



HAL
open science

Contribution à une Surveillance Indirecte suite à une Commande Robuste des Systèmes : Approche Multicritères

Achraf Jabeur Telmoudi

► **To cite this version:**

Achraf Jabeur Telmoudi. Contribution à une Surveillance Indirecte suite à une Commande Robuste des Systèmes : Approche Multicritères. Automatique / Robotique. Université de Tunis; Ecole Supérieure des Sciences et Techniques de Tunis, 2011. Français. NNT : . tel-00614151

HAL Id: tel-00614151

<https://theses.hal.science/tel-00614151>

Submitted on 9 Aug 2011

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.

Université de Tunis
Ecole Supérieure des Sciences et Techniques de Tunis
Ecole Doctorale Sciences et Technologies

**Contribution à une Surveillance
Indirecte suite à une Commande
Robuste des Systèmes : Approche
Multicritères**

**Thèse en vue de l'obtention du Doctorat en
Génie Electrique**

Présentée et soutenue publiquement par

Achraf Jabeur TELMOUDI

Le 16 Avril 2011

Membres du jury :

Mr. Farhat FNAIECH	Professeur à l'Université de Tunis	Président
Mr. Lotfi NABLI	Maître de conférences à l'Université de Sousse	Directeur de thèse
Mr. Etienne CRAYE	Professeur à l'Ecole Centrale de Lille	Rapporteur
Mr. Moncef TAGINA	Professeur à l'Université de La Manouba	Rapporteur
Mr. Messaoud HASSANI	Professeur à l'Université de Monastir	Membre

Remerciement

Je tiens avant tout à remercier très vivement mon directeur de thèse **Radhi M'HIRI**, Professeur à la **Faculté des Sciences de Tunis**. Je lui suis absolument reconnaissant pour son encadrement, ses orientations et ses conseils.

Aussi, j'exprime toute ma profonde gratitude à **Lotfi NABLI**, Maître de conférences à l'**Institut Supérieur des Sciences Appliquées et de Technologie de Sousse** de m'avoir accueilli au sein de sa jeune équipe de **Commande, Surveillance et Sécurité de fonctionnement des Systèmes Automatisés de Production** subordonnée de l'Unité de Recherche **Automatique, Traitement de Signal et Image**. Je le remercie de m'avoir toujours encouragé vers l'avant, pour toute la confiance qu'il a porté en moi et qui m'a permis d'acquérir une précieuse expérience du métier d'enseignant chercheur. Je le remercie tout simplement pour sa sincère amitié et ses précieuses qualités humaines.

Je remercie pareillement les membres du jury pour la caution qu'ils ont bien voulu apporter à ce travail. J'adresse mes remerciements aux :

- Professeur **Farhat FNAIECH**, directeur de l'Unité de Recherche **Signal, Image et Commande Intelligente des Systèmes Industriels** de l'**Ecole Supérieure des Sciences et Techniques de Tunis**. Je le remercie particulièrement d'avoir accepté de présider le jury de soutenance,
- Professeur **Moncef TAGINA** de l'**Ecole Nationale des Sciences de l'Informatique** et Professeur **Etienne CRAYE**, directeur de l'**Ecole Centrale de Lille**, pour leur travail de lecture critique du manuscrit,
- Professeur **Messaoud HASSANI**, directeur de l'Unité de Recherche **Automatique, Traitement de Signal et Image** de l'**Ecole Nationale d'Ingénieurs de Monastir**, de m'avoir honoré en acceptant d'être examinateur.

Je suis très reconnaissant à Professeur **Mohamed ANNABI**, Ex-directeur de l'Unité de recherche **Systémique, Energétique, Productique et Environnement** de l'**Ecole Supérieure des Sciences et Techniques de Tunis**, pour son accueil chaleureux durant mon mastère et ma première année de thèse.

A l'âme de Professeur **Alain BOURJAULT**, Ex-directeur du **Laboratoire d'Automatique de Besançon**, que j'adresse mes remerciements les plus respectueux.

Je remercie très sincèrement Professeur **Naceur BENHADJ-BRAIEK**, directeur de l'**Ecole Supérieure des Sciences et Techniques de Tunis**.

*Ces années de thèse ont été pour moi l'opportunité de savoir des personnes exceptionnelles qui m'ont tout simplement offert leur sincère amitié, et avec qui j'ai passé d'agréables moments. Je remercie très vivement **Nidhal, Salim, Isam, Youssef, Ahmed, Ali, Hatem** et sa femme **Hayet, Sami, Mounir, Ilyes, Mohamed, Jalloul, Amira** et enfin **Bannour**, pour tout ce qu'on a vécu ensemble.*

*Enfin, merci à **mes parents** pour toute l'éducation qu'ils m'ont inculquée et à mon frère **Aymen**, ma sœur **Ameni** et ma fiancée **Siwar** de m'avoir encouragé et permis de réaliser une thèse de doctorat.*

Table des matières

Introduction générale.....	1
-----------------------------------	----------

Chapitre 1

Surveillance-Commande des Systèmes Industriels : État de l'art et proposition d'une nouvelle approche.....	4
-------------------------------------------------------------------------------------------------------------------	----------

1.1-	Introduction.....	6
1.2-	Concepts de base sur la conduite des systèmes flexibles de production manufacturière	6
1.2.1-	Systèmes Automatisés de Production	6
1.2.2-	Systèmes flexibles de production manufacturière.....	8
1.2.2.1-	Flexibilité physique de l'atelier	8
1.2.2.2-	Flexibilité décisionnelle	8
1.2.3-	systèmes à Evénements discrets.....	9
1.2.4-	Terminologies de référence.....	10
1.3-	Commande des systèmes à événements discrets.....	13
1.3.1-	Principe de Commande des SED	13
1.3.2-	Approches de Commande des SED	14
1.3.3-	Gestion des perturbations et mesure de robustesse	14
1.3.3.1-	Gestion des perturbations.....	14
1.3.3.2-	Mesure de robustesse.....	15
1.3.4-	Approches de commande robuste des SED.....	17
1.3.4.1-	Robustesse vis-à-vis des contraintes de temps.....	17
1.3.4.2-	La robustesse face à des contraintes non-temporelles.....	18
1.4-	Surveillance des systèmes à événements discrets	18
1.4.1-	Les fonctions de la surveillance	19
1.4.2-	Types de surveillance.....	20
1.5-	Intégration ou séparation de la surveillance à la commande.....	23
1.5.1-	Fusion de la surveillance au système de commande	24
1.5.1.1-	Approche de travaux d'A. Toguyeni [TOG 92].....	24
1.5.1.2-	Approche de Travaux de P. Berruet [BER 98], [BER 00], [BER 05].....	25
1.5.1.3-	Approche de Travaux de O. Kazushi et G. S. Kang de l'université de Michigan (USA) [KAZ 01]	26
1.5.1.4-	Travaux de l'équipe Génie Industriel et Logiciel (GIL) du Laboratoire d'Automatique et de Mécanique Industrielles et Humaines de Valenciennes (LAMIH) [PAR 92] [MAB 96]	27
1.5.1.5-	Travaux du Laboratoire Universitaire de Recherche en Production Automatisée (LURPA) [HAM 09]	28
1.5.2-	Séparation de la surveillance de la commande.....	29
1.5.3-	Approche mixte.....	30
1.5.3.1-	Approches de Travaux de LAAS.....	30
1.5.3.2-	Approches de Travaux de LAI.....	32
1.5.3.3-	Approches de Travaux de LAB	32
1.5.3.4-	Travail de Buyurgan de l'Université d'Arkansas [BUY 08].....	33
1.6-	Notre contribution : vers une Surveillance Indirecte suite à une Commande Robuste	33
1.6.1-	Bloc Commande.....	35
1.6.2-	Bloc Surveillance	34
1.6.3-	Explication d'échanges informationnels	35
1.6.4-	Les fonctions auxquelles on va s'intéresser	35
1.7-	Conclusion.....	36

Chapitre 2

Méthodes et outils pour la Commande et la Surveillance des systèmes industriels..... 39

2.1-	Introduction.....	41
2.2-	Modélisation de la robustesse des ateliers à contraintes temporelles et non temporelles	41
2.2.1-	Automates à états finis	42
2.2.2-	Réseaux de Petri.....	43
2.2.2.1-	RdP P-temporels	46
2.2.2.1.1-	Définition formelle.....	46
2.2.2.1.2-	Règles de fonctionnement.....	47
2.2.2.1.3-	Définition fondée sur l'âge des marques.....	47
2.2.2.1.4-	Définition fondée sur l'intervalle de temps.....	48
2.2.2.1.5-	Condition de franchissement d'une transition.....	48
2.2.2.1.5.1-	États d'un jeton.....	48
2.2.2.1.5.2-	Calcul de l'état suivant	49
2.2.2.1.6-	RdP P-temporels commandés	50
2.2.2.1.7-	Exemple d'application des RdP P-temporel.....	51
2.2.2.2-	RdP à Intervalles.....	52
2.2.2.2.1-	Définition formelle.....	52
2.2.2.2.2-	Règles de fonctionnement.....	53
2.2.2.2.3-	Définition d'un état	54
2.2.2.2.4-	Calcul de l'état suivant	54
2.2.2.2.5-	Les RdP P-temporels : une sous classe des RdP à intervalles [COL 03]	55
2.2.2.2.6-	Exemple d'application des RdP à Intervalles.....	56
2.2.2.3.1-	RdP t-temporels	57
2.2.2.3.1.1-	Définition formelle	57
2.2.2.3.1.2-	Etats et règle de tir.....	57
2.2.2.3.1.3-	Exemple d'application des RdP t-temporels.....	58
2.2.2.3.2-	RdP temporels à flux indépendants.....	59
2.2.2.3.2.1-	Définition formelle	59
2.2.2.3.2.2-	Règles de franchissement	59
2.2.2.3.2.3-	Exemple d'application des RdP temporels à flux indépendants.....	60
2.2.3-	Synthèse et choix des outils de modélisation adéquats pour notre approche	60
2.3-	Outils et méthode pour la surveillance des systèmes industriels.....	61
2.3.1-	Méthodes et outils de surveillance avec modèle	63
2.3.1.1-	Surveillance par modélisation fonctionnelle.....	63
2.3.1.1.1-	AMDEC	63
2.3.1.1.2-	Arbres de Défaillances	63
2.3.1.2-	Méthodes de surveillance par modélisation physique.....	64
2.3.1.2.1-	Redondances physiques et analytiques	64
2.3.1.2.1.1-	Redondances physiques.....	64
2.3.1.2.1.2-	Redondances analytiques.....	64
2.3.1.2.2-	Estimation paramétrique	65
2.3.2-	Méthodes et outils de surveillance sans modèle.....	67
2.3.2.1-	Surveillance avec outils statistiques.....	67
2.3.2.1.1-	Test de franchissement de seuils	67
2.3.2.1.2-	Test de moyenne et test de variance.....	67
2.3.2.2-	Surveillance avec outils symbolique.....	67
2.3.2.2.1-	Surveillance par modélisation symbolique	67
2.3.2.2.2-	Surveillance par modélisation causale	69
2.3.2.2.3-	Surveillance par reconnaissance des formes	69
2.3.2.2.3.1-	Reconnaitances des formes par utilisation des outils statistiques	70
2.3.2.2.3.2-	Reconnaitances des formes par utilisation de la logique floue	70
2.3.2.2.3.3-	Reconnaitances des formes par utilisation des réseaux de neurones	71
2.3.2.2.3.3.1-	Eléments de base des réseaux de neurones	71
2.3.2.2.3.3.2-	Apprentissage des Réseaux de neurones artificiels	73
2.3.2.2.3.3.3-	Architectures neuronales les plus utilisées en surveillance	74
2.3.2.2.3.3.3.1-	Perceptron Multi Couches (PMC)	74
2.3.2.2.3.3.3.2-	Réseau à Fonctions de base Radiales (RFR).....	75
2.3.2.2.3.3.4-	Réseaux de neurones temporels	77

2.3.3-	Synthèse et choix des outils de surveillance adéquats pour notre approche.....	77
2.4-	Conclusion.....	78

Chapitre 3

Modélisation et analyse d'une commande robuste multicritères..... 80

3.1-	Introduction.....	81
3.2-	Définition d'une terminologie.....	81
3.3-	Modèles RdP de commande pour la génération de robustesse : approche monocritère.....	82
3.3.1-	Mise en situation.....	82
3.3.2-	Modélisation d'un paramètre.....	83
3.3.2.1-	Modélisation d'un paramètre flux.....	83
3.3.2.2-	Modélisation d'un paramètre qualité.....	83
3.3.3-	Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop.....	84
3.3.3.1-	Modélisation d'un circuit élémentaire.....	84
3.3.3.1.1-	Modélisation d'un circuit élémentaire de flux C_p	84
3.3.3.1.1.1-	Cas où $b_{ai} \geq a_{pi+1}$	84
3.3.3.1.1.2-	Cas d'un goulot d'étranglement : $b_{ai} < a_{p.i+1}$	86
3.3.3.1.2-	Modélisation d'un circuit élémentaire de qualité C_q	88
3.3.4-	Aspect formel pour le modèle de robustesse hybride monocritère.....	90
3.3.4.1-	Aspect formel pour les circuits élémentaires.....	90
3.3.4.2-	Aspect formel pour un système de production.....	94
3.4-	Modèles RdP de commande pour la génération de robustesse d'une ressource : approche bi-critère....	97
3.4.1-	Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop.....	97
3.4.1.1-	Cas des variables qualitatives unitaires en parallèle.....	97
3.4.1.2-	Cas des variables qualitatives unitaires en série.....	99
3.4.2-	Aspect formel pour le modèle de robustesse hybride bi-critère d'une ressource.....	101
3.4.2.1-	Aspect formel pour le cas des variables qualitatives unitaires en parallèle.....	101
3.4.2.2-	Aspect formel pour le cas des variables qualitatives unitaires en série.....	102
3.5-	Modèles RdP de commande pour la génération de robustesse d'un Système de production: approche bi-critère.....	102
3.5.1-	Modélisation d'une commande robuste hybride bi-critère d'un système de production de type flow-shop.....	102
3.5.2-	Aspect formel pour le cas des variables qualitatives unitaires en série.....	104
3.5.3-	Application.....	105
3.6-	Conclusion.....	108

Chapitre 4

Surveillance Indirecte : approche multicritères..... 109

4.1-	Introduction.....	111
4.2-	Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales (RR ² FR) pour le pronostic ...	111
4.2.1-	Réseaux de neurones récurrents à fonctions de base radiales (RRFR).....	112
4.2.2-	Proposition d'une architecture améliorée des RFR : Le RR ² FR [TEL 09b].....	115
4.2.2.1-	Topologie de l'architecture RR ² FR.....	115
4.2.2.2-	Analyse de comportement de mémorisation et de corrélation d'un RR ² FR.....	116
4.2.2.3-	Apprentissage des RR ² FR.....	120
4.2.2.3.1-	Phase 1 : initialisation les paramètres de neurones bouclés.....	120
4.2.2.3.2-	Phase 2 : détermination des centres et rayons d'influence des neurones gaussiens.....	120
4.2.2.3.3-	Phase 3 : calcul des poids des connexions de sortie.....	122
4.2.2.4-	Application des RR ² FR.....	124
4.2.2.4.1-	Application des RR ² FR à la Série temporelle <i>MackeyGlass</i> [TEL 09b].....	124
4.2.2.4.1.1-	Simulation.....	124
4.2.2.4.1.2-	Interprétations.....	127

4.2.2.4.2-	Application des RR ² FR à la prédiction d'une concentration de sortie en CO ² d'un four à gaz.....	127
4.2.3-	Interprétation générale et apport de l'architecture RR ² FR.....	131
4.3-	Détection des symptômes par suivi des paramètres de loi de commande à travers les RdPTàO : approche qualité-flux.....	132
4.3.1-	Modèle de Combacau pour la détection des symptômes temporels [COM 91].....	132
4.3.2-	Modèle de Suivi-Détection multicritères : Qualité-flux [TEL 08c].....	134
4.3.3-	Application.....	137
4.3.4-	Interprétation générale et apport de Modèle de Suivi-Détection proposé.....	141
4.4-	Synchronisation entre le bloc Pronostic et le bloc Suivi-Détection.....	141
4.4.1-	Principe de communication.....	141
4.4.2-	Application.....	142
4.5-	Conclusion.....	144

Chapitre 5

Validation de notre approche sur un SFPM..... 147

5.1-	Introduction.....	148
5.2-	Présentation de la chaîne de production.....	148
5.2.1-	Présentation de la chaîne de production de double concentré de tomate.....	148
5.2.2-	Contrôle qualité de double concentré de tomate.....	150
5.2.3-	Paramètres temporels de la chaîne de production de double concentré de tomate.....	153
5.3-	Application de l'architecture « Surveillance Indirecte suite à une Commande Robuste » à la chaîne automatisée de production de DCT.....	156
5.3.1-	Modèle commande : commande robuste.....	156
5.3.1.1-	Loi de commande robuste activement bi-critère : qualité-flux.....	156
5.3.1.2-	Loi de commande robuste hybride bi-critère : qualité-flux.....	157
5.3.2-	Modèle Surveillance.....	162
5.3.2.1-	Fonction pronostic : prédiction de la qualité de produit DCT par RR ² FR.....	163
5.3.2.1.1-	Modèle RR ² FR pour la prédiction de Brix de produit DCT.....	164
5.3.2.1.2-	Base de données.....	165
5.3.2.1.3-	Simulation.....	166
5.3.2.1.4-	Interprétations.....	169
5.3.2.2-	Fonction suivi-détection.....	169
5.3.3-	Synchronisation entre les différents éléments de l'architecture.....	174
5.4-	Conclusion.....	174

Conclusion générale..... 177

Références bibliographiques..... 180

Introduction générale

Introduction générale

Dans un contexte mondial instable caractérisé par une concurrence très intense, les entreprises s'imposent aujourd'hui face aux exigences contraignantes du marché. Ces dernières sont dues à la demande des produits variés de haute qualité fabriqués en petites et moyennes séries. Les objectifs des entreprises face à cette situation sont synonymes à "zéro défaut", "zéro délai", "zéro stock" et "zéro panne". Une gestion globale et efficace des différentes structures opérantes de l'entreprise mène à la réalisation de ces objectifs.

Vu ce nouveau contexte, le système de production, qui présente l'unité opérationnelle terminale permettant la réalisation des perspectives de l'entreprise, doit s'adapter à ces nouvelles impératives. Ces dernières sont traduites au niveau du système de production par son aptitude à s'ajuster aux mutations de production et de se révéler flexible, réactif, robuste et proactif, afin de répondre aux exigences en diversité, en fluctuation, en productivité, en qualité, en optimisation des coûts de production et en amélioration de fiabilité par la réduction des risques des défaillances.

Pour répondre à ces contraintes, il est primordial d'étudier les tendances sollicitées du système de production ainsi que celles de son environnement. Cette étude nécessite une maîtrise du cycle de développement et de production des produits. Ceci est garanti par l'utilisation de méthodes, techniques et outils adaptés aux problématiques abordées dans les différentes phases du cycle de vie des produits (analyse, définition, production, exploitation, maintenance et fin de vie) tout en estimant les aléas.

Les systèmes automatisés de production ont présenté durant l'époque de production en masse un outil très performant pour l'assurance d'une haute productivité. La réalité du marché a implicitement imposé aux concepteurs le défi d'évoluer les caractéristiques de ces systèmes pour qu'ils s'adaptent réellement avec les notions sollicitées. L'objectif est que le système de production automatisé acquiert la capacité de compléter sa mission quelques soient les perturbations (internes ou externes) gênant l'évolution normale du processus de fabrication.

Les démarches de recherche contemporaine contribuent à l'amélioration et le renforcement de fermeté de la spécification et de la conception d'un système de conduite du procédé de production réactif non seulement face aux situations jugées normales, mais aussi face à celles qui ne sont pas connues a priori. En effet, Trois modules coopèrent afin d'assurer la réussite du système de conduite : la surveillance, la supervision et la commande.

Nous nous intéressons dans le cadre de ce travail à la problématique de Surveillance-Commande des Systèmes Flexibles de Production Manufacturière. Les études qui en découlent ont considéré discret, l'espace d'états, de ces systèmes dynamiques. Leur évolution est interprétée par l'occurrence d'événements discrets modifiant l'état du système.

Les systèmes de production manufacturière sont soumis généralement à des perturbations qui influent implicitement les sorties prescrites. Ceci nécessite une commande robuste de procédé permettant la conservation des aptitudes de système. En effet, la définition des intervalles de

conformité des paramètres de système doit toujours anticiper la phase de conception de la loi de commande cible qui devra garantir le respect de ces spécifications.

Nous prospectons, par cette thèse, à proposer une architecture de Surveillance-Commande performante en termes de maintien de la robustesse passive du système en vue de minimisation de recours à la maintenance corrective. En effet, une telle architecture doit comporter une commande robuste et une surveillance préventive.

Les systèmes concernés sont les Systèmes Flexibles de Production Manufacturière de type flow-shop ; au cas limite, les ateliers d'assemblage qui ont les caractéristiques intrinsèques d'un atelier flow-shop. En effet, la particularité de ce type d'atelier à cheminement unique est que les produits exploitent les machines dans le même ordre.

Notre objectif consiste donc à développer une architecture à caractère prévisionnel. Nous cherchons, en premier lieu, à concevoir une loi de commande robuste qui permet la gestion des perturbations fluctuant les marges de robustesse active et passive des paramètres de qualité de produit et de flux de production. En effet, l'étude de l'observabilité et de la constatation de type de la robustesse devient plus aisée par une telle commande. En second lieu, par l'installation d'un outil de surveillance indirecte, nous prospectons, à contrôler et analyser les variations des paramètres de la loi commande afin de maintenir leurs marges de robustesse.

Notre mémoire de thèse est structuré en cinq chapitres.

Dans le premier chapitre, nous présentons la problématique de conduite des Systèmes Automatisés de Production (SAP), d'où les Systèmes Flexibles de Production Manufacturière (SFPM) qui ont une propension aux Systèmes à Événements Discrets (SED). Ensuite, nous développons une étude bibliographique concernant la terminologie relative à notre problématique ainsi qu'aux architectures Surveillance-Commande. Nous clôturons ce chapitre par la présentation de notre architecture proposée : « Surveillance Indirecte suite à une Commande Robuste ».

Le second chapitre est consacré à la présentation et au choix des différents outils et méthodes de commande et de surveillance. Pour la commande, nous détaillons bien les propriétés des Automate à états finis et des Réseaux de Petri. Concernant la surveillance, nous présentons les outils et les méthodes avec modèle et sans modèle. Nos choix pour la commande et la surveillance sont bien justifiés.

Au troisième chapitre, nous développons notre modèle de commande hybride modélisant, à la fois, la robustesse passive et active et qui permet l'observation de la robustesse des ressources et l'interprétation de type de la robustesse globale des systèmes de production manufacturière face à des perturbations de flux et de qualité. Nous cumulons l'aspect de modélisation par un développement théorique.

Le quatrième chapitre s'intéresse à la présentation de notre méthodologie de surveillance indirecte préventive. En effet, nous présentons nos développements théoriques et applicatifs relatifs aux modèles des fonctions suivi-détection et pronostic. Nous terminons, par la mise en interaction de ces deux fonctions en faisant expliquer le principe de synchronisation et d'échanges informationnels.

Le dernier chapitre fait l'objet d'une application de notre architecture à une chaîne automatisée de production d'une industrie alimentaire.

Chapitre 1
Surveillance-Commande des
Systemes industriels : État de l'art et
proposition d'une nouvelle approche

Chapitre 1

Surveillance-Commande des Systèmes Industriels : État de l'art et proposition d'une nouvelle approche

Table des matières du chapitre 1

1.1-	Introduction	6
1.2-	Concepts de base sur la conduite des systèmes flexibles de production manufacturière.....	6
1.2.1-	Systèmes Automatisés de Production.....	6
1.2.2-	Systèmes flexibles de production manufacturière	8
1.2.2.1-	Flexibilité physique de l'atelier	8
1.2.2.2-	Flexibilité décisionnelle.....	8
1.2.3-	systèmes à Evénements discrets	9
1.2.4-	Terminologies de référence	10
1.3-	Commande des systèmes à événements discrets	13
1.3.1-	Principe de Commande des SED.....	13
1.3.2-	Approches de Commande des SED.....	14
1.3.3-	Gestion des perturbations et mesure de robustesse.....	14
1.3.3.1-	Gestion des perturbations	14
1.3.3.2-	Mesure de robustesse.....	15
1.3.4-	Approches de commande robuste des SED.....	17
1.3.4.1-	Robustesse vis-à-vis des contraintes de temps	17
1.3.4.2-	La robustesse face à des contraintes non-temporelles	18
1.4-	Surveillance des systèmes à événements discrets.....	18
1.4.1-	Les fonctions de la surveillance	19
1.4.2-	Types de surveillance	20
1.5-	Intégration ou séparation de la surveillance à la commande	23
1.5.1-	Fusion de la surveillance au système de commande	24
1.5.1.1-	Approche de travaux d'A. Toguyeni [TOG 92]	24
1.5.1.2-	Approche de Travaux de P. Berruet [BER 98], [BER 00], [BER 05]	25
1.5.1.3-	Approche de Travaux de O. Kazushi et G. S. Kang de l'université de Michigan (USA) [KAZ 01].....	26
1.5.1.4-	Travaux de l'équipe Génie Industriel et Logiciel (GIL) du Laboratoire d'Automatique et de Mécanique Industrielles et Humaines de Valenciennes (LAMIH) [PAR 92] [MAB 96]	27
1.5.1.5-	Travaux du Laboratoire Universitaire de Recherche en Production Automatisée (LURPA) [HAM 09]	28
1.5.2-	Séparation de la surveillance de la commande	29
1.5.3-	Approche mixte	30
1.5.3.1-	Approches de Travaux de LAAS.....	30
1.5.3.2-	Approches de Travaux de LAI	32
1.5.3.3-	Approches de Travaux de LAB	32
1.5.3.4-	Travail de Buyurgan de l'Université d'Arkansas [BUY 08].....	33
1.6-	Notre contribution : vers une Surveillance Indirecte suite à une Commande Robuste	33
1.6.1-	Bloc Commande.....	35
1.6.2-	Bloc Surveillance	34
1.6.3-	Explication d'échanges informationnels	35
1.6.4-	Les fonctions auxquelles on va s'intéresser.....	35
1.7-	Conclusion.....	36

1.1- Introduction

Pour être performante, une entreprise doit maîtriser le cycle de développement et de production des produits afin de garantir une certaine qualité de production dans un délai fixé par le client. Cette maîtrise passe par l'utilisation de méthodes, techniques et d'outils adaptés aux problématiques abordées dans les différentes phases du cycle de vie des produits (analyse, définition, production, exploitation et maintenance, fin de vie). Par ailleurs, pour répondre aux contraintes de délai de fabrication, les industriels doivent assurer la fiabilité et la disponibilité de leur outil de fabrication. Cela passe par la mise en place d'un système de conduite du procédé tenant compte de procédé lui-même et de son environnement. Trois modules participent à la réussite d'un tel système à savoir : la commande, la surveillance et la supervision.

Dans ce chapitre, nous présentons dans la première partie les concepts de base sur les Systèmes Flexibles de Production Manufacturière tout en introduisant les notions des Systèmes Automatisés de Production (SAP), ainsi que celles de la classe de Systèmes à Evénements Discrets (SED). L'étude qui en découle traite l'aspect opérationnel assimilé par l'atelier et l'aspect décisionnel présenté par le système de gestion associé. Nous abordons finalement, quelques définitions d'une liste de mots clés dans le domaine de la conduite des Systèmes à Evénements Discrets.

Dans la deuxième partie nous nous intéressons aussi bien à la description de structure de la commande des SED qu'aux travaux écoulés concernant cette fonction. Notre point d'intérêt a été posé d'une façon qu'on s'intéresse principalement aux travaux ayant traité la problématique de la commande robuste.

La troisième partie, s'occupe de la surveillance des SED. Nous débutons par une présentation des différentes fonctions de la surveillance. Puis, nous présentons les différents types de surveillance. Nous faisons notamment un état de l'art sur les approches de surveillance indirecte.

Dans la quatrième partie nous nous intéressons de la séparation ou de l'intégration de surveillance à la commande. Nous présentons des approches qui ont étudié principalement la problématique de la conception des architectures des systèmes Surveillance-Commande. La présentation de ces approches suit la classification suivante : fusion de la surveillance au système de commande, séparation de la surveillance de la commande et l'approche mixte.

La dernière partie de ce chapitre sera consacrée à la présentation de l'architecture de notre contribution.

1.2- Concepts de base sur la conduite des systèmes flexibles de production manufacturière

1.2.1- Systèmes Automatisés de Production

Un système peut être défini comme un ensemble de composants interdépendants, conçus pour réaliser une fonction donnée, dans des conditions données et dans un intervalle de temps donné. Pour chaque système, il est important de définir clairement les éléments qui le caractérisent, à savoir : la fonction, la structure, les conditions de fonctionnement, les

conditions d'exploitation et l'environnement dans lequel il est appelé à opérer. Ces derniers, présentent un Système de Production (SP), lorsqu'ils s'unissent afin d'élaborer un produit qui répond à des objectifs économiques et techniques. Ainsi, un SP peut être précisé par le résultat de plusieurs combinaisons entre l'ensemble de l'exploitation caractérisé par l'agencement particulier des facteurs de production, des choix techniques et des combinaisons des productions. Ces combinaisons doivent être gérées afin d'augmenter la valeur ajoutée désirée au départ tels que le poids, la forme, la dimension, etc., conformément à des objectifs de productivité, de qualité et de sûreté.

