèmes de production. [16](#), [17](#)
- KORBAA, O., YIM, P. et GENTINA, J.-C. (2000). Solving transient scheduling problems with constraint programming. *European Journal of Control (EJC)*, 6(6):511–524. [51](#)

- LEE, E.-J. (2006). *Reconfiguration dynamique de la commande d'un système manufacturier : approche par la synthèse de la commande*. Thèse de doctorat, Ecole Centrale de Lille. 9
- LIN, F. (1994). Diagnosability of discrete event systems and its applications. *Discrete Event Dynamic Systems, Kluwer Academic Publishers, Boston, USA*, 4:197–212. 85, 87
- LIN, F. et WONHAM, W.-M. (1988). On observability of discrete event systems. *Information sciences*, 4(33):173–198. 7
- LIU, B. (2014). *An Efficient Approach for Diagnosability and Diagnosis of DES Based on Labeled Petri Nets : Untimed and Timed Contexts*. Thèse de doctorat, Université des Sciences et Technologies de Lille 1. 22, 88
- LIU, B., GHAZEL, M. et TOGUYENI, A.-A.-K. (2012). K-diagnosability of labeled petri nets. *Proceeding of 9ème édition de la conférence MANifestation des JEunes Chercheurs en Sciences et Technologies de l'Information et de la Communication, MajecSTIC , Lille, France*. 88
- LIU, B., GHAZEL, M. et TOGUYENI, A.-A.-K. (2013). Evaluation à la volée de la diagnostiquabilité des systèmes à événements discrets temporisés. *Journal Européen des Systèmes Automatisés*, 47(1-3):227–242. 88
- LY, F. (1999). *Optimisation de la maintenance dans les SFPM : Approche par la structuration et la surveillance prédictive indirecte*. Thèse de doctorat, Université des Sciences et Technologies de Lille 1. 23, 25, 33
- LY, F., TOGUYENI, A.-A.-K. et CRAYE, E. (2000). Indirect predictive monitoring in FMS. *Robotics and computer integrated manufacturing*, 16(5):321–338. viii, 9, 33, 34, 71, 72, 79, 80, 81
- MANUAL, U. (2010). *IBM ILOG Solver Version 6.8*. 64, 77
- METHLOUTHIAND, I. et BOUAMAMA, S. (2011). New honeybee optimization for constraint reasoning : Case of max-csps. *Knowledge-Based and Intelligent Information and Engineering Systems, LNCS*, 6882:325–334. 104
- MHAMDI, L., DHOUBI, H., LIOUANE, N. et SIMEU-ABAZI, Z. (2013). Multiple fault diagnosis using mathematical models. *9th Asian Control Conference (ASCC), Istanbul*, pages 1–68. 24
- MORRISON, D. et UPTON, D. (1994). Fault diagnosis and computer integrated manufacturing systems. *IEEE Transactions on Engineering Management*, 41(1):69–83. 1, 8, 11
- MOSTERMAN, P.-J. (2001). Diagnosis of physical systems with hybrid models using parameterized causality. *4th International Workshop on Hybrid Systems : Computation and Control, Rome, Italy, LNCS*, 2034:447–458. 22
- NABLI, L. (2000). *Surveillance préventive conditionnelle prévisionnelle indirecte d'une unité de filature textile : approche par qualité*. Thèse de doctorat, Université des Sciences et Technologies de Lille 1. 9, 23, 25
- PENCOLE, Y. (2004). Diagnosability analysis of distributed discrete event systems. *Proceedings of the 16th European Conference on Artificial Intelligence ECAI-04, Valencia, Spain*, 1:43–47. 86, 89