Un Système Automatisé de Production (SAP) est un SP qui possède une partie (ou la totalité) des ses activités automatisée. Par un schéma classique (Fig.1.1), nous représentons les blocs des SAP ainsi que les différentes interactions entre eux.

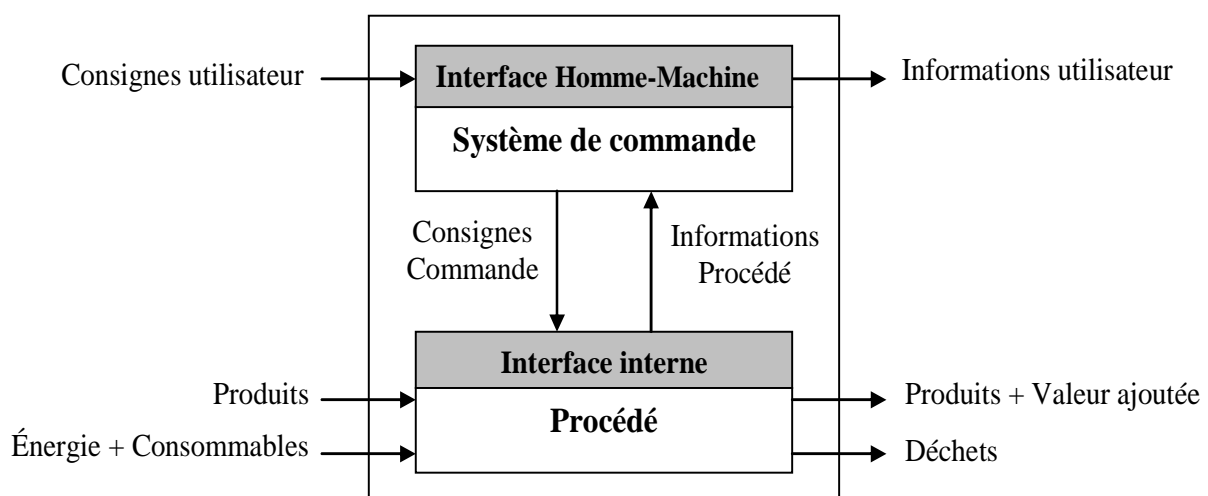


Fig. 1.1- Synthèse d'un SAP

On peut constater aisément qu'un SAP est composé de trois blocs principaux :

- Le bloc de procédé, appelé aussi partie opérative, englobe l'ensemble des entités de conversion physiques, spatiales, etc. (assemblage, soudage, transport, etc.) qui agissent directement sur la matière première ou sur le produit semi-fini afin de créer un produit fini. Ce bloc figure la définition du produit à obtenir grâce aux transformations effectuées par le processus.
- Le bloc de commande, appelé aussi partie commande, regroupe l'ensemble des moyens logiciels et matériels ainsi que les informations permettant la gestion du procédé. Selon les consignes de l'utilisateur et les informations reçues portant l'état de procédé, ce bloc produit des décisions, appelées consignes de commande, qui portent sur les actions à entreprendre par le procédé sur le produit.
- Le bloc d'interface interne assure le dialogue entre les deux autres blocs. Il met en forme les informations, respectivement les consignes, qui circulent entre le bloc de commande et celui de procédé. La fonctionnalité de l'interface est assurée par l'intermédiaire de deux classes de composants, à savoir : les actionneurs et les capteurs. Les actionneurs ont la fonction de transformer les consignes de commande

en actions physiques. Les capteurs représentent les instruments de mesure pour la commande. Ils se divergent selon la nature des quantités qu'ils mesurent.

1.2.2- Systèmes flexibles de production manufacturière

La disparition progressive des productions de type mono-produit qui ont été substituées par des fabrications multi-produits de haute qualité, impose aux entreprises, dans le domaine manufacturier, d'utiliser les Systèmes Flexibles de Production Manufacturière (SFPM).

Un SFPM est un système de production manufacturière souvent automatisé avec un grand degré de flexibilité. Il vise à répondre aux demandes variées du marché. Ces systèmes sont caractérisés par des critères liés à la haute rentabilité et à la souplesse de production qui envisagent la capacité d'adaptation [BER 98].

Dans la littérature, plusieurs définitions plus ou moins proches ont été introduites. Suivant [BER 98], la flexibilité est définie comme étant la capacité d'adaptation à la production pour un système non étudié. Cela suppose une adaptation totale du SP au produit courant ou son passage d'une configuration à une autre suite aux changements externes ou internes. Les mots-clés qui résument la flexibilité sont : la capacité et la vitesse de réaction, le potentiel d'absorption des discontinuités et l'adaptabilité.

On peut distinguer deux familles de flexibilité. La première est la flexibilité statique, elle se manifeste dans un environnement où les fluctuations sont bien connues et jugées par des distributions de probabilité. C'est le cas, par exemple, des industries à produits saisonniers. La deuxième est la flexibilité dynamique, elle apparaît dans un milieu où les variations ne sont pas régies par des lois connues [BOU 89].

Les critères de flexibilité doivent se manifester à deux niveaux : d'une part, au niveau de l'atelier qui accède à exécuter réellement la diversité de demandes énoncées par le marché, d'autre part, au niveau de l'organisation (système de gestion) qui doit posséder la capacité de gérer ces demandes [ZAM 97], [DAS 03a], [DAS 03b].

1.2.2.1- Flexibilité physique de l'atelier

Elle présente la capacité d'adapter les mutations de la production par des redondances et/ou des polyvalences des ressources de production agissant sur le produit et aussi par une organisation intelligente des ressources qui aboutit à piloter constamment des produits divers. En outre, cette flexibilité dépend des potentialités de chaque ressource et de l'agencement des équipements pour qu'ils soient capables de traiter la diversité des produits concernés. Sur le plan de sûreté de fonctionnement, la flexibilité physique de l'atelier est l'aptitude du SP à produire en présence d'une défaillance dans un élément du système.

1.2.2.2- Flexibilité décisionnelle

C'est "*la capacité de reconfiguration que le processus de production, compte tenu les potentialités du procédé, pour spécifier et coordonner la transformation d'un produit par les diverses ressources de l'atelier*" [DAS 03b].

[BEN 96] a mis en évidence une structure décisionnelle optimale de cinq niveaux hiérarchiques (Fig. 1.2) : Planification, Ordonnancement Prévisionnel, Ordonnancement Temps Réel, Coordination et Commande Locale.

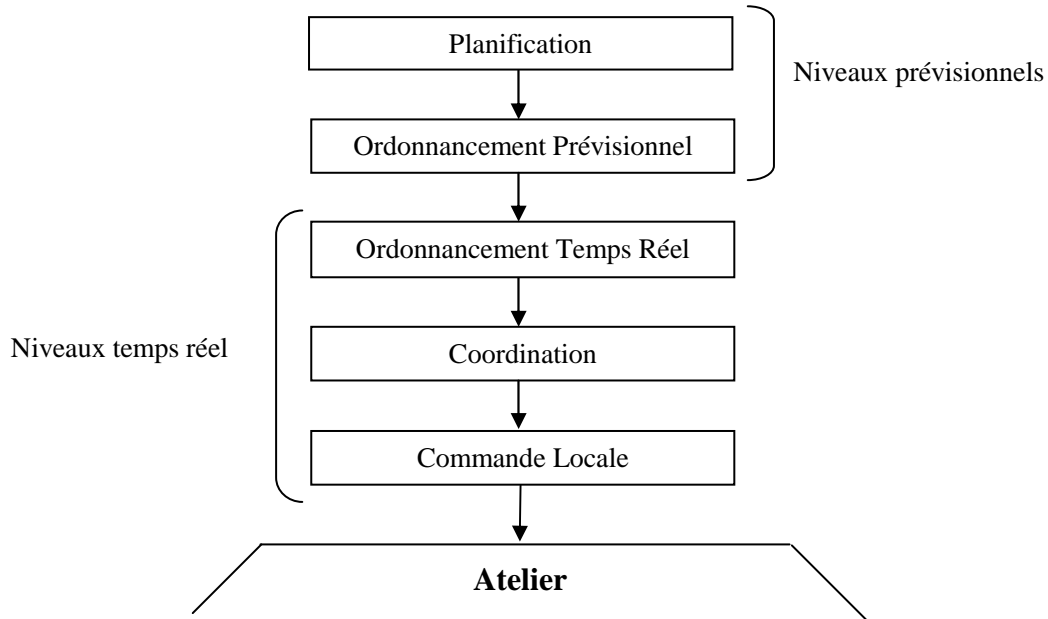


Fig. 1.2- Structure décisionnelle à cinq niveaux pour un atelier flexible

La Planification tente à dégager le plan de production (quantité, qualité, délais de production, pré-allocation des ressources) à partir des demandes et des exigences du marché et des performances des ressources de production.

Postérieurement, *l'Ordonnancement Prévisionnel* vise à déterminer les séquences de passage des produits à fabriquer sur l'ensemble des ressources de l'atelier tout en respectant les contraintes de capacité de ressources et les critères d'optimisation économiques et temporelles.

L'Ordonnancement Temps Réel assure la cohérence entre les décisions prises antérieurement (aux niveaux de *Planification* et d'*Ordonnancement Prévisionnel*) et les contraintes temps réel traduisant l'état réel de l'atelier.

La *Coordination* consiste à assurer une gestion cohérente des interactions entre l'ensemble des ressources de l'atelier et les contraintes déjà mentionnées.

Le dernier niveau de cette structure décisionnelle est *la Commande Locale* ; elle permet à mettre en alliance les actionneurs et les capteurs, d'une part, et les décisions concrétisées par le niveau précédent, d'autre part.

On peut ordonner ces niveaux en deux classes. La première a une spécificité prévisionnelle ; elle comprend la *Planification* et l'*Ordonnancement Prévisionnel*. La seconde présente un caractère temps réel ; elle conçoit les trois derniers niveaux.

1.2.3- Systèmes à Événements discrets

“Le mot discret ne signifie ni « temps discret », ni « état discret » mais réfère au fait que la dynamique est par des événements dont les dates d'occurrence n'ont pas une importance fondamentale. Seul compte réellement l'ordre de l'occurrence de ces événements.” [DAS 03b].

Les Systèmes à Événements Discrets (SED) peuvent se définir de façon négative par opposition aux systèmes classiques étudiés par la théorie du contrôle et dont l'évolution est continue et décrite par des équations différentielles. Ils apparaissent de façon naturelle dans la modélisation des systèmes informatiques, des réseaux de télécommunications, des réseaux de transport ou des Systèmes de Production (lignes d'assemblage, ateliers flexibles,...).

Partant des travaux antérieurs sur la conduite des systèmes de production, un SAP ou un SFPM peut être considéré comme un SED vu l'inutilité de la connaissance exacte de l'évolution des paramètres de l'état du procédé. Des valeurs particulières ou des positions atteintes sont suffisantes [AND 96]. Ces valeurs qui varient d'une manière brusque à des instants déterminés sont des variables d'état permettant la modélisation du système [CAS 93], [ZAM 97].

Pour un SP, le principe des SED est exploité pour accéder à :

- la mesure et/ou l'évaluation de la performance des systèmes
- la modélisation de la commande de la production en optimisant la fiabilité du système. Elle contribue à la permission ou au blocage d'une succession d'événements.

1.2.4- Terminologies de référence

Trois modules participent à la réussite d'un tel système, à savoir : la surveillance, la supervision et la commande. Dans ce qui suit, nous présentons les définitions de ces modules ainsi que quelques autres terminologies liées à ce domaine. Ces définitions sont prises de [LAP 92], [BAI 94], [PAT 99], [NAB 00], [DAS 03b], [DHO 05].

Commande : elle consiste à attribuer des séquences d'ordres aux actionneurs afin de gérer le processus de production. Ces ordres présentent les lois de commande. Elles recouvrent le nombre d'états possibles de système tels que les opérations relatives à la réalisation du produit, les opérations de réglage et d'essai effectuées lors de substitution de fabrication d'un produit, les actions exécutées pour réparer les dégradations apparues sur le système de production, les actions de protection des opérateurs et des biens, etc. Ces lois peuvent être mises à jour lors d'apparition des états inattendus.

Surveillance : elle permet la collection des informations provenant du système de la commande et des données du procédé pour déterminer l'état présent et/ou futur du système. Le rôle de la surveillance est limité au traitement des données et n'intervient pas directement sur les modèles ou sur le procédé.

Supervision : elle a le rôle de contrôle, de calcul et de mise à jour des paramètres du système de production, puis, si nécessaire, d'élévation des hésitations dans le système de commande. Lors des opérations normales, elle contrôle l'enchaînement de commande à exécuter en prenant compte de l'état du système de commande, de l'état du procédé et des contraintes imposées par l'environnement de système de production (ordonnancement temps-réel, optimisation, mise à jour de la commande, et remplacement d'une loi de commande par une autre, etc.). Si un symptôme de défaillance est apparu au niveau de commande ou de procédé, le rôle de ce module est de gérer la situation par la proposition d'un état de fonctionnement curatif en attendant la résolution de problème apparu (actions d'urgence, re-ordonnancement, opérations de recouvrement, etc.).

Autres terminologies :

La liste de termes ci-dessous suit un ordre alphabétique de présentation.

- *Défaillance* : interruption incessante de fonctionnement du système dans des conditions données.
- *Défaillance complète* : c'est une défaillance résultant des déviations d'une ou des caractéristiques. Ces déviations entraînent une disparition de la fonction requise.
- *Défaillance partielle* : c'est une défaillance résultant de déviations d'une ou des caractéristiques au-delà des limites spécifiées mais telles qu'elles n'entraînent pas une disparition complète de la fonction requise.
- *Défaillance progressive* : c'est une défaillance prévisible par un examen antérieur des caractéristiques d'un élément du procédé.
- *Défaillance soudaine* : c'est une défaillance imprévisible par un examen antérieur des caractéristiques d'un élément du procédé.
- *Défaut* : l'écart entre la valeur réelle et nominale d'un paramètre de système.
- *Disponibilité* : c'est l'aptitude d'une entité à être en état d'accomplir une fonction requise dans des conditions données, à un instant donné ou pendant un intervalle de temps donné, en supposant que la fourniture des moyens extérieurs nécessaires soit assurée. La disponibilité est donc basée sur deux concepts, l'un de disponibilité instantanée pour l'aptitude à un instant donné, et l'autre de disponibilité moyenne pour l'aptitude sur un intervalle de temps donné.
- *Erreur* : Incompatibilité d'une partie du modèle défini avec les paramètres et les spécifications de procédé. Une erreur est la conséquence d'une faute.
- *Erreur effective* : l'erreur devient effective à partir du moment où la partie erronée du système est utilisée.
- *Erreur latente* : une erreur est définie latente si une partie erronée du système n'a pas été utilisée.
- *État d'échec* : C'est une conséquence d'une défaillance. À partir duquel le composant du système ne fournit pas le service sollicité.
- *Faute* : une dérive des spécificités du système par rapport aux paramètres de fonctionnement normal.
- *Fiabilité* : c'est l'aptitude d'une entité à accomplir une fonction requise, dans des conditions données, pendant un intervalle de temps donné. La fiabilité traduit une notion de continuité de fonctionnement.
- *Perturbation* : un signal d'entrée non contrôlé dont l'existence est non préférable. Elle contredit une prévision faite aussi bien sur le fonctionnement du système opérant que sur l'évolution des objectifs élaborés à partir de la prévision de l'évolution de l'environnement. On distingue deux classes de perturbations (Fig. 1.3) : les perturbations externes et les perturbations internes.

- *Perturbation externe* : une perturbation qui affecte l'entrée du SAP ou du procédé. elle peut être produite suite à la non qualité de matière première, à l'instabilité des instants d'arrivée des produits, etc.
- *Perturbations interne* : une perturbation qui affecte l'état physique du procédé ou son fonctionnement. Elle peut être produite suite aux pannes des machines, aux modifications des durées opératoires, à la cession de ressources pour maintenance préventive, etc.

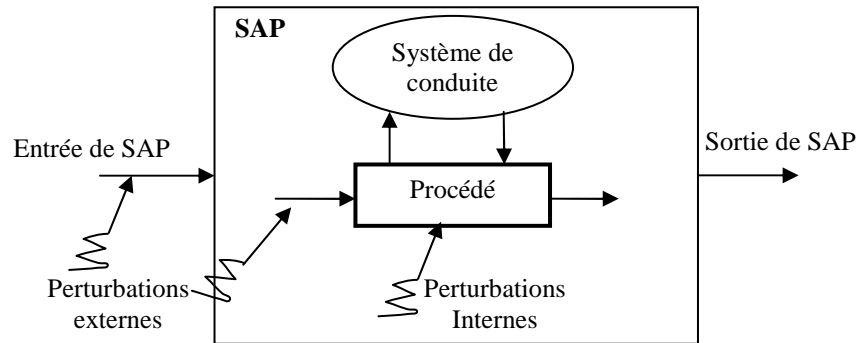


Fig. 1.3- Classes des perturbations

- *Proactivité* : c'est la capacité d'un système à prédire les changements d'état, à adapter ses règles de fonctionnement et à se réorganiser.
- *Réactivité* : c'est l'aptitude d'un SP à réagir dans un temps précis aux évolutions de son environnement interne ou externe par rapport au régime de fonctionnement. La réactivité est donc la vitesse de satisfaction aux demandes non anticipées.
- *Pilotage/Conduite* : c'est l'ensemble des actes réalisés afin de mener et guider le système de façon à garantir la convenance et la cohérence des différents composants du système en présence de perturbations.
- *Robustesse* : Pour un système manufacturier, la robustesse est son aptitude à conserver ses propriétés spécifiées face à des perturbations prévues ou imprévues. La robustesse est donc, la conséquence de deux éléments intrinsèques qui sont le type de variations, d'une part, et la définition des qualités nécessaires pour la sortie du système, d'autre part. Pour réagir aux perturbations, un système doit disposer de critères de décisions lui permettant de prendre en compte la notion de robustesse.
- *Robustesse active* : elle s'intervient lorsque les propriétés spécifiées peuvent être maintenues, mais au prix d'un calcul total ou partiel de la conduite.
- *Robustesse externe* : c'est la capacité d'un SP à maintenir ses propriétés spécifiées en présence des variations, intentionnelles ou non, à l'entrée du système (perturbations externes).
- *Robustesse interne* : c'est l'aptitude d'un SP à conserver ses propriétés spécifiées en présence des changements de valeur de paramètres du modèle du procédé (perturbations internes)
- *Robustesse passive* : elle répond au cas où aucune modification dans la conduite n'est

nécessaire pour que les propriétés spécifiées soient conservées en présence de variations

- *Sécurité* : c'est l'aptitude d'une entité à éviter de faire apparaître, dans des conditions données, des événements critiques ou catastrophiques. Ces événements peuvent être critiques pour l'opérateur, le système ou son environnement.
- *Symptôme* : une déviation élémentaire conduisant à un défaut susceptible de causer une dégradation d'un ou plusieurs éléments. C'est la seule information connue par le module de surveillance au niveau de la détection.

1.3- Commande des systèmes à événements discrets

1.3.1- Principe de Commande des SED

La commande des SED permet à amener le procédé d'un état initial donné à un autre final tout en passant par un ensemble d'états intermédiaires dans un ordre chronologique déjà défini. Pour atteindre chaque état, il faut assurer la satisfaction des consignes envoyées vers le procédé. Le principe de communication entre le système commande et le procédé est de type Appel/Réponse [ZAM 96], [DHO 05].

La partie commande transmet, en effet, un appel (requête) vers le procédé. La réalisation de cette requête coïncide à une évolution du procédé vers un état prévu par la commande. Une fois cet état est atteint, une réponse (compte rendu) envoyée vers la partie commande, affirme la réalisation entière de service demandé. Pour que la partie commande puisse tenir en compte de tels comptes rendus, elle doit, certainement, disposer d'une représentation des états de ce procédé. La connaissance de ces états lui permet de prévoir l'état suivant [ZAM 97], [DHO 05].

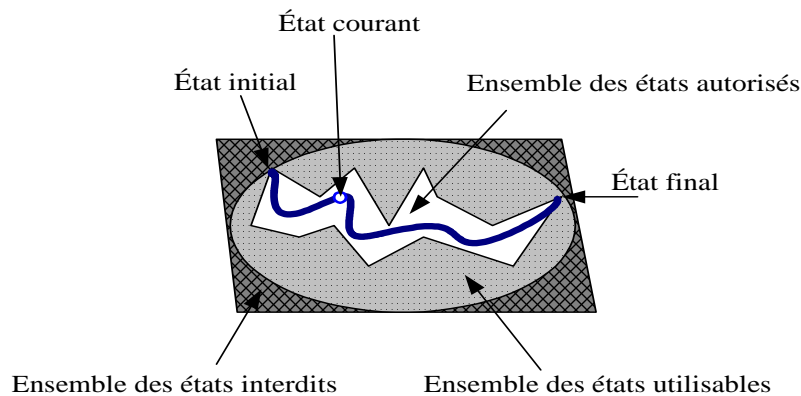


Fig. 1.4- États d'un système

En effet, on peut différencier les états suivants (Fig. 1.4) [COM 91], [ZAM 97] :

- États interdits : à partir desquels certaines lois de fonctionnement correct des composants risquent d'être violées,
- États utilisables : ce sont les états observables non interdits,

- États autorisés : ils appartiennent à un sous-ensemble des états utilisables. Ils s'occupent des contraintes imposant une séquence spécifique d'états tout en interdisant certaines évolutions du système,
- État initial du système, son état actuel et l'état final attendu.

1.3.2- Approches de Commande des SED

Les différentes approches proposées pour la commande des SED peuvent être classées en trois types [LY 98] : l'approche cyclique, l'approche réactive et l'approche dynamique.

L'approche cyclique [PRO 94], [CAM 97], [KOR 98] : il s'agit de définir hors ligne une commande prévisionnelle à flux de production maximale tout en présumant que :

- les pannes sont rares,
- les durées des opérations sont déterministes,
- l'écart entre la commande prévisionnelle et le déroulement réel de production est permis.

L'approche réactive [ROU 88], [ART 97] : elle vise à définir une commande réactive ajustable en cas de perturbations dues à une dérive ou dégradation de fonctionnement. L'objectif cible est de définir une séquence de groupes d'opérations commutables.

Cette approche sert à réagir aux perturbations d'une manière hiérarchique descendante. Elle vise donc à traiter au premier lieu les perturbations localement observées dans le groupement d'opérations en cours d'exécution sur la ressource perturbée. Si le traitement des perturbations au niveau local est impossible, on passe au niveau plus haut.

Cette approche permet de réagir aux perturbations de type panne produit (liées à un problème d'approvisionnement) ou de type flux.

L'approche dynamique [TAW 95] : elle a pour objectif de corriger, d'une manière dynamique, le comportement du procédé au plus tôt possible face aux perturbations. Cette approche est fondée sur le principe d'ordonnancement prévisionnel, effectué hors ligne, qui fixe les dates de début et de fin des opérations ainsi que les délais et les charges des machines.

Deux types de perturbations peuvent apparaître en cours de production :

- perturbation sur une ressource de transformation : dans ce cas, la réaction s'intervient par le biais d'une permutation locale des opérations, un échange de l'opération sur une autre machine ou par un réordonnancement
- perturbation sur le transfert : la réponse à ce type de perturbation, consiste à repérer le plus court chemin entre les deux ressources de transformation en tenant compte de la disponibilité des moyens de transport.

Cette approche est générale mais elle n'intègre pas de contraintes de qualité.

1.3.3- Gestion des perturbations et mesure de robustesse

1.3.3.1- Gestion des perturbations

Commander un système consiste à lui imposer un ensemble des lois de commande de façon à atteindre des objectifs définis au départ qui se coïncident aux besoins du marché.

Des perturbations peuvent procréer des déviations au niveau du modèle souhaité, au niveau de produit ou au niveau du procédé. Ceci doit être étudié par l'identification des perturbations (définition d'une base de données), l'élaboration des procédures de gestion de ces perturbations et finalement la mise à jour de la base de données au cas où une nouvelle perturbation est apparue.

Une perturbation peut-être due à trois types de conséquences [DHO 05a] :

- un *rebut* qui correspond à la production des pièces qui ne répondent pas aux spécifications et qui ne peuvent pas être récupérées,
- une *non-conformité* qui convient lorsqu'on peut corriger des pièces qui ne sont pas absolument conformes aux spécifications déjà définies,
- une *défaillance machine* qui correspond aux arrêts d'une machine pour réparation suite à une panne.

Dans le cadre de nos travaux, nous supposons que les racines (symptôme, défaut) d'une perturbation de type *défaillance machine* peuvent provoquer des perturbations *quantitatives* et *qualitatives* au niveau de produit.

Une perturbation *qualitative* peut-être de type *rebut* et *non-conformité* (Fig 1.5). Une perturbation quantitative est traduite par les performances temporelles du système de production tels que le temps de cycle, le taux de production, etc.

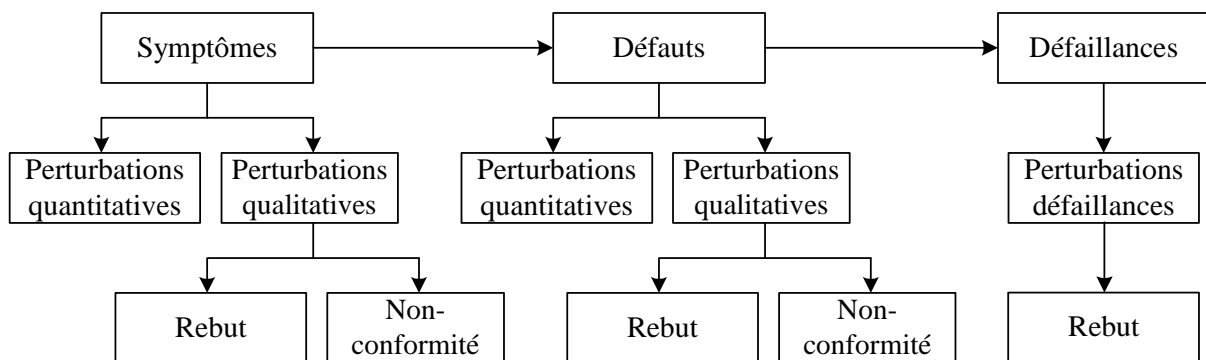


Fig. 1.5- Types des perturbations

Les stratégies de gestion des perturbations dans un système de production manufacturière sont nombreuses. La gestion des perturbations repose donc sur des systèmes robustes au niveau opérationnel (algorithmes, techniques d'ordonnancement...).

La majorité des stratégies mises en œuvre tend à définir des lois de commandes robustes en exploitant les marges sur les contraintes qualitatives et quantitatives ou à gérer les critères d'une manière floue. Les lois de commande robuste permettent la production des consignes sur les marges de conformités.

1.3.3.2- Mesure de robustesse

Vu que la robustesse d'un SP consiste à maintenir les spécificités de produit face aux perturbations, la mesure de la robustesse d'un SP se définit par rapport à deux paramètres principaux [COL 97] :

- le type de variations,
- la définition des qualités requises pour la sortie du système.

Le respect de la gamme est généralement le critère de sortie pour un système de production manufacturière. Divers critères comme le nombre d'en-cours, la régularité de sortie des produits ou le temps de séjour dans l'atelier peuvent enrichir la sortie du système. Pour chacun de ces différents objectifs, nous pouvons définir une commande donnée qui sera plus ou moins robuste par rapport à telle ou telle perturbation.

Lorsque les objectifs sont maintenus sans modification de la commande, on parlera d'une robustesse passive, par contre, la robustesse active traduit la capacité d'un système à assurer les performances au prix d'une modification en temps réel de la conduite. La détermination de cette robustesse fournit un critère de décision pour le calcul d'une nouvelle conduite dans le cas où la marge de robustesse passive est violée (Fig. 1.6).

Définition 1.1

Pour un système manufacturier, la robustesse est définie comme son aptitude à conserver ses propriétés spécifiées face à des perturbations prévues ou imprévues [COR 96].

Définition 1.2

La robustesse passive répond au cas où aucune modification dans la conduite n'est nécessaire pour que les propriétés spécifiées soient conservées en présence de variations [COR 96].

Définition 1.3

La robustesse active correspond au cas où les propriétés spécifiées peuvent être maintenues, mais au prix d'un calcul total ou partiel de la conduite [COR 96].

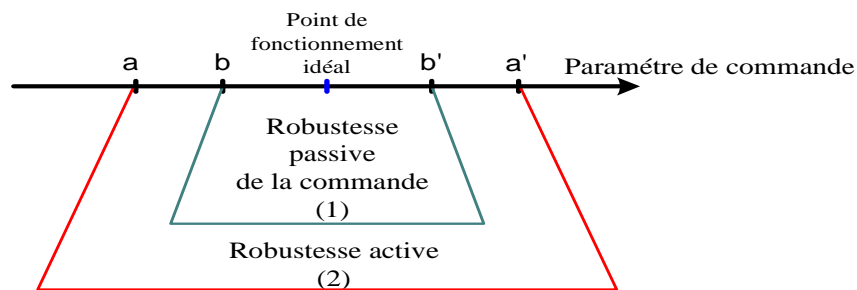


Fig. 1.6- Robustesses active et passive

Les marges $[a, a']$ et $[b, b']$ représentent les degrés de liberté du procédé physique. Chaque paramètre de commande de type quantitatif ou qualitatif sortant de ces limites entraîne une violation du cahier des charges.

(1) les propriétés spécifiées sont garanties sans aucun changement de la commande. Les valeurs a et a' correspondent à la robustesse passive.

(2) une commande quantitative ou qualitative doit être inventoriée; les marges dynamiques sont modifiées mais le séquencement demeure le même. Les valeurs b et b' correspondent à la robustesse active.

L'exploitation de la modélisation est adaptée comme une voie indispensable de recherche pour la détermination de la robustesse dans la conduite des systèmes de production.

L'évaluation de la robustesse a déjà été traitée précédemment en utilisant les Réseaux de Petri comme modèle :

- une preuve de robustesse externe a été apportée [COL 94]
- une évaluation de la robustesse passive inter-produits qui permet une commutation, d'un produit de type A vers un produit de type B [BON 00]
- une évaluation de la robustesse passive et active des systèmes de production à contraintes non temporelles face aux perturbations de type « dérives de qualité » [DHO 05a].
- ...

H. Hillion [HIL 89] a utilisé l'outil Graphes d'Événements (GE) en fonctionnement au plus tôt afin de répondre au critère « taux de production ».

1.3.4- Approches de commande robuste des SED

Les travaux de commande robuste des SED ont porté principalement sur deux problématiques :

- La robustesse vis-à-vis des contraintes de temps,
- La robustesse face à des perturbations non-temporelles.

1.3.4.1- Robustesse vis-à-vis des contraintes de temps

Elle englobe la plupart des travaux élaborés en matière de la commande robuste des SED. Les systèmes industriels qui ont ce type de contraintes sont particulièrement répandus. On cite l'industrie pharmaceutique, l'agroalimentaire, la galvanoplastie utilisée en métallurgie, etc. La majorité des travaux est développée au sein de LISTIC d'Annecy. Nous citons dans ce qui suit principalement les travaux de Collart Dutilleul [COL 94], [COL 97] et les travaux de Bonhomme [BON 00], [BON 01].

Dans ses travaux [COL 94], [COL 97], Collart Dutilleul a étendu une méthode de synthèse de commande robuste vis-à-vis des spécifications dans les ateliers qui ont des contraintes de temps de séjour corrélées à la qualité du produit. Une procédure de séparation et d'évaluation utilisant des contraintes est étendue.

Cette approche a traité la problématique d'optimisation des systèmes de production à contraintes de temps de séjour par l'étude de la dynamique du système afin de maintenir un taux de production défini au départ face aux perturbations externes acceptables, et ce, après le choix d'une référence spécifique dans la marge de la robustesse passive, afin de permettre une conservation de la périodicité de la sortie des pièces fabriquées en lissant les perturbations externes durant le cycle de production.

Les travaux de Bonhomme [BON 00], [BON 01] ont traité la problématique de *la robustesse passive inter-produits* afin d'optimiser le temps de séjour des divers produits fabriqués au même atelier. En effet, *la robustesse passive inter-produits* est l'ensemble des intervalles de fonctionnement admissibles autorisant une production selon les normes. Pour le cas de deux types de produits différents, de type *g* et de type *h*, *la robustesse passive inter-produits du*

produit de type *g* représente l'ensemble des périodes de fonctionnement du produit de type *g* permettant un échange de type de produit, et ceci à l'instant d'introduction d'une nouvelle pièce dans le système considéré.

1.3.4.2- La robustesse face à des contraintes non-temporelles

Les travaux en cette matière sont très rares. A notre connaissance les travaux de Dhoubi développés au sein de l'équipe « Systèmes à Evénements Discrets » de LAGIS de Lille sont les seuls qui ont traité cette problématique.

Hédi Dhoubi a étudié dans sa thèse la problématique de la robustesse et de la réactivité des systèmes de production à contraintes non temporelles face aux perturbations de type dérives de qualité [DHO 05a]. L'auteur suppose que les dérives de qualité sont dues à des variations de la qualité de la matière première.

Le travail de Dhoubi porte sur l'évaluation des deux types de robustesse :

- *la robustesse passive* face à des perturbations au niveau local d'un procédé de production suite à une dérive de qualité de matière première qui affecte le paramètre critique et significatif du produit fini, à savoir le poids de l'unité fabriquée.
- *la robustesse active* face aux dérives des paramètres critiques de qualité du produit, par le calcul du nombre de cycles requis pour passer d'un réglage à un autre sans violation des contraintes.

Dans le cadre de cette étude, seule la commande robuste des SED comportant des contraintes non temporelles est abordée. Les systèmes industriels contenant ce type de contraintes sont particulièrement répandus. En effet, le type de système étudié est un système de production manufacturière composé d'un ensemble de machines identiques à vitesses constantes et montées en parallèle. Dans ses travaux, Dhoubi a validé l'approche qu'il a développée à une manufacture de tabac. Il a traité le problème de régulation de la qualité en temps réel tout en considérant que l'atelier de confection est sans ressources limitantes. Vu que le poids de cigarette dépend essentiellement de la qualité de la matière première qui est préparée dans un atelier en amont, la commande cherchée doit donc garantir la robustesse du procédé et gérer chaque dérive de qualité de la matière première [DHO 05a].

Par ailleurs, ces fluctuations peuvent être équilibrées par des opérations de réglage effectuées soit sur le système d'alimentation, en agissant sur le débit, soit sur les dimensions géométriques du produit au moment de sa formation par le procédé. Ces paramètres, cités antérieurement, sont également contraints par des intervalles de conformité. Les violations de ces contraintes ont pour conséquence un rejet des quantités de produit [DHO 05b].

1.4- Surveillance des systèmes à événements discrets

Dans ce paragraphe, nous commençons par aborder et définir les fonctions élémentaires du système de surveillance ainsi que les différents types de surveillance. Puis nous présentons les différentes approches d'intégration ou de séparation de la surveillance à la commande. Nous terminons ce paragraphe par une présentation des différentes approches développées par des différentes équipes de recherche subordonnées à des laboratoires. Ces approches seront ordonnées selon les classes d'intégration ou de séparation de la surveillance à la commande.

1.4.1- Les fonctions de la surveillance

Partant des études antérieures, nous pouvons distinguer quatre fonctions essentielles de la surveillance.

La première fonction est la fonction *suivi*. Elle permet de recueillir les informations provenant du procédé pour assurer une image claire et fiable du système commandé au sein du système de surveillance [TAW 95], [ZAM 97], [NAB 00], [BOU 03], [DHO 05].

La deuxième fonction qui doit être garantie par un système de surveillance est la fonction *détection*. Elle consiste à détecter les déviations de comportement du système commandé par rapport à ce qui est prévu. Cette fonction doit assurer deux tâches principales. La première consiste à traiter les données en évaluant l'écart entre la valeur mesurée et le seuil critique défini. La deuxième permet de classer les symptômes ou les déviations [HOL 90], [TOG 92], [ZAM 97], [NAB 00].

Après la détection d'une déviation, il faut nécessairement rechercher, analyser et recommander l'origine, la nature et la cause de la déviation. Ce rôle est attribué à la fonction *diagnostic* [MAB 96], [ZAM 97], [NAB 00], [BOU 03].

La dernière fonction est le *pronostic*. Elle accède à l'analyse, l'estimation et la prévision des conséquences d'un symptôme ou d'un défaut sur le fonctionnement futur du système [NAB 00], [BOU 03]. Deux types de pronostic peuvent être distingués [BOU 01]. Le premier, appelé *pronostic préliminaire*, cherche les conséquences inévitables d'une défaillance (identification de l'ensemble des tâches qui ne peuvent plus être exécutées en respectant l'ordonnancement). Le deuxième, appelé *pronostic préventif*, est centré sur l'erreur latente et sur la propagation de défaillance (la répétition de la même défaillance). Ce type de pronostic est utilisé afin d'éviter de l'exécution d'activités menant à la détection du même symptôme.

Dans ce qui suit, nous citons d'autres fonctions qui ne font pas l'accord de tous les auteurs. Ces fonctions sont rattachées essentiellement à la supervision ou à la commande. Elles sont considérées comme des fonctions découlant de la surveillance dans un nombre important des travaux tels que les travaux de Combacau [COM 91], de Zamai [ZAM 97], de P. Berruet [BER 98], [BER 05], etc.

La *reprise* : lorsque le diagnostic est établi, il faut déterminer un ensemble de décisions qui mène le système à un état de fonctionnement acceptable. La reprise est un ensemble de procédures intégrées au système de commande fondé sur le choix des actions permettant de revenir dans le domaine de satisfaction des contraintes du modèle de bon fonctionnement [DHO 05].

La *décision* : elle détermine l'état qui doit être atteint afin de continuer l'opération normale, ensuite, elle détermine la séquence d'actions correctives à réaliser pour atteindre cet état [BOU 03].

Le *recouvrement* : c'est une fonction intervenant après l'occurrence d'une panne (détectée et diagnostiquée). Il détermine un nouvel état du système, nécessitant un ensemble ordonné d'actions correctives, qui vont modifier le comportement du procédé et de la commande, afin d'assurer la sécurité et la disponibilité du système [BER 98], [BER 00].

La **reconfiguration** : la reconfiguration des systèmes de production manufacturière est considérée comme une technologie d'ingénierie qui tend à assurer la réaction de ces systèmes face à des nouvelles conditions de fonctionnement contraignantes dues aux mutations rapides et hallucinantes du marché. Elle s'intervient pour agir sur l'état de procédé afin de changer l'état d'une ressource et sur le système de commande par le réglage de ses lois de la commande [BER 00], [DUP 99]. Trois classes de reconfiguration peuvent être définies :

- mineure, réadaptation des contraintes des lois de commandes,
- significative, d'autres ressources sont considérées pour la production,
- majeure, les ressources réallouées, en cas de panne par exemple, nécessitent une préparation pour exécuter la reconfiguration.

L'**urgence** : après la détection, si de la défaillance signifie la transgression d'une contrainte structurelle ou la mise en danger de l'opérateur, il faut immédiatement réagir en appliquant des procédures prédéfinies et prioritaires [BON 93].

1.4.2- Types de surveillance

D'après la littérature, principalement, Nous pouvons spécifier deux types de surveillance (Fig. 1.7) :

- **la surveillance de la commande** : elle a pour objectif de s'assurer que la commande est en convenance avec l'état du procédé. Ce type de surveillance est dessiné pour éviter la propagation de défaillance d'un composant vers d'autres composants du système. L'approche de S. El Khattabi [ELK 93], fondée sur les filtres de commande, est dédiée à ce type de surveillance. Ces filtres ont pour rôle la vérification de la compatibilité des actions émises avec l'état du procédé. Nous pouvons aussi mentionner le travail de E. J. Lee [LEE 06] qui a contribué à la commande des systèmes manufacturiers et notamment à leurs reconfigurations dynamiques. L'idée de base fondant cette approche consiste à synthétiser les contrôleurs de commande en tenant compte, d'une part, des besoins de l'utilisateur et d'autre part, de l'état du système. Ceci afin de reconfigurer le système pour qu'il puisse répondre à ces besoins.
- **la surveillance du procédé** : elle permet le suivi, la détection, le diagnostic et le pronostic et/ou le recouvrement des anomalies apparues au niveau du procédé. Cette surveillance peut-être **corrective** ou **préventive**.

L'objectif de la **surveillance corrective** est 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. Ce type de surveillance conçoit trois fonctions principales : détection, diagnostic et recouvrement. Concernant la **surveillance préventive**, elle aboutie à éviter l'occurrence des événements cessant le fonctionnement de procédé. 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.

D'autre part, la surveillance de procédé peut être **directe** en suivant directement les paramètres de ressources tels que la viscosité de l'huile, la température, la vibration, etc. Elle peut être aussi **indirecte** en s'appuyant sur les modèles fonctionnels

représentatifs du procédé tout en surveillant les ressources à partir des perturbations qu'elles s'engendrent sur les produits tels que le flux, les paramètres qualité, etc.

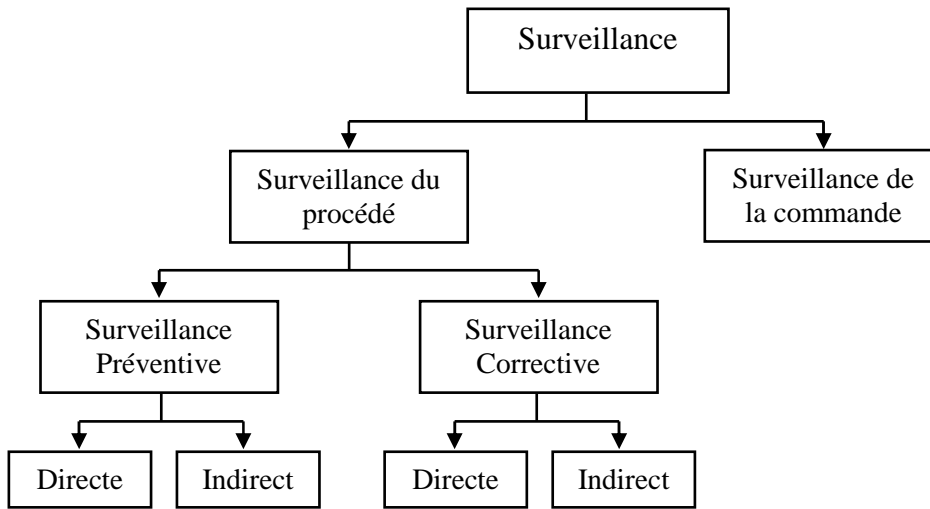


Fig. 1.7- Types de Surveillance

Dans le cadre de notre thèse, nous nous intéressons à la surveillance indirecte. Dans la littérature les travaux qui ont traité cette problématique sont nombrés. Concernant la surveillance corrective indirecte, nous pouvons mentionner les travaux de A.K.A. Toguyeni [TOG 91], [TOG 92], [TOG 96], [TOG 03], [TOG 05], de P. Berruet [BER 98], [BER 00], [BER 05], de M. Cambacau [CAM 91], de E. Minca [MIN 03], [MIN 04], etc. Ces travaux ont supposé que la défaillance d'une entité matérielle se traduit par la violation du temps accordé à l'accomplissement d'une opération.

Pour les travaux qui portent sur la surveillance indirecte préventive, nous pouvons citer les travaux de F. LY [LY 98], [LY 99], [LY 01], de L. Nabli [NAB 98], [NAB 00], [NAB 06], etc. Le principe est de détecter, de diagnostiquer une dérive de flux ou de qualité, et ensuite d'estimer l'impact de celle-ci sur le système grâce au pronostic.

1.4.2.1- Travaux de F. Ly [LY 98], [LY 99], [LY 01]

Ces travaux s'évaluent dans le cadre du projet CASPAIM (Conception Assistée de Système de Production Automatisée en Industrie Manufacturière). L'auteur contribue à une approche de Surveillance Prédictive Indirecte suivant des critères quantitatifs.

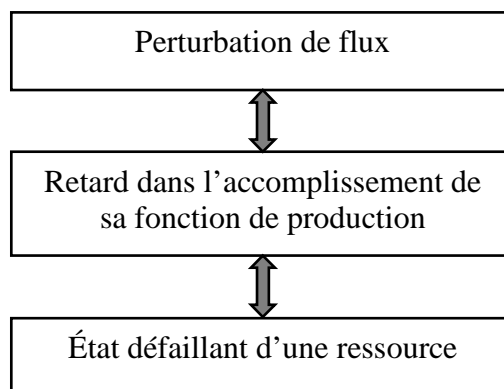


Fig. 1.8- Principe de Surveillance Prédictive Indirecte suivant des critères quantitatifs

Le principe de cette approche est de lier les perturbations de type « flux de produits » à l'état dégradé des équipements. (Fig. 1.8).

Cette démarche est adoptée afin d'optimiser la maintenance en lui offrant les moyens pour juger les services qu'elle attribue.

Le principe de cette approche est de détecter et de diagnostiquer une dérive de flux, afin d'estimer, ultérieurement, l'impact de celle-ci sur le système d'un point de vue quantitatif et temporel grâce au pronostic.

La détection des dégradations (dérives) est assurée par l'exploitation d'un modèle qui définit les taux de dérives acceptables sur les différents points mesurables du système. La connaissance des gammes opérationnelles et des ressources utilisées, permet de localiser la ressource dégradée et d'identifier l'amplitude et la fréquence de dégradation de symptôme ou de défaut (Fig. 1.9).

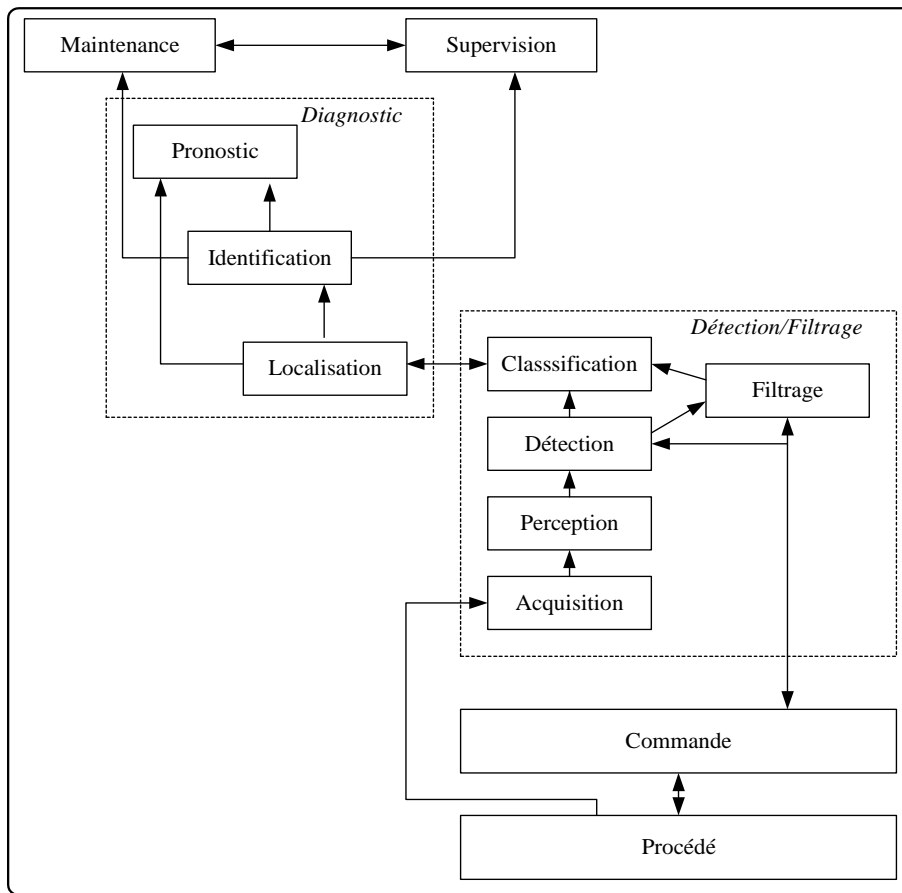


Fig. 1.9- Modèle de Surveillance Prédictive Indirecte [LY 99]

Les travaux développés ont montré qu'il est possible à partir des résultats de la Surveillance Prédictive Indirecte de mieux planifier la politique de production et de maintenance de l'entreprise.

1.4.2.2- Approche de L. Nabli [NAB 98], [NAB 00], [NAB 06]

Le but de ce travail était de développer un outil de Surveillance Préventive Conditionnelle Indirecte (SPCI) pour les Systèmes Automatisés de Production tout en exploitant les principes de normalisation.

L'approche développée par Nabli consiste au suivi de l'évolution, la détection des dérives au niveau de la qualité du produit, l'identification de l'origine de la dérive et finalement l'analyse de la cause de cette dérive [NAB 98], [NAB 00], [NAB 06].

Une hypothèse est prise : l'état défaillant d'une ressource entraîne une baisse de qualité du produit, ceci mène à une baisse de ses performances, d'où un retard dans l'accomplissement de sa fonction de production par rapport à ce qui lui avait été défini au départ (Fig. 1.10).

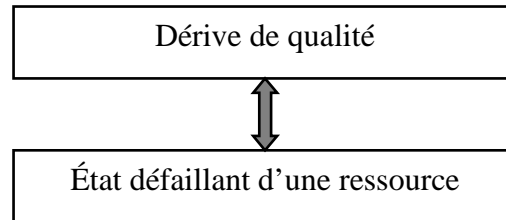


Fig. 1.10- Principe de SPCI suivant des critères qualitatifs

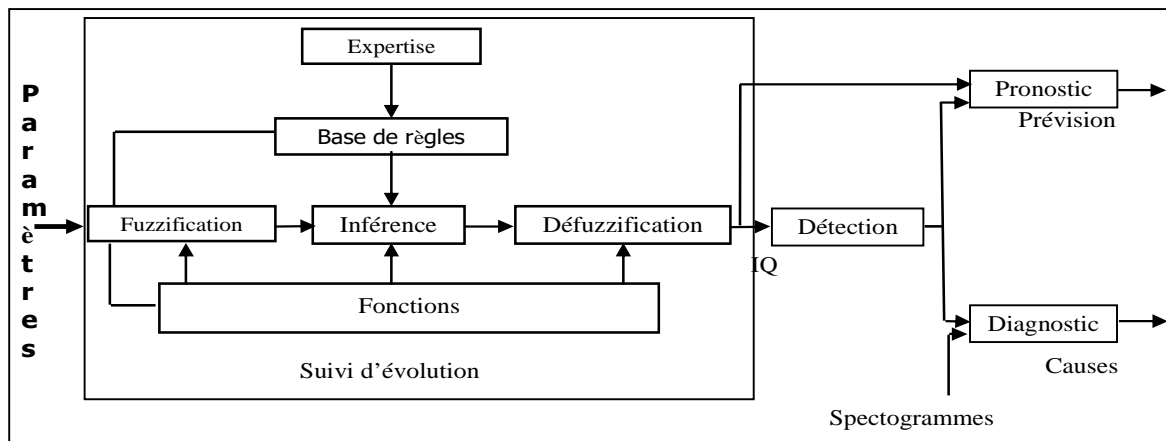


Fig. 1.11- Modèle de Surveillance Préventive Prévisionnelle Indirecte

Après une analyse fonctionnelle du système de production l'auteur a identifié les principaux paramètres de qualité aux différentes étapes de production. A partir de ces paramètres, il a défini des indicateurs de qualité qui ont été comparés à des seuils critiques, afin de détecter les dégradations de la qualité du produit.

L'approche qualitative proposée permet de détecter les dérives de qualité et d'estimer leurs impacts sur le système grâce au pronostic (Fig. 1.11).

Ce type de surveillance, développé par Nabli, est basé sur l'analyse de la tendance des écarts types des variations de la moyenne des valeurs mesurées sur les paramètres considérés. Par ailleurs, il a supposé que les classes de défauts sont prédéterminées.

1.5- Intégration ou séparation de la surveillance à la commande

Jusqu'à maintenant, nous avons présenté, dans un premier lieu, les concepts de base sur la commande ainsi que les approches de commande des SED. L'étude qui en découle traite la problématique de robustesse de la commande face à des contraintes temporelles ou non-temporelles. Ultérieurement, nous nous sommes intéressés à la surveillance. Nous avons développé le concept de base de la surveillance : définitions, fonctions, différents types et

classification.

Dans ce que suit, nous allons accéder à la problématique d'intégration ou de séparation de la surveillance à la commande. Incontestablement, la surveillance présente un palliatif à la commande. Selon la littérature, ce palliatif peut être présenté sous trois approches distinctes [ZAM 97] :

- fusion de la surveillance au système de commande,
- séparation de la surveillance de la commande,
- entre la séparation et la fusion.

1.5.1- Fusion de la surveillance au système de commande

Dans ce type d'approche, les comportements normaux et anormaux de système sont supposés connus à l'avance. Donc, il reste à traiter ces comportements par rapport à la commande afin à garantir une gestion efficiente qui permet un fonctionnement adéquat quelque soit l'état du système.

Le traitement associé à chaque évolution normale ou anormale doit être spécifique à la nature du comportement observé et élaboré hors ligne. L'efficience de cette approche est proportionnellement inévitable à la vraisemblance des évolutions normales ou anormales définies au départ par rapport à la réalité du système. Ceci ne garantit jamais une couverture intégrale des situations qui conduisent à un dysfonctionnement du système.

Dans la littérature, les travaux qui se sont intéressés à ce type d'approche sont nombrés. Nous mentionnons :

- les travaux de A.K.A. Toguyeni [TOG 91], [TOG 92], [TOG 96], [TOG 03], [TOG 05],
- les travaux de P. Berruet [BER 98], [BER 00], [BER 05] et de R. Kumar [KUM 96]
- les travaux de O. Kazushi [Kaz 01],
- les Travaux de l'équipe Génie Industriel et Logiciel (GIL) du Laboratoire d'Automatique et de Mécanique Industrielles et Humaines de Valenciennes (LAMIH) [PAR 92] [MAB 96],
- les travaux du Laboratoire Universitaire de Recherche en Production Automatisée (LURPA) en collaboration avec le LAGIS [HAM 09].

1.5.1.1- Approche de travaux d'A. Toguyeni [TOG 92]

Les travaux de Toguyeni sont élaborés au sein de LAGIS de Lille. Dans cette approche proposée pour surveiller en ligne des systèmes flexibles de l'industrie manufacturière, l'auteur s'intéresse particulièrement à la surveillance en ligne intégrée à la commande des ateliers flexibles [TOG 92]. Un outil de surveillance curative est élaboré par l'intermédiaire d'une modélisation fonctionnelle du procédé. Le module de surveillance est constitué principalement de trois fonctions : **détection**, **diagnostic** et **recouvrement** (Fig.1.12).

La **détection** est fondue sur l'hypothèse que la défaillance d'une entité matérielle se traduit par la violation du temps accordé à l'accomplissement d'une opération. Un modèle temporel est déterminé. Il permet de définir la relation d'un événement et son compte-rendu. En effet,

un symptôme est déclenché lorsque l'occurrence de compte-rendu est provoquée ou lorsque l'occurrence de compte-rendu ne s'est pas produite dans l'intervalle de validité.

Le **diagnostic** est assuré en deux étapes : la localisation et l'identification. La localisation a le rôle de limiter l'origine du défaut à une partie du système. Elle est assurée par l'interprétation des Signatures Temporelles Causales (STC) qui se coïncident à des conséquences de symptômes ordonnés dans le temps et séparés par des intervalles quantifiables. L'identification est résolue par un graphe d'hypothèses propageant le raisonnement.

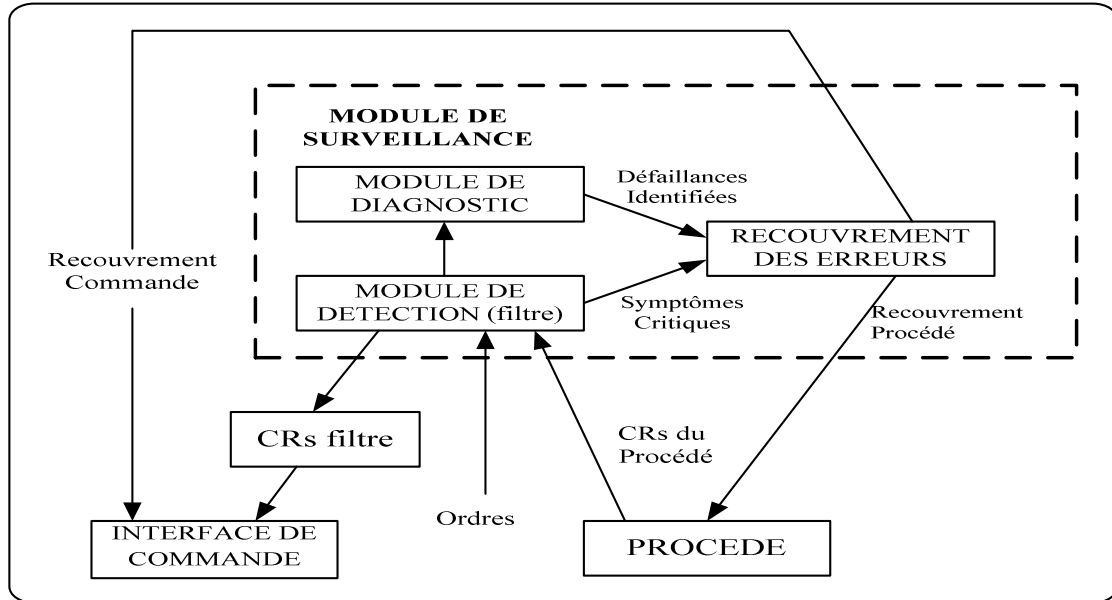


Fig. 1.12- Structuration de la fonction surveillance et positionnement par rapport à la PC et au procédé [TOG 92]

Finalement, pour la mise en œuvre de la fonction **recouvrement**, Toguyeni a proposé une méthode de classification des procédures de recouvrement. Ces propositions visent à adapter la commande et le procédé à une gestion implicite des modes de marches dégradés du système.