- PENCOLE, Y., CORDIER, M.-O. et ROZE., L. (2000). Incremental decentralized diagnosis approach for the supervision of a telecommunication network. *Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas USA*, 1:435–440. [24](#)
- PERRIN, J., BINET, F., DUMERY, J.-J., MERLAUD, C. et TRICHARD, J.-P. (2004). *Automatique et informatique industrielle : Bases théoriques, méthodologiques et techniques*. Nathan Technique, ISBN : 2-09-179452-X. [13](#)
- PHILIPPOT, A. (2006). *Contribution au diagnostic décentralisé des systèmes à événements discrets : Application aux systèmes manufacturiers*. Thèse de doctorat, Université de Reims Champagne Ardenne. [6](#), [21](#)
- QIU, W. et KUMAR, R. (2004). Decentralized failure diagnosis of discrete event systems. *7th International Workshop on Discrete Event Systems (WODES'04), Reims, France*, pages 145–150. [90](#)
- RINTANEN, J. et GRASTIEN, A. (2007). Diagnosability testing with satisfiability algorithms. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07), Hyderabad, India*, pages 532–537. [91](#)
- SADDEM, R. (2012). *Diagnosticabilité modulaire appliquée au Diagnostic en ligne des Systèmes Embarqués Logiques*. Thèse de doctorat, Université de Lille 1. [87](#)
- SADDEM, R., TOGUYENI, A.-K.-A. et TAGINA, M. (2012). Algorithme d'interprétation d'une base de signatures temporelles causales pour le diagnostic en ligne des systèmes à événements discrets. *9ème Conférence Internationale de Modélisation, Optimisation et Simulation, MOSIM 2012. Bordeaux, France*. [22](#)
- SAMANTARY, A.-K. et OULD-BOUAMAMA, B. (2008). *Model-based Process Supervision : A Bond Graph Approach*. Springer Verlag, Series : Advances in Industrial Control, ISBN :978-1-84800-158-9. [22](#), [25](#)
- SAMPATH, M., SENGUPTA, R., LAFORTUNE, S. et SINNAMOH, K. (1995). Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575. [1](#), [7](#), [9](#), [85](#), [86](#), [87](#), [88](#), [89](#)
- SAYED-MOUCHAWEH, M., PHILIPPOT, A. et CARRÉ-MÉNÉTRIER, V. (2008). Decentralized diagnosis based on boolean discrete event models : application on manufacturing systems. *International Journal of Production Research*, 46(19):5469–5490. [21](#)
- SAYED-MOUCHAWEH, M., PHILIPPOT, A., CARRÉ-MÉNÉTRIER, V. et RIERA, B. (2005). Detectability and diagnosability of discrete event systems. *2nd International Conference on Informatics in Control, Automation and Robotics (ICINCO'05), Barcelona, Spain*, pages 149–154. [87](#)
- SCHUMANN, A. et PENCOLÉ, Y. (2007). Scalable diagnosability checking of event-driven systems. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07), Hyderabad, India*, pages 575–580. [89](#)
- SENGUPTA, R. (1998). Diagnosis and communication in distributed systems. *4th International Workshop on Discrete Event Systems (WODES'98), Cagliari, Italie*, pages 26–28. [89](#)

- SIMEU-ABAZI, Z., MASCOLO, M. et KNOTEK, M. (2010). Fault diagnosis for discrete event systems : modelling and verification. *Reliability Engineering and System Safety*, 95(4):369–378. 20
- STAROSWIECKI, M. (1994). La problématique et les approches de la surveillance des systèmes technologiques. *Journée d'Étude S3 : sûreté , surveillance, supervision Détection et localisation de défaillances. GDR automatique. Paris*. 22
- STAROSWIECKI, M. et COMTET-VARGA, G. (2001). Analytic redundancy relations for fault detection and isolation in algebraic dynamic systems. *Automatica*, 37(5):687–699. 22
- SU, R. (2004). *Distributed Diagnosis for Discrete-Event System*. Thèse de doctorat, University of Toronto, Canada. 24
- TAGINA, M., CASSAR, J.-P., DAUPHIN-TANGY, G. et STAROSWIECKI, M. (1995). Monitoring of systems modeled by bond graph. *International Conference on Bond Graph Modeling and Simulation, ICBGM'95, Las Vegas, USA*, pages 275–280. 22, 25
- TELMOUDI, A.-J. (2011). *Contribution à une surveillance indirecte suite à une commande robuste des systèmes : approche multicritères*. Thèse de doctorat, Université de Tunis. 10, 23
- TELMOUDI, A.-J., NABLI, L. et BOURJAUULT, A. (2008). Symptoms detection in fms through performance indicators follow-up : A quality-flow approach. *IEEE- International Conference on Condition Monitoring and Diagnosis, CMD-08, Pékin, Chine*, pages 803–807. 1, 23, 25
- THORSLEY, D. et TENEKETZIS, D. (2007). Active acquisition of information for diagnosis and supervisory control of discrete event systems. *Discrete Event Dynamic, Springer US*, 74(4):531–583. 99
- TOGUYENI, A.-K.-A. (1992). *Surveillance et diagnostic en ligne dans les ateliers flexibles de l'industrie manufacturière*. Thèse de doctorat, Université de Lille 1. 6, 8, 9, 10, 22
- TOGUYENI, A.-K.-A. et KORBA, O. (2005). Indirect monitoring of the failures of a Flexible Manufacturing Systems under cyclic scheduling. *Robotics and Computer-Integrated Manufacturing*, 21(1):1–10. 1, 10, 23, 25, 33, 34
- TOGUYENI, A.-K.-A., KOUISSI, K. et COMBACAU, M. (2002). *Supervision d'atelier*, chapitre 4 de Fondements du pilotage des systèmes de production. viii, 8
- TRIPAKIS, S. (2002). Fault diagnosis for timed automata. *Formal Techniques in Real-Time and Fault-Tolerant Systems, Oldenburg, Germany, LNCS, Springer, ISBN : 3-540-44165-4*, 2469:205–224. 85, 87
- VALENTIN, C. (1994). Modeling and analysis methods for a class of hybrid dynamic systems. *Symposium Automatisation des Processus Mixtes : Les Systèmes Dynamiques Hybrides, ADPM'94*, pages 221–226. 17, 18, 19
- VILLEMEUR, A. (1988). *Sûreté de Fonctionnement des Systèmes Industriels : Fiabilité - Facteurs humains - Informatisation*. 6
- WANG, Y., YOO, T.-S. et LAFORTUNE, S. (2007). Diagnosis of discrete event systems using decentralized architectures. *Journal of Discrete Event Dynamic Systems*, 17(2):233–263. 89