1.5.1.2- Approche de Travaux de P. Berruet [BER 98], [BER 00], [BER 05]

- *Travaux réalisés au sein de LAGIS [BER 98], [BER 00]*

Les travaux de Berruet s'occupent au recouvrement des pannes matérielles dans les systèmes flexibles de production manufacturière [BER 98], [BER 00]. L'objectif est de conserver une haute disponibilité du système par le maintien de ses particularités face à des perturbations de type «panne/défaillance».

Ceci est achevé par la recherche d'un nouvel état à travers un ensemble d'actions correctives modifiant le comportement du procédé et de la commande de manière à assurer la sécurité du système et le maintien d'un taux de production acceptable.

La détermination des éléments critiques du système par l'intermédiaire d'un modèle de graphe GAO (Graphe d'Accessibilité Opérationnel) permet l'examen des conséquences d'une défaillance sur les fonctions du SFPM.

La fonction recouvrement peut-être due à une défaillance critique. Dans ce cas, son rôle est d'éviter la propagation d'où l'assurance d'un minimum de perte. Elle peut-être aussi due à

une défaillance significative, son rôle devient la recherche d'une nouvelle organisation du système de production en vue d'une reprise à l'aide d'actions correctives tout en passant par les deux étapes suivantes :

- analyse du système en vue d'étudier sa capacité de continuer à produire,
- détermination des chemins encore accessibles pour le choix d'une solution optimale selon différents critères (fidélité maximale à l'ordonnancement initial).

La nouvelle configuration du système doit tenir compte du produit en cours de transformation dans le SFPM au moment de l'apparition de la panne. Selon la stratégie de recouvrement considérée, ce produit peut être : évacué, traité sur la même ressource, poursuit sa gamme plus tard ou être rejeté.

○ *Travaux réalisés au sein de LESTER [BER 05]*

Au sein de l'équipe de LESTER, P. Berruet a poursuivi ces travaux de recherche sur la Surveillance-Commande des systèmes de production. Il a proposé une approche pour la conception et l'évaluation d'architecture de systèmes reconfigurables [BER 05]. Cette approche peut-être appliquée aux systèmes manufacturiers qu'aux systèmes électroniques. Une modélisation des composants du système à étudier est introduite par cette approche [BER05]. L'objectif de cette modélisation est d'analyser la structure de l'architecture matérielle, d'une part, et d'évaluer ses capacités de tolérance aux fautes et de reconfiguration, d'autre part.

1.5.1.3- Approche de Travaux de O. Kazushi et G. S. Kang de l'université de Michigan (USA) [KAZ 01]

L'approche de Kazushi [KAZ 01] s'occupe des systèmes de production reconfigurable. Elle s'occupe de la problématique d'optimisation, de modification et de réutilisation des composants de systèmes de production. L'objectif est l'amélioration de l'environnement de technologie de la conception de commande de système.

Une méthode efficace de conception des systèmes de commande pour la reconfiguration est élaborée. Cette méthode consiste à construire, dans une première étape, de point de vue concepteur, le modèle de données de SCD (System Control Design) reconfigurable et de réduire, dans une seconde étape, les descriptions superflues d'état. Le modèle de données de SCD comprend trois sous modèles qui correspondent à l'écoulement de processus, le chemin d'objet et le comportement du système. Pour chacun d'eux un STD (State Transition Diagrams) est hiérarchiquement structuré. Ce travail se concentre sur la façon de procéder pour changer le type, le nombre et l'endroit de ressources.

Un tel modèle est fondé sur les fonctionnalités de base suivantes (Fig. 1.13) :

- définition de données des points de vue du concepteur,
- structuration hiérarchique des modèles.

Les principaux éléments dans la conception de commande du système de production sont :

- écoulement de processus de fabrication,
- choix des ressources,
- objet chemin / écoulement entre les ressources,

- comportements (séquencement du programme de commande),
- actions de collaboration et de synchronisation des ressources,
- disposition des ressources.

Le modèle se compose de fonctionnalités suivantes :

- squelette de structure hiérarchique des données de SCD,
- description paramétrique du comportement de processus,
- liste des variables et des paramètres.

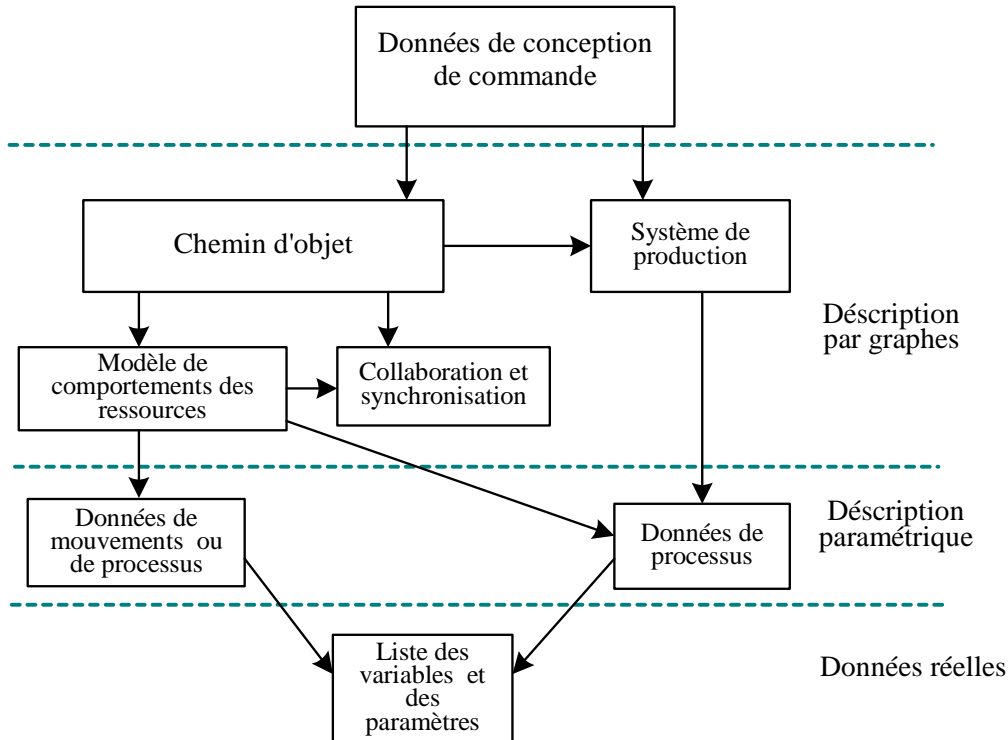


Fig. 1.13- Modèle de conception des systèmes de commande pour la reconfiguration

1.5.1.4- Travaux de l'équipe Génie Industriel et Logiciel (GIL) du Laboratoire d'Automatique et de Mécanique Industrielles et Humaines de Valenciennes (LAMIH) [PAR 92] [MAB 96]

Parayre [PAR 92] a fondé son approche sur la description des modes d'exploitation explicités à partir du Modèle d'Exploitation des Systèmes Automatisés de Production (MESAP) pour aider l'opérateur dans la phase de reconfiguration de système. Le modèle développé s'entend d'une décomposition structurelle accédant à certains niveaux de séparation tels que le niveau flux de pièces, le niveau composant et les niveaux application.

Le processus d'élaboration de la reconfiguration se réalise selon deux niveaux (Fig. 1.14) :

- Niveau stratégique : il s'intéresse aux décisions relatives au réglage des modes de fonctionnement des ressources et à la modification en termes de flux de produits.
- Niveau tactique : il s'occupe des états intermédiaires pour la transition de l'état actuel et l'état objectif.

La reconfiguration repose sur le principe d'identifier un ensemble de changements (scénarii de reconfiguration) au niveau comportement dynamique, d'une part, et au niveau des modes de d'exploitation des entités du SAP, d'autre part.

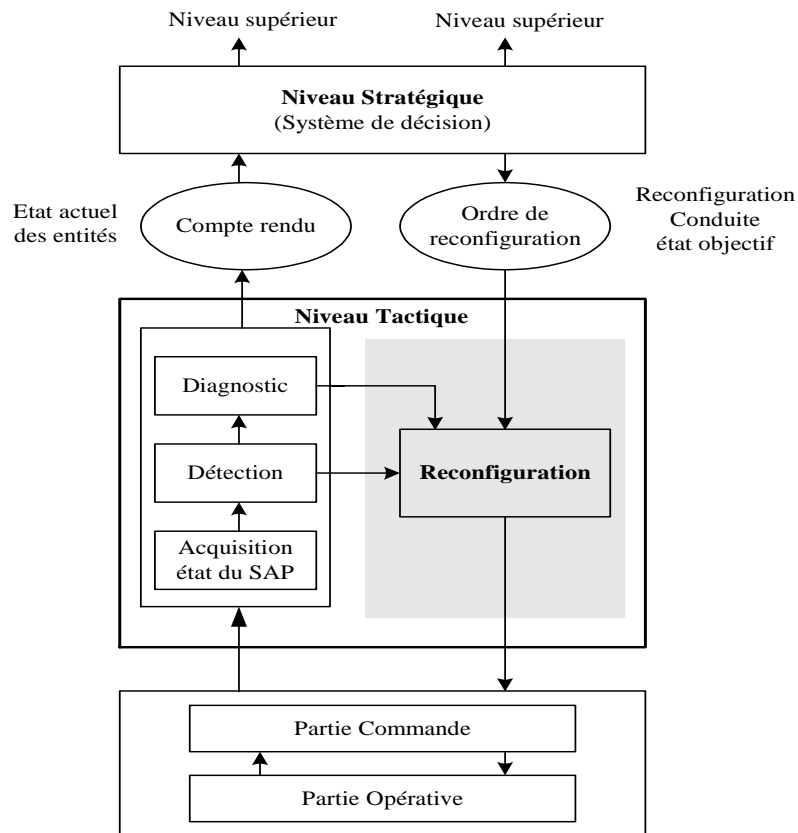


Fig. 1.14- Hiérarchie de processus de réalisation de la configuration

Une procédure proposée par M. Mabrouk [MAB 96] permet de déterminer les états transitoires avant le passage de l'état initial à l'état objectif. Cette procédure est décomposée en trois étapes :

- spécification des états de modes de fonctionnement,
- spécification des contraintes entre ces états,
- spécification des scénarii de reconfiguration.

1.5.1.5- Travaux du Laboratoire Universitaire de Recherche en Production Automatisée (LURPA) [HAM 09]

Le laboratoire Universitaire de Recherche en Production Automatisée (LURPA) en collaboration avec le LAGIS ont fondé une méthode de modélisation dédiée à la gestion des modes de marche des Systèmes Automatisés de Production tout en garantissant leurs compatibilité et cohérence [HAM 09]. L'approche est réactive et synchrone. Elle est basée sur une modélisation fonctionnelle des états d'une machine effective.

Cette méthode consiste à assurer à la fois le rôle informationnel et le rôle opérationnel dans le système de Supervision-Commande. Les informations fournies sur la fonction de l'état du

système et des sous-systèmes, permettent la mise à jour des modèles du système de commande [HAM 09].

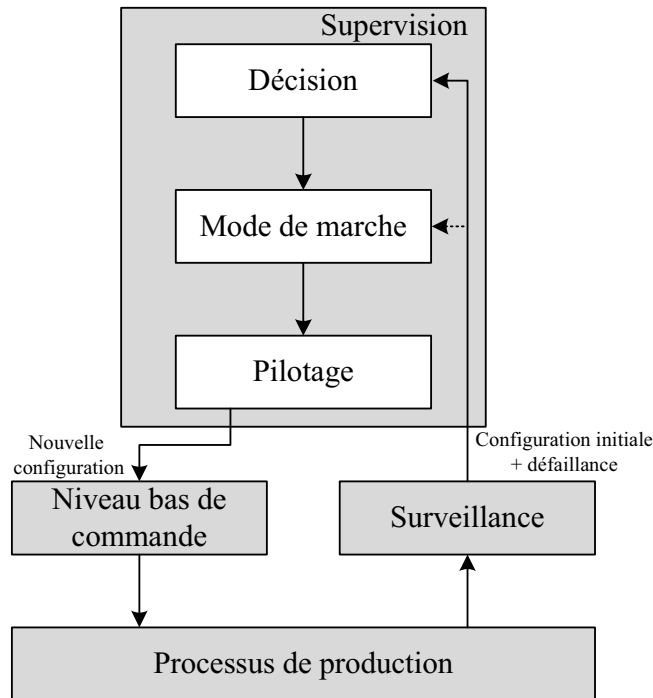


Fig. 1.15- Modèle de gestion des modes de marche des Systèmes Automatisés

Les fonctions informationnelles, décisionnelles et de pilotage de système Supervision-Commande s'occupent de quatre fonctions principales : surveillance, décision, pilotage et la mode de marche (Fig. 1.15). La surveillance sert à détecter et localiser les échecs de système opérant et de commande. La décision prévoit la nouvelle fonction de configuration du système. Et finalement, le pilotage et le mode de marche permettent la détermination des décisions au sujet de la nouvelle configuration.

En effet, en ce qui concerne son rôle opérationnel dans le cadre de commande, la mode de marche avec le pilotage met en œuvre certaines actions sur le niveau bas de commande et sur le système opérationnel. Le mode choisi et les objectifs de production sont assurés par le changement de mode de traitement. Les actions au niveau bas de commande sont effectuées par les fonctions mode de marche et le pilotage.

Les interactions de la fonction mode de marche avec toutes les autres fonctions du système de Supervision-Commande sont nécessaires.

1.5.2- Séparation de la surveillance de la commande

Cette approche, consiste à séparer la surveillance de la commande. Dans ce type d'approche, la commande est destinée à gérer le système lors de fonctionnement normal du système. Parallèlement, la surveillance fonctionne afin de suivre l'état de procédé, de détecter les symptômes (situations anormales) qui gênent ou tendent à gêner le déroulement normal du système et finalement de résoudre les problèmes repérés.

L'avantage de cette approche s'aperçoit par la possibilité d'attribuer des mécanismes différents pour la commande et pour la surveillance. Par contre, la séparation de fonctionnement de la surveillance et de la commande entraîne sûrement des désaccords d'actions sur le procédé. Si un symptôme, un défaut, ou une défaillance sont détectés, la surveillance peut exiger des actions préventives ou correctives, selon l'approche adoptée, qui ne seront pas en adéquation avec la propriété de la commande.

Dans ce qui suit, nous présentons quelques travaux qui ont séparé la surveillance de la commande. Nous considérons les approches qui s'occupent uniquement à la surveillance ou à la commande comme des travaux qui se relient à la séparation.

Parmi les travaux qui ont traité cette problématique, nous mentionnons les travaux (présentés dans le paragraphe 1.4.2) de F. Ly [LY 98], [LY 99], [LY 01], de L. Nabli [NAB 98],[NAB 00], [NAB 06],etc. pour la surveillance et les travaux (présentés dans le paragraphe 1.3.4) de Dhouibi [DHO 05], [DHO 05a], [DHO 05b], de Collart Dutilleul [COL 94], [COL 97], [COL 03] et de Bonhomme [BON 00], [BON 01], etc. pour la commande.

1.5.3- Approche mixte

Cette approche présente une solution face aux inconvénients de deux autres approches. La solution consiste à intégrer le suivi, la détection et la reprise à la commande et à séparer les autres fonctions.

Dans ce cas, il n'est pas forcément exigé de prédire l'ensemble des évolutions anormales. Il suffit de prendre en évidence les états normaux du système. Elles sont définies anormales, seulement, les évolutions qui ne peuvent être traités par la commande. Pour cette approche, la commande est étendue par l'intégration implicite de la fonction suivi, la fonction détection et la fonction reprise. Les autres fonctions de la surveillance tels que le diagnostic, le pronostic le recouvrement, etc. restent, quant à elles, séparées de la commande. Le rôle de ces derniers est de traiter les situations jugées anormales afin de proposer des solutions envisageables à être admises par la fonction reprise intégrée au module de commande. L'objectif est de revenir dans le domaine de satisfaction des contraintes du modèle de bon fonctionnement.

Dans la littérature, les travaux qui se sont intéressés à ce type d'approche sont nombreux. Parmi ceux qui ont traité cette problématique, nous mentionnons les travaux de LAAS : [CAM 91], [ZAM 97], etc., de LAI : [NOU 97] ,[NOU 04] et de LAB : [MIN 03], [MIN 04] ainsi que le travail de [BUY 09],etc.

1.5.3.1- Approches de travaux de LAAS

Les travaux développés au LAAS s'intéressent essentiellement aux architectures de Surveillance-Commande de type hiérarchique et distribué pour les Systèmes Flexibles de Production Manufacturière (SFPM).

M. Combacau, dans le cadre de ses travaux de thèse [CAM 91], a proposé une architecture de Surveillance-Commande hiérarchique. Cette dernière, présente une approche pour intégrer les fonctions : suivi et détection dans les systèmes de commande automatique, en vue de l'appliquer particulièrement aux SFPM.

L'approche proposée, était initialement structurée en blocs suivants (Fig.1.16) :

- commande : il s'agit d'un modèle qui présente l'ensemble des séquences opératoires à réaliser. En fonctionnement normal, ce modèle présente le seul bloc

qui communique avec les autres modules de l'hierarchie de Surveillance-Commande,

- référence : il est présenté par un modèle qui définit l'ensemble des états utilisables du système de production. Il permet l'assurance de fiabilité des paramètres de système par le contrôle de chaque signal envoyé par le procédé et de chaque ordre déclaré par la partie commande,
- diagnostic : il s'agit du bloc qui permet l'identification des causes si les symptômes ont été reconnus. Il est basé sur des règles d'inférences,
- décision : il permet la mise en œuvre de la fonction décision par la modification de commande, l'activation des procédures d'urgence, le déclenchement de l'exécution de la reprise et finalement la décision concernant le transfert du traitement de défaillance vers le niveau supérieur,
- procédures d'urgence : ce bloc permet à la décision la possibilité d'agir directement sur le procédé en lui attribuant des enchaînements d'actions prédéterminées et prioritaires.
- interface opérateur.

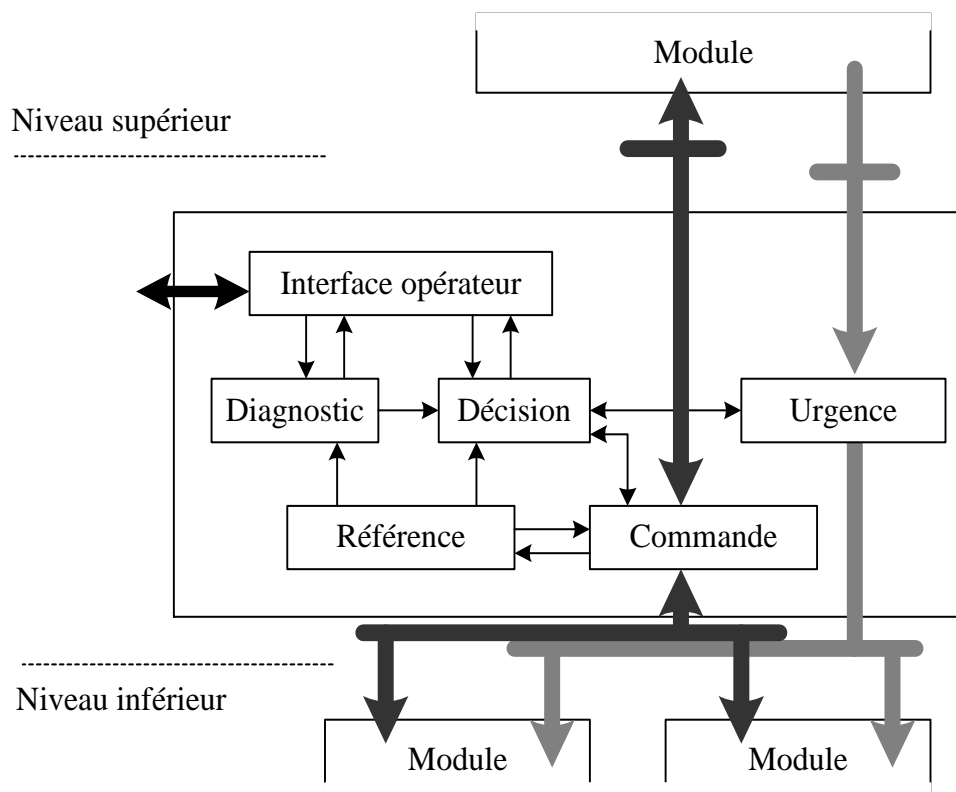


Fig.1.16- Architecture du Surveillance-Commande hiérarchique

Les fonctions suivi et détection sont assurées par la coopération et la synchronisation existantes entre le modèle de référence et celui de commande. Ces derniers évoluent alors en parallèle. En fonctionnement normal avant chaque requête, la commande consulte le module de référence afin d'en examiner la convenance vis-à-vis des contraintes du procédé. Après la

transmission d'un ordre, elle se met alors en attente d'un compte-rendu d'exécution. En cas de réception convenable (vis-à-vis d'une fenêtre temporelle) de ce compte-rendu, l'information est envoyée vers le module de référence pour la mise à jour de l'image du procédé. Si un symptôme est détecté seul le modèle de référence est mis à jour.

Les travaux de A. Chaillet-Subias [CHA95] et E. Zamai [ZAM97], du LAAS, accèdent à traiter le système l'information à mettre en œuvre afin d'assurer l'intégration des différentes fonctions dans un contexte hiérarchisé. Vu que la réalité des industries implique que les ressources (hard, soft, humains, etc.) sont géographiquement séparées et que la bonne progression du processus dépend de l'évolution synchronisée de ces ressources. M. DA SILVEIRA [DAS 03b] a étendu cette approche dans une structure distribuée.

1.5.3.2- Approches de travaux de LAI

Ces travaux sont basés sur la théorie des automates. Ils se sont introduits pour faire basculer un fonctionnement nominal à un fonctionnement dégradé, face aux perturbations.

L'approche développée consiste à implanter un module de surveillance à la commande par supervision. Le module surveillance a le rôle de prendre en charge la reconnaissance et la gestion des situations jugées anormales [NOU 97]. Il s'agit de détecter des dérives comportementales du procédé et ensuite, de compenser les défaillances apparues afin de garder le système en état de fonctionnement. Tout au long de la durée de compensation, le système fonctionne en mode dégradé. La durée de mode de fonctionnement dégradé entame dès l'apparition de la dérive jusqu'à la phase de reprise.

Le module de surveillance additionné comprend deux blocs (Figure 1.17) : un bloc observateur qui suit l'évolution du système et détecte les déviations comportementales ; et un bloc compensateur permettant la réception des informations à travers l'observateur et la sureté de fonctionnement du système en mode dégradé.

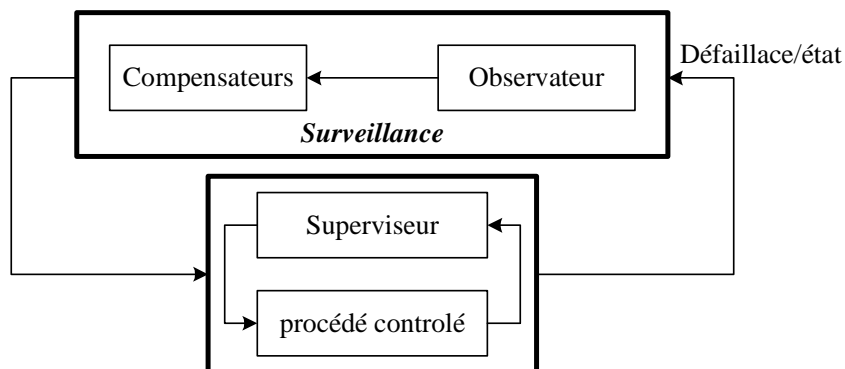


Fig.1.17- Surveillance de commande supervisée

Dans [NOU 04], une application de cette théorie à un système automatisé de production manufacturière est présentée. Trois étapes sont suivies afin d'implanter cette théorie : la modélisation de module de commande et de comportement sous la forme d'automates ; la synthèse automatique des modules non-confliktuels superviseurs en utilisant un outil informatique ; et finalement, l'implémentation du modèle sur le plan pratique.

1.5.3.3- Approches de travaux de LAB

L'architecture développée par LAAS ainsi que ses mécanismes de détection ont été exploités postérieurement par E. Minca [MIN 03], [MIN 04], etc. dans le cadre de développement de l'ingénierie de conduite Systèmes Automatisés de Production. Dans ses travaux Minca a amélioré la fonction diagnostic par l'implantation d'un modèle permettant de l'identification des causes des défaillances. L'outil de réseaux de Petri flous est exploité.

1.5.3.4- Travail de Buyurgan de l'Université d'Arkansas [BUY 08]

La méthode de processus d'analyse hiérarchique est appliquée au problème d'ordonnancement en temps réel et à la reprise des systèmes de production manufacturière.

La structure de l'approche proposée se compose de deux modèles : un modèle système, qui représente le comportement du système par le biais d'états et des événements, et d'un modèle ordonnanceur qui permet le contrôle du modèle système. Le modèle système consiste à une modélisation des états et des événements par utilisation de l'outil des machines à états finis. Les principales tâches du modèle ordonnanceur sont le marquage et le contrôle du système modèle. Le modèle d'ordonnancement dynamique est basé sur l'optimisation des mesures de la performance désirée. Les sorties futures de l'état de système sont déterminées à partir d'une étude d'une chaîne de caractères qui représente le comportement passé.

Les modèles processus d'analyse Hiérarchique sont dynamiquement mis au point pour classer les mesures de performance ainsi que les activités du système.

1.6- Notre contribution : vers une Surveillance Indirecte suite à une Commande Robuste

L'étude bibliographique étendue, dans les paragraphes précédents, nous a montré les différentes architectures proposées. La problématique d'intégration ou de séparation de la surveillance à la commande est présentée. Par rapport aux travaux présentés précédemment, l'approche que nous optons à développer porte particulièrement sur le maintien de la robustesse de commande des systèmes de production manufacturière par la surveillance des paramètres de la loi de commande. En effet, partant des travaux qui portent sur la séparation de la surveillance à la commande, notre objectif, dans un premier lieu, est de développer un modèle commande qui s'appuie sur la robustesse de produit (d'où indirectement du procédé) de point de vue qualité et flux. Dans un deuxième lieu le modèle surveillance optera à déterminer les événements qui vont cesser probablement le fonctionnement normal du système.

Notre architecture de Commande-Surveillance est développée sous des contraintes de minimisation du recours à la maintenance corrective. Ceci est afin d'éviter des interruptions provoquant des événements fatales influant la performance globale de l'entreprise au niveau technique et/ou économique. Il s'agit donc d'une surveillance indirecte préventive suite à une commande robuste des systèmes. En effet, la surveillance des paramètres de la loi de commande robuste permet de suivre momentanément les déviations dans un contexte de robustesse active et de préjuger les événements fatales qui peuvent influencer la robustesse du système.

Nous supposons qu'un symptôme de défaillance d'une ressource entraîne une baisse des indicateurs de qualité et/ou du produit. Ceci mène à une baisse de ses performances. Nous supposons aussi que la qualité de la matière première influe d'une manière implicite la

l'indicateur de qualité de produit fini ou semi-fini.

Nous abordons notre architecture en deux blocs principaux : bloc Commande et bloc Surveillance. L'architecture proposée est présentée par la figure 1.18.

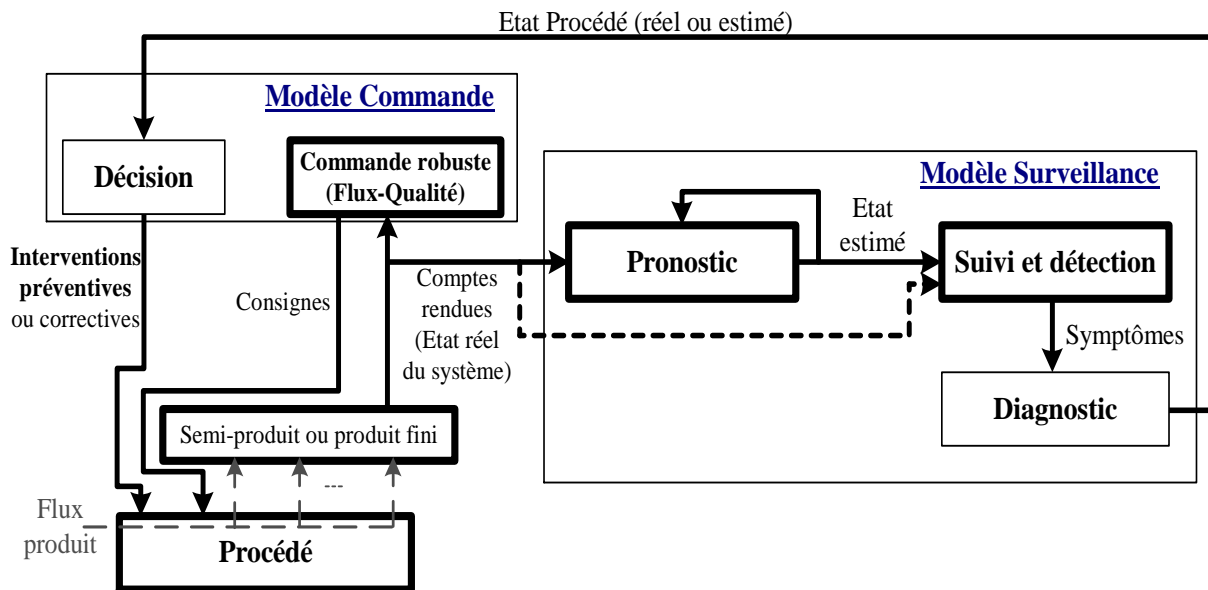


Fig.1.18- Surveillance Indirecte suite à une Commande Robuste

Le principe de fonctionnement de cette architecture consiste à gérer les informations du système d'une façon intelligente et collective qui permet la satisfaction des besoins informationnels de toutes les entités constituant le système Surveillance-Commande ainsi que ses auxiliaires (procédé et produit). Ce qu'on a appelé « auxiliaire » du système Surveillance-Commande présente en soit le sujet de notre problématique.

Donc, il s'agit de bien étudier les spécificités du système et les obligations de l'environnement afin d'implanter une commande robuste s'appuyant sur des lois permettant la gestion des contraintes. Une fois cette commande est implantée, les autres blocs et fonctions peuvent être créés et localisés. Dans ce qui suit, nous allons définir, dans une première étape, le rôle de chaque entité de ce système Commande-Surveillance. Ensuite, nous présentons une explication de fonctionnement et d'échange informationnel entre les différentes entités. Nous terminons par aborder les fonctions dont on va s'intéresser.

1.6.1- Bloc Surveillance

Le bloc Surveillance a le rôle de prévenir les états jugés anormaux du système. Ce bloc est constitué de trois fonctions qui sont : Pronostic, Suivi-Détection et Diagnostic. La fonction pronostic permet de prédire les valeurs futures des variables de la commande. La fonction suivi-détection permet principalement le suivi des états estimés par la fonction pronostic et la détection des déviations. Aussi, cette fonction à un autre rôle secondaire consistant à suivre les états réels de système. Le suivi et la détection des états réels et des états estimés sont mis en redondance active. Le but est de renforcer la fiabilité de l'architecture Surveillance-Commande. La redondance permet de couvrir les cas où le prédicateur de la fonction pronostic a donné une estimation non juste, vu les bruits de mesures par exemple.

Finalement, la fonction diagnostic a le rôle de rechercher, analyser et recommander l'origine, la nature et la cause de la déviation.

1.6.2- Bloc Commande

Le bloc Commande permet de définir les paramètres représentatifs du système de conduite. Ce bloc est constitué de deux fonctions principales : la fonction commande robuste et la fonction décision. Concernant la commande robuste, il s'agit de modéliser les modes de fonctionnement d'une manière qui permet de définir les contraintes mises sur les produits tels que le séquençement de flux et les marges de robustesse traduisant les intervalles autorisés de la performance désirée (qualité et flux). La modélisation, des contraintes misent sur ces paramètres, nous permet de contrôler l'interaction entre les incertitudes des paramètres représentatifs du système. La commande robuste, qu'on veut implanter, prend en considération, au même temps, la modélisation des marges passives et actives de la robustesse. La modélisation des marges de robustesse active et passive au même temps sur le même modèle nous permet de définir des seuils de détection des symptômes. Ceci est inspiré de la définition de la robustesse passive et celle de la robustesse active. Dans ce travail, nous prenons comme seuils de détection des symptômes les bornes inférieure et supérieure de la marge de la robustesse passive de chaque paramètre de la loi de commande. En ce qui concerne la fonction décision, son rôle primordial est de déterminer les interventions préventives nécessaires qui devront être effectuées sur le procédé afin de ne pas cesser le fonctionnement normal de procédé. Dans le cas échéant, si un symptôme est détecté suite au suivi des états réels de système, c'est à dire l'estimateur de la fonction pronostic n'a pas prédit correctement l'état futur de système de fonctionnement, le rôle de la fonction décision est de déterminer les interventions correctives à réaliser.

1.6.3- Explication d'échanges informationnels

Une fois l'architecture est implantée, les informations circulent de la façon suivante : des consignes de commande seront envoyées du bloc Commande à travers la fonction commande robuste vers le procédé, ces consignes traduisent les lois de commande. Des informations « comptes rendus », expliquant l'état du procédé, extraites du produit seront transmises vers la fonction commande robuste du bloc Commande ainsi que les fonctions pronostic et suivi-détection du bloc Surveillance. Ces comptes rendus permettent aussi la synchronisation en temps réel entre l'état du procédé et la partie de la fonction suivi-détection qui assure le suivi des états réels de système.

Ces informations sont relatives aux paramètres des lois de commande modélisées. La fonction Pronostic assure le traitement de ces informations à travers un prédicteur qui permet d'estimer l'état futur de ces paramètres ; bien évidemment du système. Autre que ces états estimés, les contraintes de robustesse seront aussi l'entrée de la fonction suivi-détection. S'il y a des symptômes détectés par le prédicteur, les informations relatives seront envoyées vers la fonction diagnostic. A cette étape, une requête des informations sera envoyée vers la fonction

décision qui a le rôle de déterminer et d'assurer, si nécessaire, les interventions préventives et/ou correctives à réaliser sur le procédé.

1.6.4- Les fonctions auxquelles on va s'intéresser

Notre architecture de Commande-Surveillance consiste à ce que le système conserve ses propriétés face aux perturbations externes et internes tout en minimisant les recours à la maintenance corrective dans le but de conserver la robustesse de système. Comme il est annoncé précédemment, dans ce travail de thèse, notre objectif principal est de maintenir la robustesse passive des systèmes. Ceci nous conduit à s'intéresser aux fonctions de l'architecture Surveillance-Commande qui touchent la conservation des spécificités de système. Ces spécificités sont normalement maintenues et assurées par le biais de trois fonctions principales qui sont : la fonction commande robuste du bloc Commande ainsi que les fonctions pronostic et suivi-détection du bloc Surveillance. Les autres fonctions à savoir : décision et diagnostic, ne présentent pas en elles mêmes un catalyseur de base pour la réussite de notre architecture. Nous pouvons utiliser n'importe quel modèle ou méthode pour ces deux fonctions. Dans la suite de ce travail de thèse, nous nous focalisons notre point d'intérêt sur :

- La fonction Commande Robuste du bloc Commande : cette fonction présente le centre de nos intérêts. Par rapport aux études écoulées, une contribution sera présentée au niveau de la modélisation des lois de commande multicritères qualité-flux.
- La fonction pronostic du bloc Surveillance : cette fonction présente aussi un élément très important de notre architecture puisqu'elle produit des estimations qui vont être la base, postérieurement, de tous les jugements et décisions permettant la conservation de la robustesse passive. Nous allons opter au développement d'un prédicteur efficace résultant une estimation efficace qui permet ensuite de bien analyser les états futurs à travers les fonctions suivi-détection et diagnostic.
- La fonction suivi-détection du bloc Surveillance : cette fonction a le rôle de bien suivre et détecter les symptômes de type qualité et flux. A partir des travaux présentés dans la littérature, un modèle de suivi-détection sera développé afin de suivre et détecter les symptômes de types flux et qualité.

1.7- Conclusion

Dans ce chapitre, nous avons présenté la problématique de conduite des Systèmes Automatisés de Production (SAP), notamment, les Systèmes Flexibles de Production Manufacturière (SFPM) qui ont une propension aux Systèmes à Événements Discrets (SED).

En effet, nous avons réalisé une étude bibliographique générale dont nous avons présenté la terminologie nécessaire pour comprendre les notions des systèmes de conduite des systèmes de production. Les méthodes, les architectures et les emplacements des modules surveillance et commande au sein de système de conduite sont étudiés. Dans la dernière partie de ce chapitre, nous avons présenté notre architecture qui présente une des contributions de cette thèse. En effet, nous avons bien détaillé le principe de fonctionnement ainsi que les fonctions de cette architecture. Nous avons aussi mentionné que les fonctions dont on va s'intéresser sont le Suivi-Détection, le Pronostic et la Commande Robuste.

Notre approche a explicitement des centres d'intérêt proche de ceux présentés précédemment. Par rapport à ces approches nous intégrons la robustesse comme notion à rôle bi-spatial : pour la commande et la surveillance.

Par le biais de l'étude bibliographique réalisée, nous avons montré que notre sujet de thèse est pertinent. En effet pour la commande, les approches de conception déjà publiées n'ont pas traité à la fois la robustesse de système de production face à des contraintes temporelles et non-temporelles. D'autre part, aucun travail, à notre connaissance, ne s'est appuyé sur la surveillance des paramètres d'une loi de commande comme un outil de maintien de la robustesse.

Dans le chapitre suivant, nous allons présenter une étude bibliographique étendue sur les outils et les méthodes destinés pour : la modélisation des commandes robustes, le pronostic ainsi que le suivi et la détection des symptômes des défaillances.

Chapitre 2
Méthodes et outils pour la
Commande et la Surveillance des
 systèmes industriels

Chapitre 2

Méthodes et outils pour la Commande et la Surveillance des systèmes industriels

Table des matières du chapitre 2

2.1-	Introduction	41
2.2-	Modélisation de la robustesse des ateliers à contraintes temporelles et non temporelles	41
2.2.1-	Automates à états finis	42
2.2.2-	Réseaux de Petri	43
2.2.2.1-	RdP P-temporels	46
2.2.2.1.1-	Définition formelle.....	46
2.2.2.1.2-	Règles de fonctionnement.....	47
2.2.2.1.3-	Définition fondée sur l'âge des marques.....	47
2.2.2.1.4-	Définition fondée sur l'intervalle de temps.....	48
2.2.2.1.5-	Condition de franchissement d'une transition.....	48
2.2.2.1.5.1-	États d'un jeton.....	48
2.2.2.1.5.2-	Calcul de l'état suivant	49
2.2.2.1.6-	RdP P-temporels commandés	50
2.2.2.1.7-	Exemple d'application des RdP P-temporel.....	51
2.2.2.2-	RdP à Intervalles.....	52
2.2.2.2.1-	Définition formelle.....	52
2.2.2.2.2-	Règles de fonctionnement.....	53
2.2.2.2.3-	Définition d'un état	54
2.2.2.2.4-	Calcul de l'état suivant	54
2.2.2.2.5-	Les RdP P-temporels : une sous classe des RdP à intervalles [COL 03]	55
2.2.2.2.6-	Exemple d'application des RdP à Intervalles.....	56
2.2.2.3.1-	RdP t-temporels	57
2.2.2.3.1.1-	Définition formelle	57
2.2.2.3.1.2-	Etats et règle de tir.....	57
2.2.2.3.1.3-	Exemple d'application des RdP t-temporels.....	58
2.2.2.3.2-	RdP temporels à flux indépendants.....	59
2.2.2.3.2.1-	Définition formelle	59
2.2.2.3.2.2-	Règles de franchissement	59
2.2.2.3.2.3-	Exemple d'application des RdP temporels à flux indépendants.....	60
2.2.3-	Synthèse et choix des outils de modélisation adéquats pour notre approche	60
2.3-	Outils et méthode pour la surveillance des systèmes industriels	61
2.3.1-	Méthodes et outils de surveillance avec modèle	63
2.3.1.1-	Surveillance par modélisation fonctionnelle.....	63
2.3.1.1.1-	AMDEC	63
2.3.1.1.2-	Arbres de Défaillances.....	63
2.3.1.2-	Méthodes de surveillance par modélisation physique.....	64
2.3.1.2.1-	Redondances physiques et analytiques	64
2.3.1.2.1.1-	Redondances physiques.....	64
2.3.1.2.1.2-	Redondances analytiques.....	64
2.3.1.2.2-	Estimation paramétrique	65
2.3.2-	Méthodes et outils de surveillance sans modèle.....	66
2.3.2.1-	Surveillance avec outils statistiques.....	67
2.3.2.1.1-	Test de franchissement de seuils	67
2.3.2.1.2-	Test de moyenne et test de variance.....	67
2.3.2.2-	Surveillance avec outils symbolique.....	67
2.3.2.2.1-	Surveillance par modélisation symbolique	67

2.3.2.2.2-	Surveillance par modélisation causale	69
2.3.2.2.3-	Surveillance par reconnaissance des formes	69
2.3.2.2.3.1-	Reconnaissances des formes par utilisation des outils statistiques	70
2.3.2.2.3.2-	Reconnaissances des formes par utilisation de la logique floue	70
2.3.2.2.3.3-	Reconnaissances des formes par utilisation des réseaux de neurones	71
2.3.2.2.3.3.1-	Eléments de base des réseaux de neurones	71
2.3.2.2.3.3.2-	Apprentissage des Réseaux de neurones artificiels	73
2.3.2.2.3.3.3-	Architectures neuronales les plus utilisées en surveillance	74
2.3.2.2.3.3.3.1-	Perceptron Multi Couches (PMC)	74
2.3.2.2.3.3.3.2-	Réseau à Fonctions de base Radiales (RFR).....	75
2.3.2.2.3.3.4-	Réseaux de neurones temporels	77
2.3.3-	Synthèse et choix des outils de surveillance adéquats pour notre approche.....	77
2.4-	Conclusion.....	78

2.1- Introduction

Dans un contexte industriel, le choix des outils et des modèles permettant la réussite de l'implantation d'une architecture Surveillance-Commande nécessite la connaissance des moyens technologiques et des techniques d'analyse adéquates. Les objectifs de l'architecture Surveillance-Commande ainsi que l'environnement interne et externe du milieu industriel jouent un rôle très important pour la détermination des outils que nous devons choisir pour réussir une telle architecture.

Après avoir présenté dans le premier chapitre une étude bibliographique en ce qui concerne notre problématique tout en situant notre architecture Surveillance-Commande qui présente la contribution générale de cette thèse, nous passons donc dans ce chapitre à présenter les différents outils et méthodes cités dans la littérature, pour la commande et la surveillance des Systèmes Flexibles de Production Manufacturière.

Ce chapitre sera organisé comme suit :

Nous nous intéresserons, dans un premier lieu, aux outils permettant la modélisation des lois de commande robuste. La prise en considération des contraintes et des objectifs de l'architecture « Surveillance Indirecte Suite à une Commande Robuste » que nous avons proposé à la fin de chapitre précédant, sera un facteur important pour choisir l'outil que nous allons utiliser dans la suite pour modéliser la fonction commande robuste du bloc commande. Ensuite, nous présenterons les différents outils et méthodes de surveillance des systèmes industriels utilisés et cités dans la littérature. Nous optons à travers cette étude à choisir les outils qui s'adaptent avec notre architecture, surtout les fonctions suivi-détection et pronostic du bloc surveillance.

2.2- Modélisation de la robustesse des ateliers à contraintes temporelles et non temporelles

L'exploitation de la modélisation est adaptée comme une voie indispensable de recherche pour l'étude et la détermination de la robustesse des systèmes de production. Un modèle peut exploiter et expliquer au mieux les potentialités et les performances d'un système à événements discrets. Pour atteindre un objectif donné moyennant une approximation et une vue partielle plus ou moins abstraite de la réalité, un modèle peut être développé. Un modèle est donc une présentation personnelle du concepteur ou d'analyste. Il est établi à partir des objectifs, des instances ainsi que de la nature et de la qualité des informations dont dispose le créateur. Il peut être exprimé par des outils mathématiques, des symboles et des schémas. Les travaux de H.Camus [CAM 97] présentent une description et une comparaison des différents outils de modélisation utilisés pour l'évaluation des performances des systèmes à événements discrets.

Un de nos objectifs, dans cette thèse, est de gérer et de déterminer la robustesse des systèmes à contraintes temporelles et non temporelles. Même si que nous nous intéressons seulement à des systèmes de type flow-shop, des synchronisations permettant la gestion des états des paramètres du produit sont nécessaires.

Les Automates à Etats Finis et les Réseaux de Petri (RdP) présentent un champ d'application appréciable, pour la modélisation et l'étude des systèmes à événements discrets. Vu la

performance de l'outil des Réseaux de Petri en termes de modélisation des synchronisations, des parallélismes, des conflits et des partages de ressources, cet outil est considéré comme la voie la plus importante de recherche pour la modélisation des systèmes à événements discrets. Affirmant cette vue, dans la littérature, l'outil le plus fréquemment utilisé pour modéliser la robustesse des systèmes de production à contraintes temporelles est les Réseaux de Petri P-temporels [KHA 97], [BON 00], [JER 06]. Néanmoins, cet outil paraît inapte à modéliser le problème de robustesse des systèmes de production à contraintes non temporelles.

Les RdP à intervalles développés par Collart [COL 03] présentent un complément aux réseaux de Petri P-temporels. Ils permettent la modélisation des contraintes sur un paramètre quelconque dans un processus manufacturier. En effet, cet outil de modélisation porte au domaine de recherche des apports et des innovations considérables. Jusqu'à présent, Il est exploité dans un contexte de modélisation pour la détermination et l'évaluation de la robustesse [COL 04], [DHO 05a].

Dans la suite de cette section de ce chapitre, nous allons présenter les principes et les champs d'application des Automates à Etats Finis et des Réseaux de Petri. Nous terminerons par choisir l'outil le plus adéquat à résoudre notre problématique et atteindre nos objectifs touchant l'étude et l'analyse de la robustesse.

2.2.1- Automates à états finis

Automate à états finis ou machine à états finis (en anglais, finite state automaton ou finite state machine (FSA, FSM)) est un automate abstrait qui sert à modéliser des états finis d'un système donné. Un automate est constitué d'états et de transitions. Un automate fini possède un nombre fini d'états distincts et un nombre borné des règles de transitions entre états : il ne dispose donc que d'une mémoire bornée.

Une machine à états finis est une représentation graphique étiquetée. Les cercles présentent les états et les flèches orientées simulent les transitions. Une transition est étiquetée par une expression. Chaque état porte son propre nom et il est étiqueté par ses actions d'entrée et de sortie. L'état initial est repéré par une flèche spéciale en forme de L.

Définition 2.1 [ALU 94]

Formellement, un Automate Fini Déterministe (DFA) est un quintuplet : (S, Σ, T, s, A)

- un alphabet (Σ)
- un ensemble fini d'états (S)
- une fonction de transition ($T : S \times \Sigma \rightarrow S$).
- un état de départ ($s \in S$)
- un ensemble d'états acceptant ($A \subseteq S$)

Alur et Dill [ALU 94] ont introduit les automates temporisés (ATs). Ce modèle est une amplification des performances des automates d'états finis avec du temps continu permettant de spécifier des systèmes temps-réel. *Les automates temporisés* sont utilisés pour modéliser des systèmes qui combinent des évolutions *discrètes* et *continues*.

Définition 2.2 [ALU 94]

Formellement, un Automate temporisé (DFA) est un 6-uplet : (S, Σ, T, s, C, A)

- un alphabet (Σ) ,
- un ensemble fini d'états (S) ,
- une fonction de transition $(T : S \times \Sigma \rightarrow S)$,
- un état de départ $(s \in S)$,
- un ensemble fini d'horloges C ,
- un ensemble d'états acceptant $(A \subseteq S)$.

La construction d'un modèle décrivant un système à événements discrets à l'aide d'une machine à états, nécessite la détermination exacte des états dans lesquels peut se trouver le système étudié. Vu les spécificités rigides Automate à états finis :

- les événements sont tous ordonnés en temps linéaire,
- les évolutions simultanées des processus,
- l'indépendance entre des horloges locales,
- les événements ne sont reliés que par des relations d'ordre partiel.

Le risque de blocage mortel du à l'incohérence dans les communications peut survenir.

Dans nos travaux, nous voulons spécifier le comportement du système en présence du mécanisme de synchronisation permettant la discrimination entre la robustesse passive et active. Pour cette raison, nous nous intéressons dans cette thèse à l'utilisation des RdP qui offrent, explicitement, par la nature de ses graphiques ce mécanisme.

2.2.2- Réseaux de Petri

Les réseaux de Petri (RdP) sont introduits en 1962 par Carl Adam Petri¹. Depuis leur apparition, ils présentaient un outil puissant de modélisation des systèmes à événements discrets [MUR 89], [SIL 96]. Une bonne modélisation de la structure des systèmes par les réseaux de Petri permet une analyse rigoureuse de leurs propriétés [DHO 05a]:

- **les propriétés dynamiques intéressantes du système** : bornitude, absence de blocage, stabilité, réversibilité, ...,
- **la représentation du processus** : les ressources partagées, les priorités intervenantes, les gammes de fabrication, les stratégies d'ordonnancement...
- **les objectifs de production** : ils correspondent aux coûts de fonctionnement et à la rentabilité en vue d'optimiser : ressources minimales nécessaires, taux de production, production au plus tard pour limiter les stocks de pièces, périodicité du système,...

Pour analyser un système modélisé par les RdP, on se recourt à :

- l'application de méthodes d'analyses, en se basant sur une définition mathématique qui donne l'espace d'état,
- la simulation en utilisant des simulateurs appropriés.

¹ Carl Adam Petri (né le 12 juillet 1926 à Leipzig) est un mathématicien allemand et un informaticien. Petri a inventé les réseaux de Petri en 1962 dans une partie de sa thèse de doctorat : *Kommunikation mit Automaten* (Communication par les automates) à l'Université de Bonn.

Définition 2.3 [VAL 01]

Un RdP non marqué est un graphe biparti constitué de places représentées par des cercles, de transitions représentées par des barres et d'arcs représentés par des flèches orientées qui relient les transitions aux places et vice-versa. Il est représenté par un quadruplet $R = \langle P, T, Pré, Post \rangle$ où :

- P est un ensemble fini de places,
- T est un ensemble fini de transitions,
- $Pré : P \times T \rightarrow IN$ est l'application places précédentes,
- $Post : P \times T \rightarrow IN$ est l'application places suivantes

C est la matrice d'incidence du réseau de Petri avec :

$$C = Post - Pré \quad (2.1)$$

Définition 2.4 [VAL 01]

Un réseau de Petri marqué est le couple :

$$N = \langle R, M \rangle \quad (2.2)$$

Où :

- R est un réseau de Petri
- M est une application de marquage

$$M : P \rightarrow \mathbb{N}$$

Chaque place p du réseau est caractérisée, à un instant donné, par le nombre de marques (jetons) qu'elle comprend $M(p)$. Le marquage d'un réseau de Petri est une opération qui consiste à assigner des jetons dans les places.

- Un réseau de Petri R est pur si et seulement si:

$$\forall p \in P \text{ et } \forall t \in T : Pré(p, t).Post(p, t) = 0 \quad (2.3)$$

Un graphe du réseau ne doit comprendre aucune boucle élémentaire. Une boucle élémentaire possède la même place aussi bien pour l'entrée que pour la sortie.

- Une transition t est franchissable (sensibilisée) si et seulement si :

$$\forall p \in P, M(p) \geq Pré(p, t) \quad (2.4)$$

Si t est franchissable pour le marquage M , le franchissement (tir, crossing) de t donne le nouveau marquage M' tel que :

$$\forall p \in P \quad M'(p) = M(p) - Pré(p, t) + Post(p, t) \quad (2.5)$$

Le tir d'une transition consiste à retirer les jetons des places d'entrée pour les mettre dans les places de sortie.

La dimension du vecteur marquage M est égale au nombre des places. M_0 désigne le marquage initial.

Suite à une séquence de franchissement s des transitions du modèle RdP modélisant un système, quittant d'un marquage initial M_0 et conduisant au marquage M , l'équation fondamentale du RdP s'écrit :

$$M = M_0 + W.S \quad (2.6)$$

Où S est le vecteur caractéristique correspondant à la séquence de franchissements s .

Cette équation fondamentale permet de connaître le marquage final à partir de la connaissance du marquage initial et de la séquence de transitions franchies. Par contre, elle est impuissante au cas où on veut vérifier si le franchissement d'une séquence donnée est possible ou non.

Définition 2.5

Pour un marquage initial M_0 , une place p est dite bornée ou k -bornée au cas où il existe un entier naturel k , tel que pour tout marquage accessible M_0 , le nombre de marques dans p reste inférieur ou égal à k .

Définition 2.6

Pour un marquage initial M_0 , un RdP est k -borné, si toutes les places sont k -bornées pour M_0 . Un RdP est dit sauf (ou binaire) s'il est 1-borné.

Définition 2.7

Pour tout marquage accessible et pour toute transition t , un RdP marqué est vivant si et seulement si, il existe au moins une séquence de franchissements de transitions qui contient t .

Définition 2.8

Un état puits (ou blocage) s'agit d'un marquage où aucune transition n'est validée.

Définition 2.9

Un RdP est réinitialisable si pour un marquage M_0 , il existe une séquence s de franchissements de transitions qui emmène de nouveau à M_0 .

Définition 2.10

Un conflit est dit structurel au cas où un ensemble de transitions ont au moins une place d'entrée commune.

Définition 2.11

Un conflit effectif convient lors de l'existence d'une part d'un conflit structurel et d'une autre part d'un marquage M tel que le nombre des marques dans la place du conflit est inférieur au nombre de transitions de sortie de cette place qui sont validées par M .

Remarque

Dans le cadre de notre thèse, nous voulons garantir le respect des marges de robustesse temporelles et non temporelles. Le problème qui nous concerne s'agit d'un RdP k -borné, sans blocage, réinitialisable, sans conflit ni structurel ni effectif. Ceci ne peut être concrétisé que si nous introduisons dans nos modèles les RdP à arcs inhibiteurs.

Définition 2.12 [HAC 75]

Un RdP à arcs inhibiteurs est un RdP $R = \langle P, T, Pré, Post \rangle$, dont on modifie seulement la définition de Pré :

$$Pré : P \times T \rightarrow \mathbb{N} \cup \{\emptyset\}$$

Dans un RdP à arcs inhibiteurs, une transition $t \in T$ est franchissable pour un marquage M si :

$$\forall p \in P : \begin{aligned} &Pré(p, t) \leq M(p) \quad \text{si} \quad Pré(p, t) \in \mathbb{N} \\ &M(p) = 0 \quad \text{si} \quad Pré(p, t) = \emptyset \end{aligned} \tag{2.7}$$

$$M(p) = 0 \quad \text{si} \quad Pré(p, t) = \emptyset \tag{2.8}$$

Les éléments tels que $Pré(p, t) = \emptyset$ sont représentés graphiquement à travers un arc extrémité par un petit rond remplaçant la flèche.

Exemple 2.1

La Figure 2.1 illustre le principe de fonctionnement d'un RdP à arcs inhibiteurs. La transition $T1$ ne peut être franchie que si la place $P1$ ne contient pas de jeton et la place $P2$ est marquée (cas 4). Dans ce cas, la proposition Px est vraie si $M(Px) > 0$.

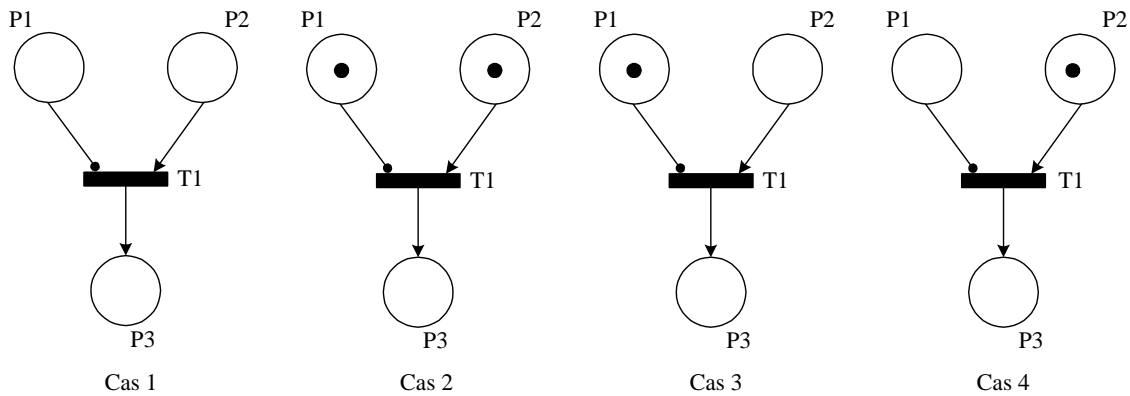


Fig. 2.1 - Principe de fonctionnement d'un RdP à arcs inhibiteurs

Dans la suite, nous nous intéressons qu'aux classes des Réseaux de Petri modélisant des contraintes temporelles et non temporelles. En effet, notre point d'intérêt se focalise sur les RdP P-temporels et les RdP à intervalles, ceci tout en présentant d'autres outils RdP qui prennent en considération, implicitement par leurs définitions, la notion de modélisation des contraintes temporelles.

2.2.2.1- RdP P-temporels

2.2.2.1.1- Définition formelle

Les fondements théoriques de l'outil RdP P-temporels, ont été élaborés par Khansa dans sa thèse [KHA 97]. Il représente un formalisme puissant et reconnu pour la modélisation de l'obligation de respect des temps de séjour (synchronisation sous obligation) [KHA 97].

Définition 2.13 [KHA 97]

Un réseau de Petri P-temporel est un tuple $\langle P, T, Pré, Post, M_0, IS \rangle$ dans lequel $\langle P, T, Pré, Post, M_0 \rangle$ est un réseau de Petri marqué muni d'un marquage initial M_0 et IS est une application définie par:

$$IS : P \rightarrow (\mathbb{Q}^+ \cup 0) \times (\mathbb{Q}^+ \cup \infty)$$

$$p \mapsto IS_i = [a_i, b_i] \quad \text{avec} \quad 0 \leq a_i \leq b_i$$

IS_x définit l'intervalle statique de temps de séjour d'une marque dans la place p_i (\mathbb{Q}^+ est l'ensemble des nombres rationnels positifs). Une marque dans la place p_i ne participe à la validation de ses transitions de sortie seulement si elle a séjourné au moins la durée a_i dans cette place. Elle doit quitter la place p_i et donc franchir l'une des ses transitions de sorties au plus tard quand sa durée de séjour devient b_i . Si elle ne peut pas le faire on dit que la marque est «morte» et ne participera plus à la validation des transitions.