- WEN, Y.-L. et JENG, M. (2004). Diagnosability of petri nets. *IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands*, 5:4891–4896. [88](#)
- YOO, T. et LAFORTUNE, S. (2002). Polynomial – time verification of diagnosability of partially – observed discrete – event systems. *IEEE Transactions on Automatic Control*, 47(9):1491–1495. [85](#), [88](#)
- ZEMOURI, M.-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. [21](#)
- ZWINGELSTEIN, G. (1995). Diagnostic des défaillances - théorie et pratique pour les systèmes industriels. *Traité des Nouvelles Technologies, série Diagnostic et Maintenance*. Hermès Science, France. [7](#), [21](#), [22](#), [25](#)

L'utilisation d'un grand nombre de capteurs de surveillance dans les systèmes automatisés de production peut diminuer notablement la disponibilité de ces systèmes en raison de la fragilité de ces instruments. Cependant le fait d'avoir peu d'observations du comportement interne du système provoque un manque de sensibilité ne permettant pas de détecter certaines défaillances de faibles amplitudes. Les travaux de cette thèse de doctorat s'inscrivent dans le cadre du diagnostic indirect des Systèmes Flexibles de Production Manufacturière (SFPM). Ils sont consacrés au développement d'un outil de diagnostic en ligne par analyse quantitative de la production. Afin d'augmenter la réactivité du diagnostic indirect, nous avons opté pour une commande cyclique qui permet de définir des cycles au niveau du système de production et d'avoir des références régulières en termes d'observation et de contrôle de la production en sortie. Les contributions présentées dans cette thèse comprennent essentiellement cinq volets qui mettent en place dans le cadre de l'hypothèse de défaillance unique, des algorithmes pour le diagnostic indirect, le diagnostic incrémental, le pronostic, l'étude de la diagnosticabilité et finalement l'introduction d'une nouvelle notion dans l'ordonnancement qui est la définition des propriétés d'un ordonnancement diagnosticable. En effet, l'approche de diagnostic proposée a les capacités de diagnostiquer un SFPM en se limitant à l'observation des sorties des pièces et à mettre à jour régulièrement le diagnostic à chaque nouvelle observation obtenue au cours de la production. Le diagnostic incrémental permet d'identifier la cause première d'une défaillance en raffinant l'ensemble des hypothèses obtenues par les observations précédentes. L'étude de la diagnosticabilité proposée permet de savoir si compte tenu des points d'observations du système, l'ordonnancement considéré pour la production courante sur le système permet de détecter, localiser et identifier au bout d'un temps fini les causes premières d'une défaillance. Afin de garantir la qualité de diagnostic, cette étude permet de mesurer le nombre de cycles maximal pour qu'une défaillance soit diagnostiquée.

Mots clés : SFPM, Diagnostic indirect, Diagnostic incrémental, Ordonnancement cyclique, Diagnosticabilité.

Using a large number of monitoring sensors in automated production systems can reduce the availability of the monitored system because of the fragility of these instruments. However having few observations of the internal behavior of the system causes a lack of sensitivity of some failures of small-amplitude. This study is about indirect diagnosis of fault in flexible manufacturing systems (FMS). For this purpose, we propose an on-line fault diagnosis of FMS based on flows analysis. In order to have additional time references to observe the system, one considers an FMS subjected to cyclic scheduling. Indeed, a deterministic command of cycling scheduling allows us to optimize the production flows and to control the production constantly by a precise knowledge of the system state. The major merits of this approach lie in its capacity to indirect diagnose faults on-line over several cycles and predict the delays carried out in the future. The incremental diagnosis does not restart the resolution after each new observation, but it pursues the resolution from the set of assumptions obtained with the previous observations until the system converges to a unique solution. The study of the diagnosability proposed in this work allows to check off-line before the production, if taking into account the choice of the system observation points, the current production scheduling is adapted to detect, locate and identify after a finite time the causes a failure.

Keywords :FMS, Indirect Diagnosis, Incremental Diagnosis, Cyclic Scheduling, Diagnosability.