A l'état initial, l'intervalle associé aux marques est $[0, \infty[$, et dès qu'une marque arrive dans une place par franchissement d'une transition, elle prend l'intervalle associé à la place.

2.2.2.1.2- Règles de fonctionnement

La détermination de l'état de fonctionnement d'un RdP P-temporel, nécessite la caractérisation d'une situation donnée du réseau à un instant donné. Cette situation peut être caractérisée en utilisant la notion d'état.

En effet, l'analyse et l'étude de comportement du réseau peut être concrétisé selon deux approches : « approche intervalle de temps » et approche « âge des marques ». La première approche répond au cas où on veut étudier le comportement, la seconde est utilisée pour l'analyse des propriétés d'un réseau.

2.2.2.1.3- Définition fondée sur l'âge des marques

Définition 2.15 [KHA 97]

A un instant donné, l'état est défini par un doublet $E = \langle M, G \rangle$ où :

- M est une application du marquage, assignant à chaque place du réseau un certain nombre de marques ($\forall p \in P, M(p) \geq 0$).
- G est une application temps de séjour qui associe à chaque marque k dans la place p_i un nombre réel g_i^k où g_i^k est l'âge de cette marque (le temps écoulé depuis son arrivée dans la place p_i).

Soit $[a_i, b_i]$ l'intervalle statique associé à la place p_i . Alors une marque k peut participer à la validation de ses transitions de sortie si et seulement si :

- g_i^k n'est pas inférieur à a_i : $g_i^k \geq a_i$
- g_i^k n'est pas supérieur à b_i : $g_i^k \leq b_i$

La marque k est morte quand son âge devient strictement supérieur à b_i .

L'âge g_i^k d'une marque k (donné par une horloge locale associée à celle-ci) est relatif à l'instant d'arrivée de cette marque dans la place p_i .

Soit t l'instant, absolu (donné par une horloge globale), d'arrivée de la marque k dans p_i . L'âge de cette marque à cet instant est égal à zéro. Son âge à l'instant absolu t' est $g_i^k = t - t'$. Elle ne participe à la validation des transitions de sortie de la place qui la contient, qu'à partir de l'instant $t' = t + a_i$ et elle sera morte dès l'instant $t' > t + b_i$.

Une marque sera morte si son âge devient strictement supérieur à la borne supérieure de l'intervalle statique associé à sa place d'accueil et si aucune des transitions de sortie de la place contenant cette marque n'est validée à cet instant.

2.2.2.1.4- Définition fondée sur l'intervalle de temps

Définition 2.16 [KHA 97]

A un instant donné, l'état est défini par un doublet $E < M, I >$ avec :

- M est une application du marquage, désignant à chaque place du réseau un certain nombre de marques ($\forall p \in P, M(p) \geq 0$).
- I est une application de tir, associant à chaque marque k dans une place p_i un intervalle $[a_i^k, b_i^k]$ qu'on nommera « intervalle dynamique » pour le distinguer de l'intervalle statique (initial) associé à la place qui contient cette marque. Cet intervalle dynamique correspond à l'instant d'arrivée de la marque dans la place p_i .

Supposons que la marque k arrive dans la place p_i à l'instant c .

A l'instant $c + d$ ($a_i \leq d \leq b_i$) l'intervalle dynamique de k est :

$$[a_i^k, b_i^k] = [\max(a_i - d, 0), b_i - d] \quad (2.9)$$

Le tir d'une transition dépend des intervalles dynamiques associés aux marques dans toutes ses places d'entrée.

2.2.2.1.5- Condition de franchissement d'une transition

2.2.2.1.5.1- États d'un jeton

Définition 2.17 [KHA 97]

Une transition t_i est potentiellement tirable (validée au sens des RdP P-temporel) à partir de l'état $E(M, G)$ si et seulement si :

- $\forall p_i \in P : m(p_i) \geq Pr \acute{e}(p_i, t_i)$
- $\forall p_i \in P$, il existe au moins $Pr \acute{e}(p_i, t_i)$ marques dans cette place telle que :

$$\min(b_i - \theta_i^k) - \max(0, \max(a_i - \theta_i^k)) \neq 0 \text{ où :}$$

$$k = 1, 2, 3, \dots, Pr \acute{e}(p_i, t_i)$$

$[a_i, b_i]$ est l'intervalle statique associé à la place p_i .

θ_i^k est l'âge de la marque k dans la place p_i .

Une transition peut donc être tirée au sens des RdP autonomes et ne pas l'être au sens des RdP P-temporels à cause des conditions dites aux contraintes de temps.

Une marque peut avoir trois états comme l'indique la figure 2.2 :

- marque non disponible,
- marque disponible,
- marque morte.

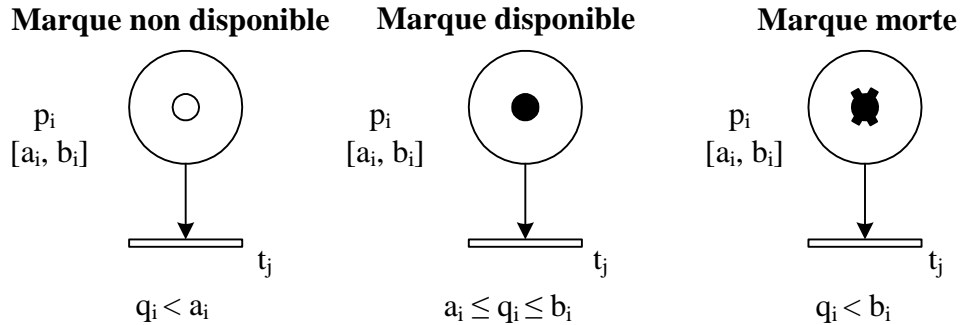


Fig. 2.2 - Différents états d'un jeton dans un RdP P-temporel

2.2.2.1.5.2- Calcul de l'état suivant

Partant de la notion d'état et à partir d'un état donné, on constate que par continuité de temps, les états possibles d'un graphe modélisé par un RdP P-temporel peuvent atteindre un nombre infini. Il existe deux possibilités pour avoir un nouvel état à partir d'un état donné :

- l'écoulement du temps,
- le franchissement d'une transition franchissable.

Les deux définitions suivantes donnent l'état suivant pour chacune de ces deux possibilités.

Définition 2.18 [KHA 97]

L'état $E'(M', G')$ est un état accessible à partir de l'état $E(M, G)$ par écoulement du temps τ si et seulement si :

- $M'=M$,
- $\forall j$ une marque dans la place p_j , nous aurons :

$$a_i'^j = \max(a_i^j - \tau, 0) \quad (2.10)$$

et

$$b_i'^j = b_i^j - \tau \geq 0 \quad (2.11)$$

où a_i^j et b_i^j (respectivement $a_i'^j$ et $b_i'^j$) représentent la borne inférieure et la borne supérieure de l'intervalle dynamique associé à la marque j dans la place p_i depuis l'état E (respectivement E').

Définition 2.19 [KHA 97]

L'état $E'(M', G')$ est atteignable depuis l'état $E(M, G)$ par le franchissement d'une transition t_j si et seulement si :

- t_i est franchissable (tirable) à partir de E
- $\forall p \in P,$

$$M'(p) = M(p) - \text{Pré}(p, t_i) + \text{Post}(p, t_i) \quad (2.11)$$

Les marques qui ne se déplacent pas, gardent le même âge dans E et E' (nous supposons que la durée de franchissement d'une transition est nulle). Les marques déplacées ou créées prennent l'âge zéro associé à leurs nouvelles places d'accueil.

La règle précédente permet de calculer les états et les relations d'accessibilité entre eux. L'ensemble des séquences de tir réalisables depuis l'état initial caractérise le comportement du RdP P-temporel, de la même manière que l'ensemble des marquages accessibles ou les séquences de tir réalisables pour les RdPT et pour les RdP autonomes.

2.2.2.1.6- RdP P-temporels commandés

L'objectif d'introduire ce type des PdP P-temporel est de fournir à la commande des informations sur les instants des tirs afin de les retarder ou les avancer, Ceci, afin que la commande puisse agir sur la commandabilité d'un certain nombre des transitions. Aussi, afin de différencier le fait que des parties du processus de production sont instrumentées par des capteurs et d'autres non. Si des activités de processus de production sont non instrumentées par des capteurs, leurs débuts ou leurs fins ne sont pas directement observables. Les PdP P-temporel commandé emmènent à une distinction entre les transitions : certaines sont observables et d'autres non, certaines sont commandables et d'autres non.

Définition 2.20 [LON 93]

Un RdP P-temporel commandé (R_{PC}) est défini par Long [LON, 93] comme suit :

$R_{PC} = (R_p, \varphi, U, U_0)$, avec :

- R_p est un RdP P-temporel décrivant le système en boucle ouverte.
- φ est l'application de l'ensemble des transitions T de R_p vers l'ensemble des opérations Γ : $\varphi : T \rightarrow \Gamma$; avec $T = T_C \cup T_{UC} \cup T_O \cup T_{UO}$, où T_C est l'ensemble des transitions commandables, T_{UC} est l'ensemble des transitions non commandables, T_O est l'ensemble des transitions observables (la couche commande sait si elles sont tirées ou non sans qu'aucun calcul ne soit effectué), T_{UO} est l'ensemble des transitions non observables et T_S est l'ensemble des transitions de synchronisation.
- U est la commande externe sur les transitions de R_p , construite à partir des prédicats utilisant l'occurrence d'événements observables internes ou externes au système : $U : T \rightarrow \{0, 1\}$
- U_0 est la valeur initiale du vecteur des prédicats.

Une transition est observable si la couche de commande connaît si une transition est tirable ou non sans aucun calcul [COL 97]. Cela coïncide au cas où un capteur est localisé pour la détection directe de la fin ou du début d'une activité.

Une transition $t \in T$ est appelée « état validé » pour un marquage M si elle est validée dans R_P pour M .

Une transition t est nommée « commande validée » (respectivement commande non validée) si $U(t) = 1$ (respectivement $U(t) = 0$). Par définition, les transitions non commandables sont toujours commande validée.

2.2.2.1.7- Exemple d'application des RdP P-temporels

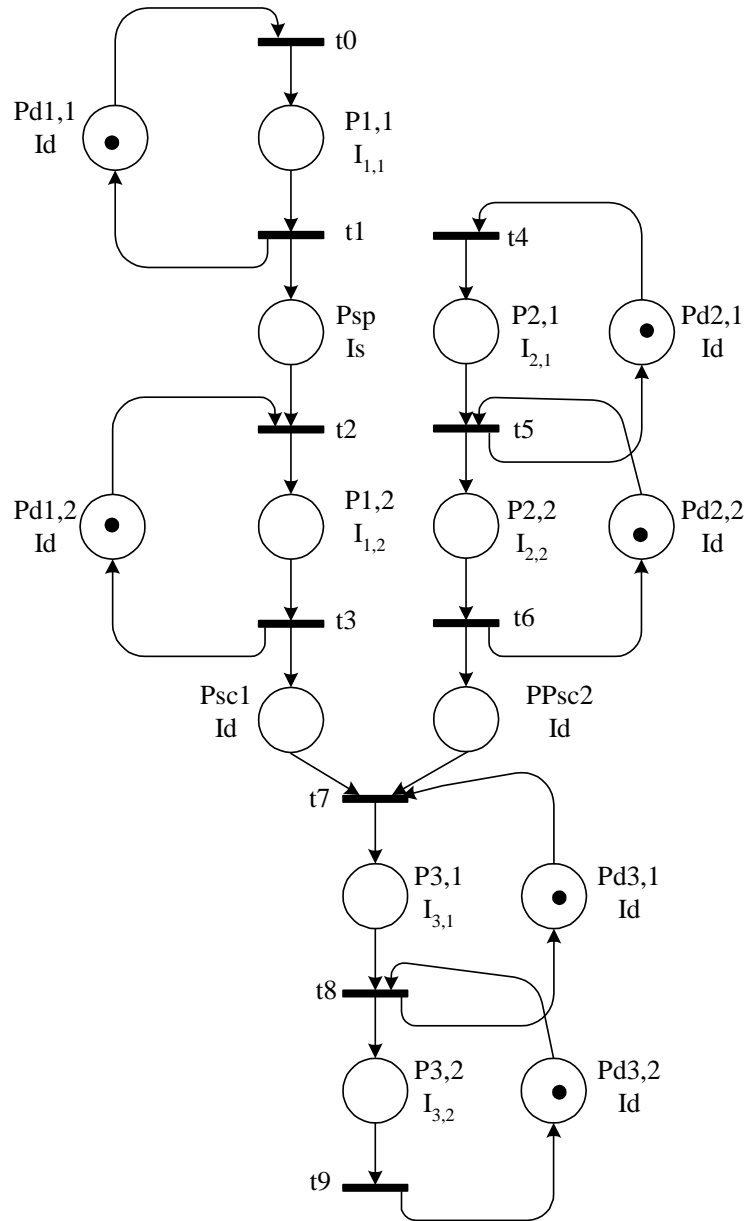


Fig. 2.3 - Modèle RdP P-temporel de S

Considérons un système automatisé de production S composé de trois parties élémentaires E_1 , E_2 et E_3 . E_1 sert à fabriquer un premier produit semi-fini A . E_2 permet la production d'un deuxième produit semi-fini B . E_3 permet, dans un premier lieu, d'assembler A et B par une ressource $R_{3,1}$, puis d'effectuer une transformation mécanique par $R_{3,2}$. E_1 est composée de deux ressources de transformation en séries $R_{1,1}, R_{1,2}$. E_2 est composée de deux ressources de

transformation successives $R_{2,1}$, $R_{2,2}$. Chaque ressource $R_{i,j}$ du système de production est caractérisée par un intervalle temporelle de robustesse $I_{i,j}=[a_{i,j},b_{i,j}]$; l'indice i,j est associé à la $j^{\text{ème}}$ ressources de la $i^{\text{ème}}$ partie élémentaire. Nous considérons que : $b_{1,2}>a_{1,1}$, $a_{2,1}\geq b_{2,2}$ et $a_{3,1}\geq b_{3,2}$.

L'objectif de cet exemple est d'implanter un modèle qui génère la robustesse temporelle globale de système de production

La figure 2.3 présente le modèle RdP P-temporel modélisant le principe de fonctionnement de ce système avec la prise en compte des contraintes temporelles ; Avec :

$P_{i,j}$: place modélisant la ressource $R_{i,j}$,

$I_{i,j}$: intervalle alloué à la place $P_{i,j}$,

$P_{di,j}$: place modélisant la disponibilité de la ressource $R_{i,j}$,

$I_d=[0,+\infty]$: intervalle alloué à la place $P_{di,j}$,

P_{SC1} et P_{SC2} : place modélisant respectivement les deux stocks des produits A et B ,

P_{sp} : place modélisant le stock intermédiaire entre la ressource $R_{1,1}$ et $R_{1,2}$,

$I_s=[0,+\infty]$: intervalle alloué aux places P_{sp} , P_{SC1} et P_{SC2} .

La localisation de la place P_{sp} entre les deux place $P_{1,1}$ et $P_{1,2}$ est due à la contrainte $b_{1,2}>a_{1,1}$ qui peut provoquer un goulet d'étranglement.

Le découplage de deux flux de A et B est garanti par une structure de synchronisation constituée de deux places P_{SC1} et P_{SC2} modélisant respectivement les deux stocks des produits A et B et d'une transition $t7$.

2.2.2.2- RdP à Intervalles

2.2.2.2.1- Définition formelle

Les RdP à intervalles ont été introduits par Collart [COL 03] pour étendre le champ d'application des RdP P-temporels via une abstraction fonctionnelle sur le paramètre associé aux places. Il serait donc logique de retrouver rigoureusement la même définition mathématique de l'outil. Néanmoins, la restriction du paramètre associé aux places ayant des rationnels positifs se justifie quand ce dernier est une durée. Elle n'a plus de raison d'être pour une variation d'un poids ou d'une position par exemple. En effet, cette dernière peut très bien être positive ou négative.

De plus, l'introduction d'un nouveau formalisme a été le prétexte à une remise à plat de la définition initiale. Ainsi, cet outil présente de manière non équivoque le marquage comme un multi-ensemble. De même, la transmission d'une quantité véhiculée avec le jeton est représentée de manière explicite.

Définition 2.21 [DHO05]

Un RdP à intervalles est un tuple $\langle R, m, IS, D, Val, Val_0, X, X_0 \rangle$ dans lequel :

- R est un RdP non marqué,
- m une application qui affecte un marquage au réseau telle que :

Soit Va un ensemble de variables à valeur dans \mathbb{Q} .

Soit V un ensemble non vide de formules utilisant des variables de Va .

Soit μV un multiensemble défini sur l'ensemble V .

$$m : P \rightarrow \mu V$$

$p \in P \rightarrow m(p)$, où $m(p)$ est le marquage d'une place.

On note M l'application qui à chaque place associe le cardinal de son marquage.

- $M : P \rightarrow \mathbb{N} I$ (ensemble des entiers positifs)

$$p \rightarrow \text{card}(m(p)),$$

- $IS : P \rightarrow \mathbb{Q} \cup \{-\infty, +\infty\} \times \mathbb{Q} \cup \{-\infty, +\infty\}$ définit les intervalles associés aux places du réseau :

$$p_i \rightarrow IS_i = [a_i, b_i] \text{ avec } 0 \leq a_i \leq b_i,$$

- D est une application qui associe à chaque couple (place, marque) une valeur rationnelle q qui doit être inférieure au égale à b_i . Cette grandeur correspond à la modification de la valeur associée au jeton qui est intervenue sur la place p_i ,

- $D: m(p) \times P \rightarrow Va$

$$\forall i, 1 \leq i \leq n, n = \text{Card}(P)$$

Soit k un jeton, $k \in m(p_i)$,

$$k \rightarrow q_i \mid a_i \leq q_i \leq b_i,$$

où a_i, b_i sont les valeurs rationnelles fixées par IS ,

- X est une application qui assigne à chaque variable une valeur :

$$X : Va \rightarrow \mathbb{Q}$$

$$va \rightarrow u \in \mathbb{Q}$$

En fixant une valeur à chaque variable, X fixe les q_i :

- X_0 définit les valeurs initiales des variables,
- Val associe à chaque jeton une formule à valeur dans \mathbb{Q} dont la signification physique est volontairement non définie,

Val est une application de l'ensemble des marques $m(P)$ dans V :

$$m(P) \rightarrow V$$

$(k \in m(p)) \rightarrow v \in V$, où k est une marque,

- Val_0 définit les formules initiales associées aux jetons.

2.2.2.2.2- Règles de fonctionnement

Une marque k dans la place p_i participe à la validation de ses transitions de sortie seulement si $q_i(k)$ a acquis la valeur minimale a_i de cette place. Elle doit quitter la place p_i , au plus tard

quand $q_i(k)$ atteint la valeur maximale b_i . Dans le cas contraire, nous disons que la marque devient morte.

Fort logiquement, après le franchissement d'une transition, des jetons sont générés dans les places de sorties et leurs variables associées sont égales à :

$$Val(k)+ q_i(k) \tag{2.12}$$

Les significations physiques des paramètres q et $Val(k)$ sont non définies. Ils sont introduits par une abstraction fonctionnelle.

Bien plus, si nous envisageons l'existence d'une relation entre l'évolution des variables associées aux couples (place, jeton), nous ne la fixons pas dans la définition du modèle. Par exemple, dans les RdP P-temporels, nous trouvons la relation :

$$\forall q, dq/dt=1 \tag{2.13}$$

Où t est le temps.

Dans un RdP à intervalles l'application X n'est pas mathématiquement imposée. On verra, par exemple, une application où les paramètres q représentent des variations de poids des cigarettes. Dans ce cas là, les valeurs des paramètres associés aux couples (place, jeton) sont indépendantes.

2.2.2.2.3- Définition d'un état

Définition 2.22 [DHO05a]

L'état est défini par un quadruplet $E = \langle m, D, Val, X \rangle$ où :

- m affecte un marquage au réseau,
- D et X s'associent pour assigner à chaque marque k dans la place p_i un nombre rationnel $q_i(k)$,
- Val associe une formule utilisant des variables rationnelles à chaque jeton. La valeur effective des ces variables rentre dans la définition de l'état (c'est X qui les fixe indirectement).

Le $q_i(k)$ associé à une marque k dans la place p_i doit être inférieur ou égal à b_i où $[a_i, b_i]$ est l'intervalle statique associé à la place p_i . La marque k dans la place p_i peut participer à la validation de ses transitions de sortie quand sa valeur est supérieure ou égale à a_i . Elle meurt quand sa valeur associée devient supérieure à b_i .

La fixation de valeur est donc possible quand l'évolution des variables associées aux couples (place, jeton) est parfaitement définie. Dans le cas contraire, il existe cependant une possibilité de faire évoluer le modèle et d'en extraire des propriétés. Une abstraction mathématique pourra être utilisée.

2.2.2.2.4- Calcul de l'état suivant

D'après la notion d'état nous constatons qu'un nombre infini d'états peut être atteint à partir d'un état donné. Il existe deux possibilités pour avoir un nouvel état à partir d'un état donné :

l'évolution des variables associées et le franchissement d'une transition franchissable. Les deux définitions qui suivent donnent l'état suivant pour chacune de ces deux évolutions.

Définition 2.23 [DHO05a]

L'état $E'(m', D', Val', X')$ est un état accessible à partir de l'état $E(m, D, Val, X)$ par évolution des variables associés si et seulement si :

- $m'=m$,
- $\forall j$ une marque dans la place p_i :

$$q'_i(j) = q_i(j) + \Delta q_i(j), a_i \leq q'_i(j) \leq b_i \tag{2.14}$$

où a_i et b_i sont les valeurs associées minimum et maximum dans la place p_i .

Définition 2.24 [DHO05a]

L'état $E'(m', D', Val', X')$ est un état atteignable depuis l'état $E(m, D, Val, X)$ par le franchissement d'une transition t_i si et seulement si :

- t_i est franchissable (tirable) à partir de E ,
- $\forall p \in P, m'(p) = m(p) - Pré(p, t_i) + Post(p, t_i)$,
- Les marques qui restent dans la même place, gardent la même valeur associée dans E et E' .

Les marques qui sont créées prennent la valeur zéro pour le compteur q associé à leurs nouvelles places d'accueil, la valeur associée au jeton k' par Val est :

$$Val(k') = Val(k) + q(k) \tag{2.15}$$

où k est un jeton qui était dans une place p_j d'entrée de t_i et qui a été utilisé pour le tir de t_i .

La règle de tir précédente permet de calculer les états et les relations d'accessibilité entre eux. L'ensemble des séquences de tir réalisables depuis l'état initial caractérise le comportement du RdP de la même manière que l'ensemble des marquages accessibles ou les séquences de tir réalisables pour les RdP autonomes.

2.2.2.2.5- Les RdP P-temporels : une sous classe des RdP à intervalles [COL 03]

Les RdP P-temporels présentent par définition une sous classe des RdP à intervalles en posant les restrictions suivantes:

L'ensemble d'arrivé de l'application IS est restreint de $\mathbb{Q} \cup \{-\infty, +\infty\} \times \mathbb{Q} \cup \{-\infty, +\infty\}$ à $(\mathbb{Q}^+ \cup +\infty) \times (\mathbb{Q}^+ \cup +\infty)$.

La signification du paramètre est définie comme étant l'âge du jeton dans une marque. Cette définition est acceptable pour les RdP à intervalles.

L'application D régissant l'évolution des paramètres est complètement fixée puisqu'elle correspond à l'évolution linéaire du temps. Nous vérifions bien que ce temps est une fonction monotone qui fixe l'évolution de tous les âges.

Lemme 2.25 [DHO05a]

Un RdP P-temporel peut être représenté par un RdP à intervalles de validité où la valeur associée au couple (place, jeton) est :

- une durée d'un point de vue fonctionnel,
- un rationnel positif d'un point de vue mathématique.

Théorème 2.26 [DHO05a]

Les propriétés structurelles démontrées sur les RdP P-temporels sont transposables à toute instance des RdP à intervalles si et seulement si :

$$\forall i | 1 \leq i \leq n, n = \text{Card}(P), \min_i(q_i) \neq -\infty \tag{2.16}$$

2.2.2.2.6- Exemple d'application des RdP à Intervalles

L'exemple considéré correspond à celui étudié dans la thèse de Dhouibi [DHO 05a]. En effet, le système étudié est une unité de manufacture de confection des cigarettes. Le contrôle du poids des unités fabriquées conduit à l'amélioration du taux de production, la qualité et le coût, en limitant les rebuts.

Pour une cigarette donnée, le poids est fonction de cinq paramètres : le module, la longueur, la compacité, le niveau du tabac sur le tapis d'alimentation (le débit) et la quantité d'écrêtage. Les trois derniers définissent la valeur de la densité de cigarette. L'étude présentée consacrera aux paramètres module (*m*), longueur (*l*) et densité (*d*) afin de maintenir le poids (*P*) d'une cigarette.

L'objectif est d'examiner le respect de la tolérance sur le poids via les trois paramètres *m*, *l* et *d* qui doivent rester, chacun à son tour, dans un intervalle donné.

Une cigarette peut être assimilée à un cylindre de diamètre (*m*) et de longueur (*l*). Le poids (*P*) de la cigarette est donné alors par la relation suivante :

$$P = k \cdot m^2 \cdot l \cdot d \tag{2.17}$$

Où *k* est une constante ($k = \frac{\pi}{4}$).

Après un développement de Taylor au premier ordre et le remplacement de l'accroissement de cette fonction par l'accroissement d'une forme linéaire la relation devient :

$$\Delta P \leq K_1 \Delta d + K_2 \Delta m + K_3 \Delta l \tag{2.18}$$

Avec :

$$K_1 = k l m^2,$$

$$K_2 = 2 k l m d,$$

$$K_3 = k d m^2,$$

$$\Delta P = [P_{min}, P_{max}].$$

Le réseau de Petri de la figure 2.4 modélise l'équation différentielle (2.18). Il s'agit d'une modélisation des paramètres influant la variation du poids dans les trois postes de transformation : la densité, le module et la longueur.

Les places $P1$, $P2$ et $P3$ ont la fonction de borner la quantité de jetons pour une opération donnée. Elles présentent, en elles mêmes, les places modélisant la disponibilité respectivement de $P1$, $P2$ et $P3$. La place Pd modélise la contrainte mise sur la densité. La place Pm modélise la contrainte mise sur le diamètre. La place Pl modélise la contrainte mise sur la longueur. La place Pp modélise la contrainte mise sur le poids. La place $P4$ aide à la prise de décision quant à la disponibilité de la Pd . Les intervalles alloués à $P1$, $P2$, $P3$ et $P4$ modélisent les contraintes temporelles de disponibilité.

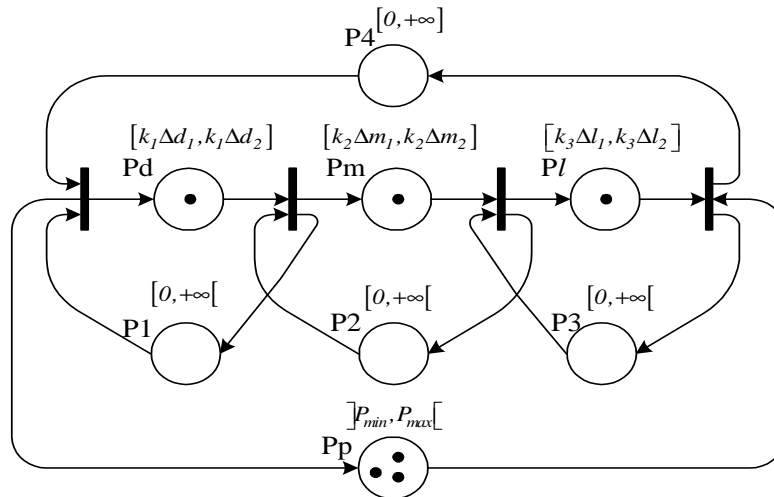


Fig.2.4- RdP à intervalles « poids valué »

2.2.2.3- Autres outils RdP pour la modélisation des contraintes

Dans la littérature, nous rencontrons d’autres outils RdP qui intègrent la notion de temps dans leurs fondations théoriques. Nous mentionnons notamment les RdP t-temporels et les RdP à flux temporels. Ces types des RdP modélisent des contraintes temporelles.

2.2.2.3.1- RdP t-temporels

2.2.2.3.1.1- Définition formelle

Définition 2.27 [BER 91]

Un réseau de Petri t-temporel est un tuple $\langle P, T, Pré, Post, M_0, IS \rangle$ dans lequel :

- $\langle P, T, Pré, Post, M_0 \rangle$ est un Réseau de Petri,
 - $IS : T \rightarrow \mathbb{Q}^+ \times (\mathbb{Q}^+ \cup \{\infty\})$ la fonction intervalle statique (\mathbb{Q}^+ est l’ensemble des nombres rationnels).
- $t_i \rightarrow IS_i = [a_i, b_i]$ avec $0 \leq a_i \leq b_i$.

a_i est la date statique de tir au plus tôt et b_i est la date de tir au plus tard de t_i .

Une transition est sensibilisée (validée) à partir de l’instant a_i avant de pouvoir être tirée (franchie). Si elle n’est pas tirée durant l’intervalle IS_i , elle ne peut être jamais sujet de sensibilisation.

2.2.2.3.1.2- Etats et règle de tir

Définition 2.28 [BER 91]

Un état d’un réseau t-temporel est une paire $E = \langle M, I \rangle$ dans lequel :

- M est un marquage,
- I est une application intervalle de tir, assignant à chaque transition du réseau l'intervalle de temps dans lequel elle peut être tirée $t \in T, I(t) = \emptyset$ si et seulement si t est non validée).

L'état initial E_0 est formé du marquage initial M_0 et de l'application intervalle de tir I_0 qui fait transmettre, à chaque transition t validée par M_0 , son intervalle statique de tir $IS(t)$.

A une date relative τ et à partir d'un état $\langle E, I \rangle$, une transition t ne peut être franchissable que si est seulement si les deux conditions suivantes sont satisfaites :

- La transition t est sensibilisée par le marquage $M : \forall p \in {}^\circ t : M(p) \geq Pr \acute{e}(p, t)$,
- τ n'est pas strictement inférieure à la date au plus tôt de t et τ n'est pas strictement supérieure à la date au plus tard des autres transitions sensibilisées par : $M : \forall k, M \in {}^\circ t : Pr \acute{e}(k) \Rightarrow \tau \leq b_k$.

Le franchissement d'une transition validée t , à l'instant τ , à partir de l'état $E = \langle M, I \rangle$, conduit à un nouveau état $E' = \langle M', I' \rangle$ déterminé comme suit :

le nouveau marquage M' est déterminé par l'équation classique :

$$\forall p \in P, M'(p) = M(p) - Pr \acute{e}(p, t) + Post(p, t) \tag{2.19}$$

le nouveau intervalle de tir I' pour les transitions, est :

- 1) $I' = [0, 0]$, si les transitions k sont non sensibilisées par M' ,
- 2) $I' = [max(0, a_{k-\tau}), b_{k-\tau}]$ si $b_{k-\tau}$ est fini } si les transitions j sont sensibilisées par M , et
 $I' = [max(0, a_{k-\tau}), \infty]$ si non } non en conflit avec la transition t pour M ,
- 3) si non,
 $I' = I([a_i, b_i])$ l'intervalle statique associé aux autres transitions.

2.2.2.3.1.3- Exemple d'application des RdP t-temporels

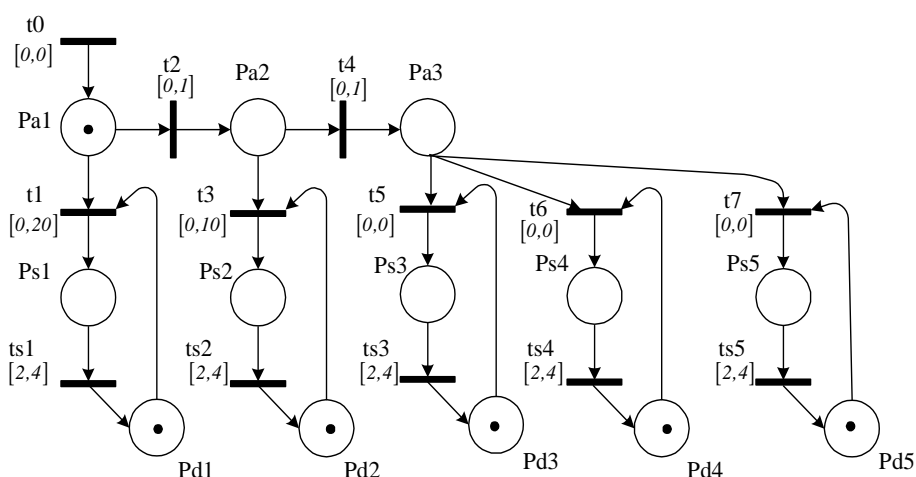


Fig.2.5- Modèle RdP t-temporel du principe de fonctionnement des files d'attente

Soit une agence bancaire équipée par cinq guichets de service. Afin qu'elle concrétise les besoins de service de ses clients, elle a mis les règles de fonctionnement suivantes pour la gestion des files d'attente :

Si un client arrive à la banque à un instant t , il passe à la file d'attente du guichet numéro 1. S'il n'est pas servi avant 20 mn de son arrivée, il passe à la file d'attente du guichet numéro 2. Si le client dépasse une durée de 10 mn dans la file d'attente du guichet n° 2, il sera servi automatiquement par l'un des autres guichets. Le temps de déplacement d'un guichet à un autre est estimé entre 0 mn et 1mn. Le temps de service est estimé entre 2mn et 4 mn.

La figure 2.5 présente le modèle RdP t-temporel modélisant le principe de fonctionnement des files d'attente de cette agence bancaire.

Avec :

$t0$: transition modélisant la ressource l'entrée du système, elle modélise la porte de l'agence,

$Pa1, Pa2$ et $Pa3$: places modélisant les files d'attente,

$t2$ et $t4$: transitions modélisant la durée de déplacement d'une file d'attente à une autre,

$t1, t3, t5, t6$ et $t7$: transitions modélisant l'intervalle de la durée d'attente allouée à chaque file d'attente,

$Ps1, Ps2, Ps3, Ps4$ et $Ps5$: place modélisant les guichets de service,

$ts1, ts2, ts3, ts4$ et $ts5$: transitions modélisant l'intervalle de la durée de service,

$Pd1, Pd2, Pd3, Pd4$ et $Pd5$: place modélisant la disponibilité des guichets de service.

2.2.2.3.2- RdP temporels à flux indépendants

Soit A_j l'ensemble des arcs qui entrent dans une transition t_j (l'ensemble est noté A pour une transition t) :

$$A_j = \{a_k = (p_k, t_i) \mid \text{Pré}(p_k, t_i) \neq 0\} \quad (2.20)$$

Où, $\text{Pré}(p_k, t_i)$ est le poids de l'arc a_k

2.2.2.3.2.1- Définition formelle

Définition 2.29 [DIA 03]

Un réseau de Petri temporels à flux indépendants est un triplet $\langle R, ITA, SYN \rangle$, avec :

- R est un réseau de Petri places-transition,
- ITA est une application qui associe, à chaque arc a_i un intervalle :

$$ITA : A_j \rightarrow \Theta^+ \times \Theta^+ \times (\Theta \cup \{\infty\})$$

Soit le triplet $\langle \alpha_i, n_i, \beta_i \rangle$ associé à l'arc ou l'intervalle $[\alpha_i, \beta_i]$, α_i représente la date de tir au plus tôt, n_i une valeur nominale et β_i la date au plus tard. La valeur n_i peut ainsi être absente dans l'intervalle ou notée $*$.

- SYN est une fonction de typage des transitions ; elle associe à chaque transition un type, c'est à dire une sémantique donnée.

$$SYN : T \rightarrow \{\text{sémantique}\}$$

2.2.2.3.2.2- Règles de franchissement

Définition 2.30 [DIA 03]

Un arc (p_i, t) est franchissable par un marquage M quand sa place d'entrée p_i acquiert un nombre de jetons supérieur au poids de l'arc, ceci signifie $M(p_i) \geq \text{Pré}(p_i, t)$. Si à l'instant τ

l'arc est sensibilisé, le jeton lié à cet arc doit être tiré dans l'intervalle $[\alpha + \tau, \beta i + \tau]$. La valeur de référence de tir sera alors $\tau + n_i$.

2.2.2.3.2.3- Exemple d'application des RdP temporels à flux indépendants

Nous spécifions, figure 2.6, un PC qui porte un bouton Démarrage/Arrêt qui permet le démarrage et la fermeture. Le bouton est actif infiniment. L'intervalle modélisant ce phénomène est $[0, +\infty[$. Lorsque l'utilisateur appuie sur le bouton Démarrage/Arrêt, le PC se démarre et le bouton devient un bouton de fermeture (arrêt). Pour des raisons de sécurité, le constructeur conseille que fonctionnement du PC dure entre 3 et 480 unités de temps, et le bouton n'est actif de nouveau, après le démarrage, qu'après 3 unités de temps (le temps de démarrage).

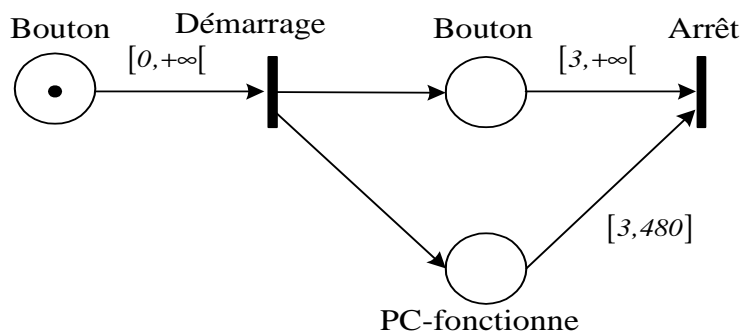


Fig.2.6- Modèle RdP temporel à flux indépendants de fonctionnement d'un PC

2.2.3- Synthèse et choix des outils de modélisation adéquats pour notre approche

Dès le début du paragraphe précédent, nous avons montré que les automates à états finis ne sont pas aptes à considérer les contraintes dictées par les objectifs de cette thèse. Cet outil est non qualifié à gérer des synchronisations, des parallélismes, des conflits et des partages de ressources. Par contre, l'outil de Réseaux de Petri est bien capable d'introduire la notion des contraintes et de bien gérer les alternances et les intersections des intervalles dues à la complexité des procédés industriels.

En effet, il existe plusieurs types des RdP qui permettent, à partir de leurs définitions, la modélisation des intervalles temporels ou non temporels : RdP t-temporels, RdP P-temporels, RdP temporels à flux indépendants et RdP à Intervalles.

Chaque type des RdP, qui intègrent des contraintes dans leurs définitions, sert des cas bien déterminé. En effet,

- le principe des RdP t-temporels consiste à associer à chaque transition du réseau un intervalle temporel de sensibilisation contrôlé implicitement par une horloge globale du RdP. La transition est considérée comme une activité et non comme un événement instantané. Ces modèles sont généralement exploités pour l'étude des systèmes de télécommunication [MER 74] ou pour passer d'un modèle discret à un autre continu lors de la modélisation des systèmes batch [AND 96].
- les RdP P-temporels sont fondées pour modéliser des intervalles de séjour. Leur principe admet qu'une place est spécifiée par un intervalle temporel modélisant le temps normalement alloué à une activité.

La non-disponibilité, la disponibilité et la mort d'un jeton séjourné dans une place sont contrôlées par l'horloge globale du RdP. Ce type des RdP est exploité, dans la plus part des cas, pour modéliser la robustesse temporelle des systèmes industriels [KHA 96], [KHA 97], [COL 97], [BON 00], [BON 05], [JER 06].

- les RdP temporels à flux indépendants accèdent à la modélisation des systèmes qui disposent d'un écoulement temporel propre, autonome et compliqué. Ces réseaux permettent de définir la synchronisation temporelle qui s'exécute. Les intervalles de validité temporelle associés aux arcs servent à gérer le passage d'une situation (acte, position) à une autre tout en modélisant la sémantique du modèle. Les RdP temporels à flux indépendants présentent un outil de modélisation unificateur pour les approches intégrant spécification et surveillance.
- l'outil des RdP à intervalles présente une extension des RdP P-temporels. L'abstraction fonctionnelle introduite au niveau des RdP à Intervalles accède à permettre de prendre en compte de contraintes de natures très diverses dans un même modèle. Autre que la prise en considération des contraintes à intervalles temporels, cette classe des RdP permet de modéliser des intervalles spécifiant des paramètres non-temporels.

Notre objectif, en ce qui concerne la commande, est d'implanter une loi de commande robuste gérant à la fois des contraintes mises sur le flux et la qualité du produit. La loi de commande que nous voulons l'implanter dans notre architecture devrait être modélisée par un outil apte à prendre en considération, à travers son concept de base, des contraintes temporelles et non temporelles. Des intervalles de séjours ainsi que des contraintes bornées sur les paramètres qualité lors des séjours vont être les paramètres sujets à modélisation.

En effet, les auteurs de travaux qui s'intéressent à la conception et la modélisation des lois de commande robuste ont exploité l'outil RdP P-temporel commandé pour modéliser des contraintes temporelles et non-temporelles. Ceci est dû à l'absence d'un outil Réseau de Petri permettant la modélisation des contraintes non temporelles. Les RdP P-temporels ont été exploités sous des hypothèses qui corrélaient la qualité (paramètres non temporels) au flux (paramètres temporels). Les contraintes non temporelles ont été masquées par d'autres intervalles temporels qui les simulent.

La corrélation entre les paramètres temporels de flux et les paramètres non temporels de qualité n'est plus nécessaire dès l'apparition de l'outil RdP à Intervalles en 2003 [COL 03].

D'après ce qui a été introduit précédemment, l'outil le plus efficace et adéquat pour notre approche ne peut être que les RdP à Intervalles. En effet, cet outil est capable de modéliser à la fois des marges de robustesse temporelles de flux et non temporelles de qualité.

2.3- Outils et méthodes pour la surveillance des systèmes industriels

Comme il est mentionné au premier chapitre, la surveillance consiste notamment à détecter et classer les défaillances ou les symptômes des défaillances tout en suivant l'évolution du système, puis à les diagnostiquer en localisant les éléments défaillants et en identifiant les causes premières. Le pronostic est une fonction auxiliaire qui permet dans les approches classiques d'analyser, d'estimer et de prédire les conséquences d'un symptôme ou d'un défaut sur le fonctionnement futur du système. Pour notre approche : « Surveillance Indirecte Suite

à une Commande Robuste », cette fonction a l'objectif de prédire les états futurs des paramètres de la loi de commande. Ceci afin de maintenir la robustesse du système de production.

La figure 2.7 présente les techniques les plus courantes en surveillance d'équipements industriels.

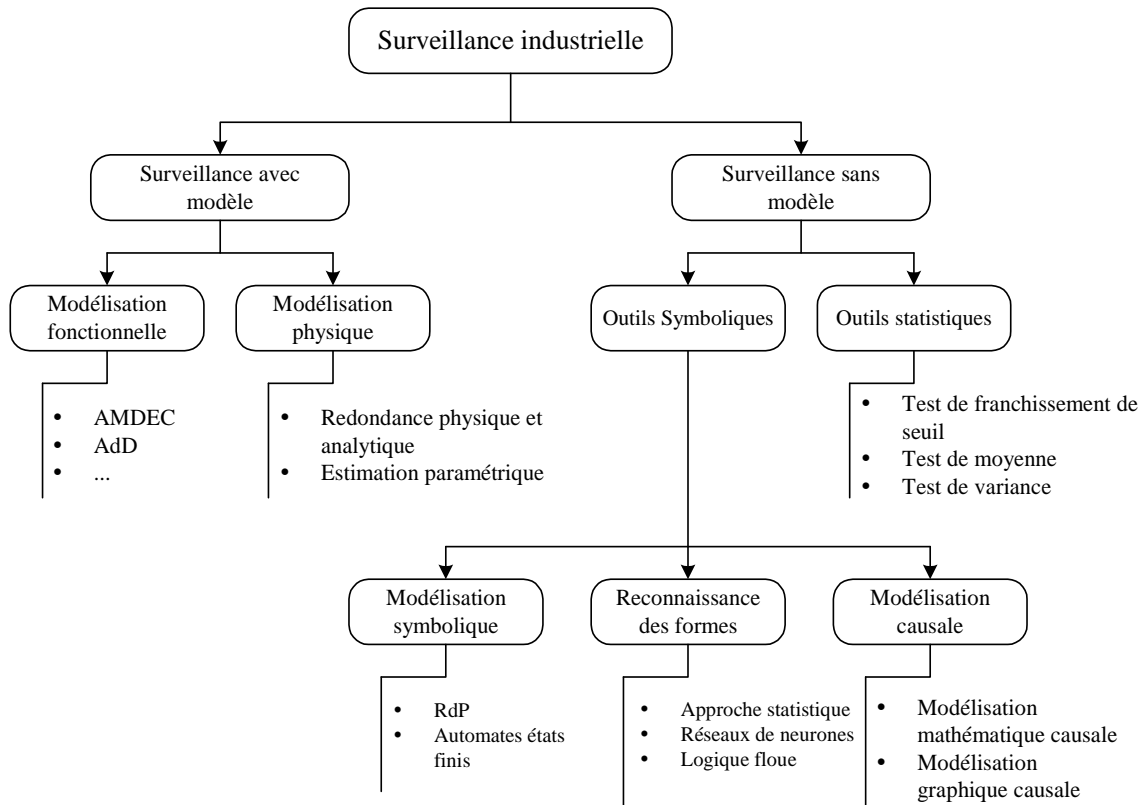


Fig. 2.7 - Classification des méthodologies de surveillance industrielle

Généralement, nous distinguons deux classes d'outils et des méthodes pour la surveillance des systèmes industriels :

- outils et méthodes de surveillance avec modèle : nous distinguons pour cette classe deux sous-classes qui sont la modélisation fonctionnelle et la modélisation physique des paramètres de système,
- outils et méthodes de surveillance sans modèle : nous distinguons aussi pour cette classe deux sous-classes qui sont les statistiques et les outils symboliques. cette dernière comprend en elle même trois sous-sous-classes qui sont la modélisation symbolique, la reconnaissance des formes et la modélisation causale.

Nous présentons, dans la suite de ce paragraphe, les différents outils et méthodes de surveillance tout en concentrant notre intérêt sur les fonctions cibles de notre approche : pronostic et suivi-détection.

2.3.1- Méthodes et outils de surveillance avec modèle

Le principe des méthodes et des outils de surveillance avec modèle consiste à comparer les mesures achevées sur le système aux résultats approvisionnés par le modèle [FRA 90]. Tout écart est alors synonyme d'une défaillance.

2.3.1.1- Surveillance par modélisation fonctionnelle

La surveillance par modélisation fonctionnelle consiste à établir de la manière la plus complète possible les liens entre les symptômes et leurs effets indésirables qui peuvent cesser le fonctionnement normal du système sujet à surveillance. La méthode « Analyse des Modes de Défaillance, de leurs Effets et de leurs Criticité (AMDEC) » et la méthode « Arbres de Défaillances (Add) » sont les plus connues et exploitées dans le domaine de surveillance industrielle par modélisation fonctionnelle.

2.3.1.1.1- AMDEC

En effet, l'AMDEC est une méthode d'analyse préventive de surveillance qui permet de traiter systématiquement, composant par composant, tous les modes de défaillance possibles tout en indiquant leurs effets sur l'ensemble du système surveillé. Cette méthode est caractérisée par deux phases. La première phase d'analyse de criticité permet le passage à la deuxième qui accède à l'organisation des risques afin de se concentrer à réduire les plus fatales défaillances qui sont classées inacceptables.

Cette technique a l'avantage d'analyse préventive pour détecter les défaillances éventuelles, estimer les risques et enclencher des actions de prévention.

L'AMDEC se présente généralement sous forme d'un tableau (tableau 2.1)

Fonction	Élément	Défaillance	Causes	Effets	Détection	Criticité

Tab. 2.1 - l'AMDEC sous forme d'un tableau

Selon la complexité du système, d'autres champs peuvent être ajoutés dans le tableau de l'AMDEC comme par exemple la fréquence ou la gravité.

2.3.1.1.2- Arbres de Défaillances

Historiquement, l'analyse par arbre des défaillances est la première méthode mise en œuvre en vue d'examiner systématiquement les risques de défaillance.

L'analyse par arbre de défaillances est une méthode de type déductif. En effet, elle consiste à rechercher tous les combinaisons et les enchaînements possibles qui peuvent conduire à un événement indésirable et les représenter sous une forme graphique au moyen d'une structure arborescente exploitant les symboles des portes logiques.

L'analyse par arbre des défaillances d'un événement redouté peut se décomposer en cinq étapes successives :

- connaissance du système,

- connaissance préalable des risques,
- définition de l'événement redouté étudié,
- élaboration de l'arbre,
- exploitation de l'arbre.

2.3.1.2- Méthodes de surveillance par modélisation physique

Les méthodes de surveillance par modélisation physique peuvent être divisées en deux techniques : techniques de redondance physique et analytique et techniques d'estimation paramétrique.

2.3.1.2.1- Redondances physiques et analytiques

2.3.1.2.1.1- Redondances physiques

Cette méthode a le but d'augmenter la fiabilité de détection des symptômes à partir des signaux mesurés. Elle présente un moyen pour différencier les défaillances capteurs des symptômes des défaillances ou des défaillances système.

Elle consiste à multiplier les capteurs de mesure dans la même partie du système de production. Si un ensemble des composants de mesure placé dans la même partie du système dont le but de mesurer une grandeur physique qualitative ou quantitative donnée attribue des signaux identiques, on constate que ces composants sont dans un état de fonctionnement nominal. Dans ce cas, la grandeur mesurée est incontestable ; elle sera utilisée pour l'étude de l'aptitude de fiabilité de système à travers les fonctions suivi et détection et par la suite par les fonctions diagnostic et pronostic si un symptôme est détecté. Dans le cas contraire, on constate que le capteur émettant une mesure différente est défaillant [ZHA 99].

La partie conceptuelle présente l'avantage de cette méthode mais au niveau investissement elle est coûteuse. Pour cette raison, il est conseillé de ne pas utiliser cette méthode que pour les sous-parties critiques des systèmes de production.

2.3.1.2.1.2- Redondances analytiques

Les méthodes de redondance analytique se représentent sous la forme d'un modèle qui comprend un certain nombre de paramètres du système à surveiller. Les valeurs sont prétendues connues lors du fonctionnement nominal du système. Si nous exploitons des mesures échantillonnées des grandeurs observables du système, le modèle de système s'écrit sous une forme discrète similaire aux modèles qui modélisent des mesures continues.

Ces méthodes permettent l'estimation d'état futur du système afin de le comparer à son état réel. L'estimation de l'état du système peut être réalisée soit à l'aide de techniques d'estimation d'état, soit par obtention de relations de redondance analytique.

Les relations de redondance analytique sont des équations qui relient entre les variables du système, mesurées dans une même fenêtre temporelle.

A titre d'exemple, nous présentons dans la suite le modèle d'espace d'état discret échantillonné qui met le vecteur d'état $x(k)$ en fonction du vecteur d'entrée $u(k)$ et du vecteur de sortie $y(k)$ du système à surveiller à travers les matrices A , B et C qui sont indépendantes du temps. Les équations de ce modèle s'écrivent sous la forme suivante :

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k)\end{aligned}\tag{2.20}$$

Les défaillances et les perturbations du système se modélisent en exploitant le principe de ces équations. Les fautes et les bruits du procédé se modélisent à travers l'équation suivante :

$$x(k+1) = Ax(k) + Bu(k) + PF_p(k) + Qb_p(k) \quad (2.21)$$

Où :

$PF_p(k)$: les fautes du procédé,

$Qb_p(k)$: les bruits du procédé.

Les erreurs de mesures des entrées et des sorties ainsi que les bruits des entrées et des sorties sont modélisés par les relations suivantes :

$$\begin{aligned} \tilde{u}(k) &= u(k) + F_u(k) + b_u(k) \\ \tilde{y}(k) &= y(k) + F_y(k) + b_y(k) \end{aligned} \quad (2.22)$$

$F_u(k)$ et $F_y(k)$: les erreurs de mesures, respectivement, des entrées et des sorties,

$b_u(k)$ et $b_y(k)$: les bruits de mesures, respectivement, des entrées et des sorties,

$\tilde{u}(k)$: le degré de réalisation du signal de commande $u(k)$ par les actionneurs,

$\tilde{y}(k)$: la mesure du signal $y(k)$.

Les différences entre les matrices de paramètres du modèle et celles du système réel s'écrivent par les équations suivantes:

$$\begin{aligned} \tilde{A} &= A + \Delta A(k) \\ \tilde{B} &= B + \Delta B(k) \\ \tilde{C} &= C + \Delta C(k) \end{aligned} \quad (2.23)$$

Tout écart entre les sorties estimées et les sorties réelles signifie une défaillance du système au niveau d'un composant bien identifié ou plus. Une décision est ensuite prise afin de déterminer si l'écart observé est dû à des aléas normaux lors de fonctionnement du système ou à des défaillances.

2.3.1.2.2- Estimation paramétrique

Ce sont des méthodes qui exigent la connaissance des paramètres de fonctionnement nominal du système. Ces méthodes décrivent le comportement du système à travers une modélisation mathématique de ces paramètres. Pour ces méthodes, les paramètres devront avoir un sens physique.

Les paramètres des modèles définissant le fonctionnement réel ainsi que leurs variations sont extraits à partir de mesures des entrées et des sorties du système. Ces méthodes sont énormément performantes lorsque les paramètres du système ainsi que leurs structures sont bien connus et les défauts ont une incidence sur les paramètres du système. Il est normal donc de constater que ces méthodes sont applicables impérativement à la surveillance des procédés

qui ont un faible nombre de paramètres. L'application de ces méthodes sur des systèmes complexes qui ont des différents modes de fonctionnement est très difficile [MON 97].

La différence entre les méthodes de redondance analytique et celles d'estimation paramétrique est que les premières confrontent entre l'état estimé et l'état théorique du système, bien que les secondes sont plus précises vu la comparaison directe entre les paramètres estimés et les paramètres théoriques du système.

[ISE 84] a défini une procédure générale d'estimation paramétrique comprenant cinq étapes :

1^{ère} étape : Établissement du modèle mathématique du procédé,

$$y(t) = f(u(t); \theta) \quad (2.24)$$

Où :

$u(t)$: les entrées du système,

$y(t)$: les sorties du système,

θ : le vecteur des paramètres physiques du modèle.

2^{ème} étape : Détermination des relations entre les paramètres du modèle et les paramètres du procédé,

$$\theta = g(p) \quad (2.25)$$

Où :

p : les constantes connues physiques du système, elles seront modifiées lorsqu'une défaillance survient.

3^{ème} étape : Estimation de $\hat{\theta}$ des paramètres θ par l'utilisation de l'équation (2.24).

$$\hat{\theta} = h(y(1) \dots y(t); u(1) \dots u(t)) \quad (2.26)$$

4^{ème} étape : Estimation de \hat{p} des paramètres p par l'utilisation de l'équation (2.25).

$$\hat{p}(t) = g^{-1}(\hat{\theta}(t)) \quad (2.27)$$

5^{ème} étape : Détermination de présence de défaillance du système ou non.

Cette étape consiste à comparer soit les paramètres modèle réels θ et estimés $\hat{\theta}$, soit les paramètres physique réels p et estimés \hat{p} .

En conclusion, ces méthodes appellent donc à bien élaborer un modèle dynamique précis et solide du système à surveiller. Ce système devrait être non complexe avec un nombre réduit des paramètres dont les interactions sont simples et bien connues. Les valeurs estimées seront exploitées, par la suite, comme base pour la détection, le diagnostic et le pronostic des défaillances.

2.3.2- Méthodes et outils de surveillance sans modèle

2.3.2.1- Surveillance avec outils statistiques

Ce sont des méthodes de détection de défaillances qui consistent à considérer que les signaux émis par les capteurs ont des propriétés statistiques. Des tests statistiques permettent, par la suite, de vérifier si ces propriétés se manifestent dans un échantillon des signaux mesurés ou non. Il existe plusieurs types de tests, appropriés aux échantillons mesurés [BAS 88]. Dans la suite de ce sous-paragraphe nous ne présentons que trois parmi ces tests statistiques : test de franchissement de seuils, test de moyenne et test de variance.

2.3.2.1.1- Test de franchissement de seuils

C'est un test très simple qui consiste à comparer les mesures émises par les capteurs avec des seuils de conformité préétablis. Généralement, on distingue deux types de seuils. Le premier type est dit seuil de pré-alarme, il conduit à une intervention de maintenance préventive. Le second type est appelé seuil d'alarme, il exige l'arrêt de la production en vue d'effectuer une action de maintenance corrective. Cette méthode est sensible face aux fausses alarmes.

2.3.2.1.2- Test de moyenne et test de variance

Ce sont deux tests semblables qui permettent la comparaison de la moyenne (ou de la variance) de l'échantillon des mesures émises avec la moyenne (ou la variance) de conformité. Tant que la moyenne (ou la variance) se place dans une marge donnée située autour d'une valeur nominale bien établie, l'évolution du système est supposée normale.

2.3.2.2- Surveillance avec outils symbolique

2.3.2.2.1- Surveillance par modélisation symbolique

Ce sont des méthodes qui modélisent, selon l'objectif de la surveillance, le comportement du système afin de simuler son état réel. Généralement, ces modèles traitent les modes de bon fonctionnement contrairement aux modèles numériques qui se basent sur la physique du système. Ce sont méthodes exploitables, souvent, pour les fonctions suivi et détection.

On distingue dans la littérature deux types des outils pour la surveillance avec modélisation symbolique qui sont les automates à Etats Finis et les Réseaux de Petri.

L'outil Réseaux de Petri Temporels à Objets (RdPTàO) présente un outil très efficace pour le suivi de comportement du système et la détection des symptômes des défaillances [COM 91], [DAS 03], [MIN 03], [MIN 04], etc.

Nous citons dans la suite la définition ainsi que le principe de fonctionnement des RdPTàO.

Définition 2.31 [VAL 01]

Un Réseau de Petri Temporels à Objets est constitué de :

- *Un ensemble fini P de Places.*
- *Un ensemble fini T de Transitions. chaque transition peut être composée de plusieurs blocs :*
 - *Le premier bloc est celui relatif aux prédicats : il supporte l'expression de la condition de franchissement.*
 - *Le deuxième bloc est dédié à l'aspect temporel : il permet d'exprimer des intervalles de franchissement d'une transition. Un intervalle temporel I est constitué de deux bornes $[a, b]$: a et b étant respectivement la date de tir au plus tôt et la date de tir au plus tard.*

- *Le troisième bloc est dédié au verrouillage de la transition. il est possible d'interdire le franchissement d'une transition ou d'un ensemble de transitions.*
- *Le quatrième bloc est relatif aux actions. Les actions sont en effet associées aux transitions et exécutées lors du tir de la transition. Le jeton ne sera déposé dans la place aval uniquement lorsque l'action sera terminée.*
- *Un ensemble fini C_{class} de Classes, éventuellement organisé en une hiérarchie et définissant pour chaque Classe un ensemble d'attributs.*
- *Un ensemble de variables V typées par Class.*
- *Une fonction place précédente appelée Pré qui à chaque Arc d'entrée d'une Transition fait correspondre une somme formelle de n-uplets d'éléments de V .*
- *Une fonction place suivante appelée Post qui à chaque Arc de sortie d'une Transition fait correspondre une somme formelle de n-uplets d'éléments de V .*
- *Une application A_{tc} qui à chaque Transition associe un ensemble de conditions qui font intervenir les variables formelles associées aux arcs d'entrée et les attributs des classes correspondantes.*
- *Une application A_{ta} qui à chaque Transition associe un ensemble d'actions qui font intervenir les variables formelles associées aux arcs d'entrée et de sortie, les attributs et les méthodes des classes correspondantes.*
- *Une application M_0 est le marquage initial qui associe à chaque Place une somme formelle de n-uplets d'instances d'objets appelés Jetons (les objets doivent être représentés par des identificateurs, leurs noms par exemple).*

Une activité est représentée par un RdPTàO comme il est montré sur la figure 2.8

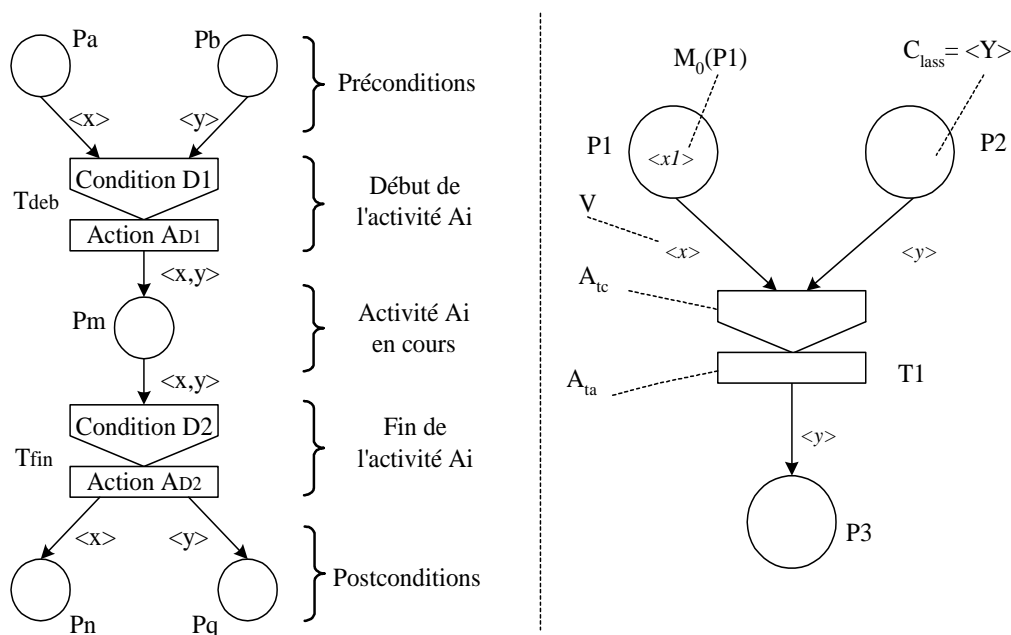


Fig. 2.8- Modélisation d'une activité par RdPTàO

Un tel réseau possède les caractéristiques suivantes :

- prise en compte des contraintes logiques
- prise en compte des contraintes temporelles
- création d'objets par franchissement
- modélisation des politiques décisionnelles locales
- absence de blocage.

Ces méthodes de modélisation symbolique sont généralement exploitées en les fusionnant avec d'autres outils tels que la logique floue, le test de franchissement de seuils, etc.

2.3.2.2.2- Surveillance par modélisation causale

Ce sont des modèles qui permettent une modélisation liant les défaillances avec leurs causes. Elles englobent les notions de modèles de pannes introduites dans [AGH 97]. En effet, ce type de modélisation offre un support explicatif et adductif associé, généralement, au diagnostic et notamment à la tâche de localisation.

Nous distinguons dans la littérature deux classes de structure causale :

- Modélisation mathématique causale : elle permet d'écrire un modèle des relations causales issues de la connaissance des relations mathématiques entre les paramètres du système. Ce modèle peut être acquis soit par le mécanisme d'ordonnancement causal [IWA 86], soit par la théorie de graphe bipartie [BLA 06].
- Modélisation graphique causale : Elle consiste à ordonner causalement les connaissances fonctionnelles du procédé sous forme d'un graphique. Ce dernier comprend des nœuds qui représentent les variables significatives de la surveillance et des arcs liant les nœuds et simulant les phénomènes physiques [DEK 86].

Les méthodes basées sur la modélisation causale permettent une concentration sur certaines relations et empêche l'actualisation de tous les résidus.

2.3.2.2.3- Surveillance par reconnaissance des formes

Les méthodes de surveillance par reconnaissance des formes permettent de produire une information à partir de l'analyse d'un ensemble de mesures (continues ou discrètes) effectuées sur le système à des états de fonctionnement bien connus. L'information produite va être exploitée par la suite pour prendre une décision. Ces méthodes peuvent servir la surveillance en deux cas à savoir la reconnaissance des formes et la prédiction.

Généralement ces méthodes sont bien exploitées pour servir les deux fonctions de surveillance : diagnostic et pronostic.

On distingue trois techniques de reconnaissances des formes. La première technique utilise les outils statistiques classiques de discrimination basée sur les outils de la probabilité. Les deux autres sont les réseaux de neurones et la logique floue qui sont des méthodes d'intelligence artificielle.

2.3.2.2.3.1- Reconnaissances des formes par utilisation des outils statistiques

Ces techniques de reconnaissance des formes consistent à décider à quelle classe parmi M classes ($\alpha_1, \alpha_2, \dots, \alpha_M$) appartient le vecteur d'entrée x . Une application d qui associe un scalaire i à chaque vecteur d'entrée x est utilisée :

$d(x) = i$, où x est associé à la classe α_i ; avec $i=1, \dots, M$.

Les probabilités $Pr(\alpha_i)$ des classes $\alpha_1, \alpha_2, \dots, \alpha_M$ sont connues. On distingue deux cas possibles:

- Cas appelé monde fermé : toutes les classes sont bien connues. Ceci signifie que la somme de toutes les probabilités est égale à un :

$$\sum_{i=1}^M Pr(\alpha_i) = 1 \quad (2.28)$$

- Cas appelé monde ouvert : toutes les classes ne sont pas connues. Dans ce cas, une probabilité α_0 associée à une classe de rejet en distance est utilisée pour remplir l'insuffisance d'information sur le problème. Cette classe remplace implicitement toutes les autres classes non considérées par l'utilisateur. Ceci signifie la relation suivante :

$$\sum_{i=0}^M Pr(\alpha_i) = 1 \quad (2.29)$$

La qualité de la décision des classes d'affectation de x est quantifiée par plusieurs méthodes et heuristiques. Parmi lesquelles, la méthode « coût de décision $C(i, \alpha_j)$ », qui définit le coût de décider α_i quand α_j ($i, j=0, \dots, M$) est la vraie classe de x [DUD 73].

La décision finale est examinée par la méthode de détermination de risque moyen de décision R dont on prospectera à inspirer la valeur minimale [FUK 90].

2.3.2.2.3.2- Reconnaissances des formes par utilisation de la logique floue

Introduite par Zadeh² en 1965, la logique floue est un ensemble de théories mathématiques qui permet de représenter et manipuler de connaissances imparfaites, imprécises, incertaines et incomplètes. Elle permet de traiter des systèmes d'une grande complexité qui présentent des facteurs qualitatifs et quantitatifs. Elle est bien efficace lorsqu'elle intervient dans des problèmes qui manipulent de connaissances imparfaites.

Les classes sont représentées par des sous-ensembles flous. Cette méthode consiste à définir une fonction d'appartenance qui donne le degré d'appartenance $\lambda_i(x)$ de chaque vecteur x à la classe α_i .

La méthode de classification par utilisation de la logique floue comprend deux phases : la construction des fonctions d'appartenance et la définition des règles de décision. Le résultat

² Lotfi Askar Zadeh, né le 4 février 1921 à Bakou en Azerbaïdjan, est un scientifique connu pour ses travaux en informatique et en automatique. En 1965, il introduit la théorie des ensembles flous dont les applications industrielles sont nombreuses.

de cette approche consiste à attribuer à chaque vecteur x un ensemble des degrés d'appartenance à toutes les classes $(\lambda_1(x), \dots, \lambda_M(x))$.

L'utilisation de cette approche dans le domaine de surveillance touche notamment le diagnostic et l'aide à la décision. Néanmoins, elle peut être exploitée pour servir les autres fonctions de surveillance. Cette méthode procure un outil puissant pour soutenir d'une façon automatique des actions incertaines [BOU 96].

2.3.2.2.3.3- Reconnaissances des formes par utilisation des réseaux de neurones

Les réseaux de neurones artificiels, inspirés en 1890 par W. James³, sont des outils qui simulent le fonctionnement des neurones biologiques. Ces outils permettent certaines fonctions du cerveau, comme, par l'exemple, la mémorisation associative, l'apprentissage, le travail en parallèle, etc. Néanmoins les neurones formels n'ont pas toutes les aptitudes et les performances des neurones biologiques, comme l'activation membranaire, la répartition de synapses, etc. Pour ces raisons, les réseaux de neurones artificiels sont toujours loin d'avoir l'efficacité d'un cerveau.

Les réseaux de neurones sont généralement utilisés pour la surveillance des systèmes complexes, pour lesquels il est dur de modéliser mathématiquement les paramètres de système ainsi que les interactions entre eux afin de connaître les défauts des systèmes. Pour la surveillance, ces outils permettent la résolution des problématiques liées aux fonctions détection, diagnostic et pronostic. En effet, nous distinguons deux fonctionnalités des réseaux de neurones pour la surveillance : la prédiction et la reconnaissance des formes. Pour la première, les réseaux de neurones rassemblent à un approximateur universel qui fournit un modèle sous la forme d'une boîte noire du système. La deuxième consiste à attribuer un ensemble des données (quantifiables et/ou qualifiables) à une classe d'appartenance.

L'avantage important des réseaux de neurones par rapport aux autres outils de surveillance est leur capacité d'apprendre et de généraliser des connaissances à des entrées inconnues. Aussi, ces outils sont ductiles vu qu'ils peuvent être également implémentés informatiquement et en circuits électroniques accédant l'éventualité d'un traitement temps réel.

2.3.2.2.3.3.1- Éléments de base des réseaux de neurones

Un neurone biologique est une cellule qui comprend un corps cellulaire et un noyau (Fig. 2.9). Le corps cellulaire se ramifie tout en formant les dendrites. Ces dernières permettent l'acheminement de l'information de l'extérieur vers le soma (le corps du neurone) où elle sera traitée. Si le potentiel d'action dépasse un seuil mémorisé, le corps cellulaire répond par un stimuli (signal réponse). Le signal produit par le neurone traverse ultérieurement l'axone (unique) pour être envoyé ensuite vers des autres neurones de cerveau.

La communication entre deux neurones est assurée par l'espace intercellulaire qui est constitué de quelques dizaines d'Angströms. L'espace intercellulaire se localise entre l'axone

³ William James (11 janvier 1842 à New York - 26 août 1910 à Chocorua dans le New Hampshire) est un psychologue et philosophe américain.

du neurone afférent et les dendrites du neurone efférent. On appelle la jonction des deux neurones synapse.

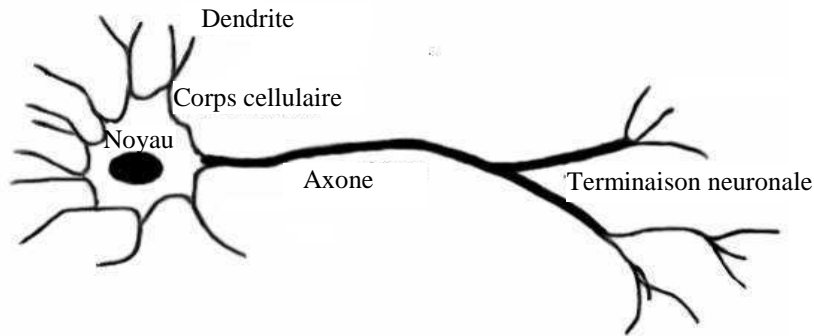


Fig. 2.9- Neurone biologique

La traduction d'un neurone biologique à un neurone artificiel est présentée par la figure 2.10.

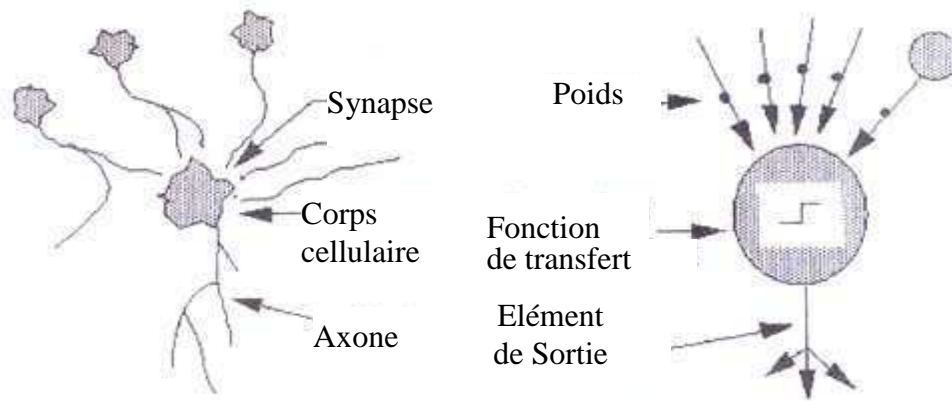


Fig. 2.10- Traduction d'un neurone biologique en neurone artificiel

En effet, chaque neurone artificiel présente un processeur élémentaire qui reçoit les informations provenant de neurones amont. A chacune de ces entrées un poids w est associé. Il signifie la force de la connexion. Chaque neurone est spécifié par une seule sortie qui s'étend ensuite pour être exploitée par les neurones aval Fig. 2.11.

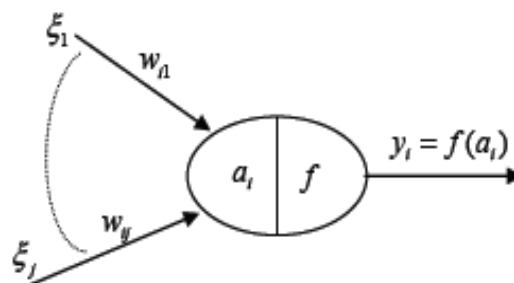


Fig. 2.11- Neurone formel

Par analogie avec le neurone biologique, le comportement du neurone artificiel se compose de deux phases :

- La phase d'activation : elle permet le calcul de la somme pondérée des entrées selon l'équation 2.30.

$$a_i = \sum_j w_{ij} \xi_j \quad (2.30)$$

- La phase de calcul de la valeur de l'état du neurone : A partir du résultat de l'équation 2.30, une fonction de transfert calcule la valeur de l'état du neurone. Cette valeur sera transmise aux neurones aval.

Il existe de nombreuses formes possibles pour la fonction de transfert. Les plus courantes formes des fonctions de transfert sont présentées sur la figure 2.12. Contrairement aux neurones biologiques dont l'état est binaire, la majorité des fonctions de transfert pour les neurones artificiels sont continues. Ceci permet d'offrir une infinité de valeurs possibles comprises dans l'intervalle $[0, +1]$ ou $[-1, +1]$.

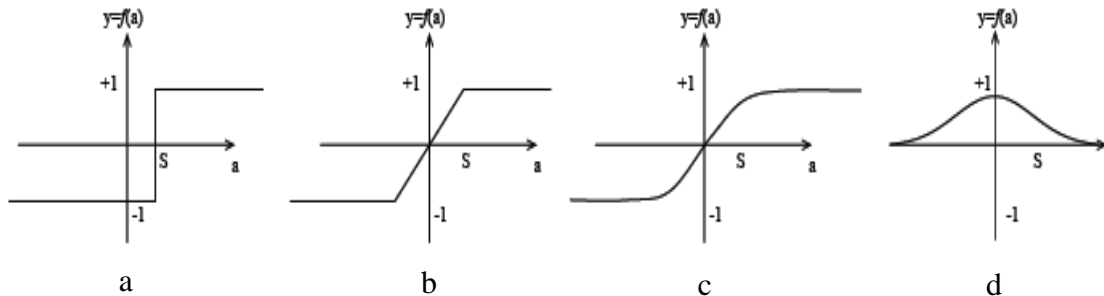


Fig. 2.12- Courantes formes de fonctions de transfert pour le neurone artificiel

- a) fonction à seuil du neurone,
- b) linéaire par morceaux du modèle Adaline,
- c) sigmoïde d'un réseau Perceptron Multi Couches,
- d) gaussienne du réseau RFR .

Avant d'implanter un réseau de neurones artificiels à un phénomène de surveillance, il faut passer par deux phases essentielles qui sont : l'apprentissage et le test. La phase d'apprentissage consiste à faire apprendre le réseau les alliances entre les paramètres d'entrée et de sortie. Durant cette phase, les entrées et les sorties sont connues. L'apprentissage est donc une phase primordiale et très importante pour qu'un réseau de neurones soit performant. La seconde phase permet d'évaluer l'aptitude des résultats obtenus lors de la phase d'apprentissage à généraliser d'autres connaissances qui présentent une suite de la base de données utilisée précédemment (les mêmes variables d'entrée et de sortie).

Dans le sous-paragraphe suivant, nous allons focaliser notre point d'intérêt sur l'apprentissage des réseaux de neurones artificiels.

2.3.2.2.3.3.2- Apprentissage des Réseaux de neurones artificiels

Pour notre cerveau, l'apprentissage est une phase très importante et complexe. Durant cette phase des modifications mineures ou majeures, selon la tâche à apprendre, s'exécutent entre les connexions des neurones. Le cerveau converge alors vers un résultat souhaité par l'être humain lui-même ou par son entourage, par exemple l'apprentissage d'un enfant à apprendre l'écriture, ou encore l'apprentissage pour s'adapter aux usages et aux coutumes d'un nouveau pays.

La mémoire humaine présente la capacité de cerveau à souvenir les expériences passées et les règles déjà apprises. Le cerveau humain fonctionne en exploitant le principe d'association. Cela signifie qu'une information incomplète ou imprécise (bruitée) conduit à résoudre un problème.

Similairement aux neurones biologiques, l'apprentissage des réseaux de neurones artificiels est une phase qui accède à la détermination ou la modification des paramètres représentatifs du réseau, afin de s'adapter à un comportement souhaité. Depuis la première règle d'apprentissage de Hebb⁴ en 1949, énormes algorithmes d'apprentissage ont été apparus.

En effet, selon les réseaux de neurones et son objectif, nous distinguons deux types d'apprentissage :

- Apprentissage supervisé : l'information est fournie. Elle se représente sous la forme d'une valeur ou d'un vecteur Y de sortie (appelé cible ou sortie désirée). Dans ce cas, le réseau de neurones doit associer au vecteur d'entrée x . L'apprentissage consiste alors à modifier les paramètres du réseau de neurones dans le but de minimiser l'erreur entre la sortie estimée par le réseau de neurones et la sortie réelle.
- Apprentissage non supervisé : L'objectif est d'inspirer les paramètres du réseau selon des critères donnés. En effet, les données ne possèdent pas d'informations sur une sortie désirée. Ce type d'apprentissage est exploitable lors d'une étude de classification des données.

2.3.2.2.3.3.3- Architectures neuronales les plus utilisées en surveillance

Concernant l'application des réseaux de neurones en surveillance industrielle, trois architectures neuronales sont souvent utilisées [KOI 94], [COU 07] : le Perceptron Multi Couche (PMC), le Réseau de neurones à Fonctions de base Radiales (RFR) et la carte de Kohonen.

Les RFR peuvent se montrer le plus performants en classification et en identification de processus que les deux autres à condition d'inspirer les paramètres adéquats. Néanmoins, le PMC est plus efficace que la carte de Kohonen et peut avoir dans des applications particulières des résultats proches ou meilleurs que ceux des RFR. Par contre, le réseau de Kohonen est moins efficace que les deux premières ; cependant ses aptitudes d'auto-adaptation (apprentissage non supervisé) sont très appréciées [KOI 94].

Les architectures de réseaux de neurones les plus souvent exploitées en surveillance s'avèrent donc être [KOI 94] :

- le Réseau à Fonctions de base Radiales (RFR),
- le Perceptron Multi Couches (PMC).

Dans la suite, nous allons présenter le principe de fonctionnement de chacune d'elles.

2.3.2.2.3.3.3.1- Perceptron Multi Couches (PMC)

Le Perceptron Multi Couches, en anglais: Multi Layer Perceptron (MLP), est un réseau de neurones à sens unique organisé en L couches. La première couche issue les entrées (données

⁴ La règle d'apprentissage de Hebb établie par Donald Hebb est une règle d'apprentissage des réseaux de neurones artificiels dans le contexte de l'étude d'assemblées de neurones. Cette règle suggère que lorsque deux neurones sont excités conjointement, cela renforce le lien qui les unit.

ou informations). La dernière produit les sorties (résultat) du PMC. Les entrées et les sorties des autres couches ne sont pas visibles ; c'est pour ça qu'elles s'appellent les couches cachées. Chaque neurone d'une couche est complètement connecté à tous les neurones de la couche suivante. Généralement, chaque neurone a une entrée supplémentaire (biais) comme le montre la figure 2.13. Les biais ont l'objectif de simuler les bruits influant le comportement de système et les paramètres non pris en considération.

L'apprentissage d'un PMC consiste à modifier les poids des connexions entre les couches, afin d'avoir une réponse proche à une sortie désirée.

Plusieurs algorithmes d'apprentissage pour les PMC ont été publiés parmi lesquels nous citons : la règle d'apprentissage de Hebb, Règle de Delta [WID 60], La rétropropagation, etc.

L'algorithme de rétropropagation de gradient est le plus utilisé pour les PMC. Il consiste à calculer le gradient de l'erreur de chaque neurone du réseau tout en suivant le sens inverse du réseau (de la dernière couche vers la première). Cet algorithme se fonde sur la descente de gradient qui a pour but de converger de manière itérative vers des valeurs optimisées des poids synaptiques. Ce résultat peut refléter un minimum local au niveau de la couche ou neurone et idéalement, un minimum global d'une fonction de coût qui optimise tous les poids synaptiques de du réseau.

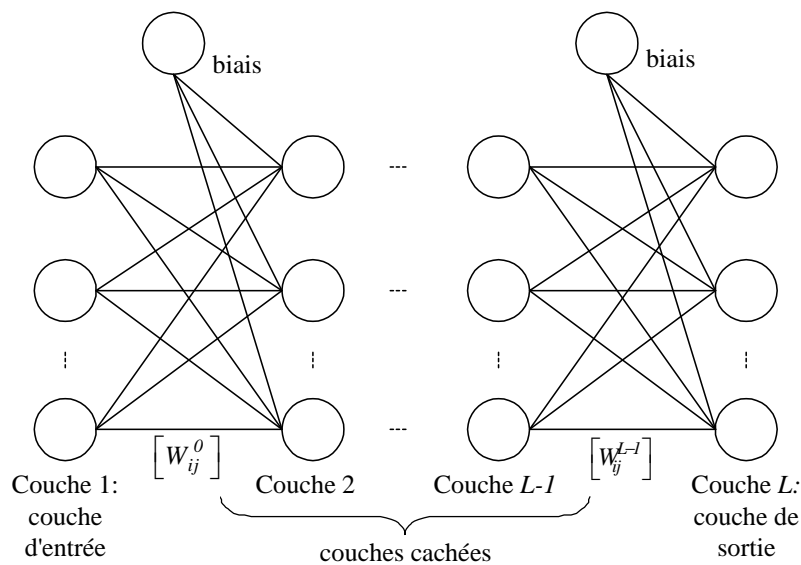


Fig. 2.13- un réseau de neurones PMC à L couches, complètement connecté et biaisé

Où :

W_{ij}^l = poids du neurone i de la couche l vers le neurone j de la couche $l+1$; $l \in [1, 2, \dots, L]$.

2.3.2.2.3.3.2- Réseau à Fonctions de base Radiales (RFR)

Les Réseaux de neurones à Fonctions de base Radiales RFR sont des réseaux à propagation avant. L'utilisation de ces réseaux a commencé aux années soixante-dix du siècle précédent par [HAR 71], [AGT 74], etc., afin de résoudre des problèmes d'interpolation multi variables. Ensuite, ces réseaux ont été plus approfondis théoriquement [POW 87], [HON 09], etc. Les RFR ont été amplifiés ultérieurement pour couvrir d'autres domaines d'application dont les

principaux sont : la prédiction de l'évolution des systèmes dynamiques [BRO 88], [CHN 96], [ROJ 00], etc., et la classification de phonèmes [REN 96] [NAB 99], etc.

Les *RFR* sont formulés d'une couche d'entrée, une seule couche cachée composée des neurones à fonctions noyaux de base radiales $\phi_j(\|\cdot\|)$ et une couche de sortie (Fig. 2.14).

Deux paramètres sont attribués à $\phi_j(\|\cdot\|)$: le vecteur de référence μ_j appelé centre ou prototype et la dimension σ_j du champ d'influence appelé rayon d'influence. La réponse, donc, est en fonction de la distance du vecteur d'entrée x au vecteur prototype μ_j , et de la taille du champ d'influence σ_j :

$$\phi_j(x) = \phi_j(\|x - \mu_j\|, \sigma_j) \quad (2.31)$$

La fonction $\Phi \rightarrow \phi_j(x)$ doit être définie positive.

La fonction de base la plus utilisée est la gaussienne. Elle s'exprime par :

$$\phi_j(x) = \exp\left(-\frac{1}{2\sigma_j^2}(x - \mu_j)(x - \mu_j)^t\right) \quad (2.32)$$

Où :

σ_j^2 est la variance de la cellule.

La fonction gaussienne possède la même décroissance dans toutes les directions de l'espace. Sa courbe est présentée alors sous la forme d'une hypersphère.

L'expression de sortie d'un réseau RFR est la suivante :

$$y(x) = \sum_{j=1}^{M_k} w_j \phi_j(x) \quad (2.33)$$

Où w_j est le poids de la connexion entre le $j^{ème}$ neurone radial et le neurone de sortie. Chaque neurone de sortie est lié à M_k neurones radiaux. La couche de sortie permet de répondre aux objectifs du réseau à travers des neurones linéaires.

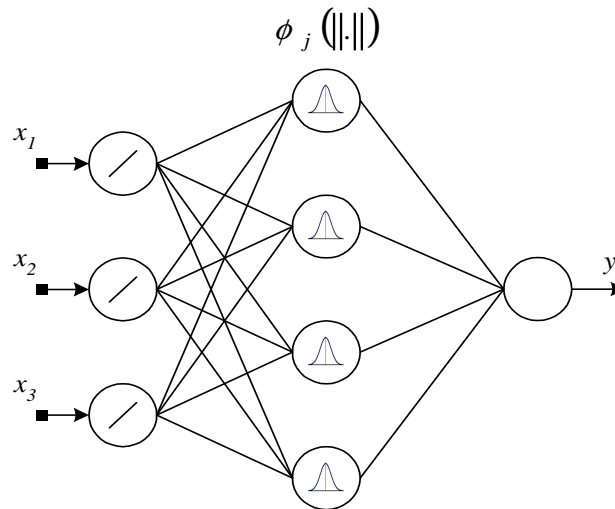


Fig. 2.14- Architecture d'un RFR.

2.3.2.2.3.3.4- Réseaux de neurones temporels

La surveillance industrielle nécessite un mécanisme performant qui permet la prise en compte d'un certain historique des signaux capteurs. La détection prédictive qui a l'objectif de reconnaître la nature de dégradation en fonction de l'évolution d'un signal, et le pronostic dans le cas de vouloir prédire l'évolution d'un signal à l'instant $t+\Delta t$, sont des fonctions qui nécessitent l'intervention de notion temporelle afin d'amplifier la performance de la surveillance dans certaines applications.

On distingue, dans la littérature, deux classes des Réseaux de neurones temporels :

- Représentation spatiale du temps : ce sont des réseaux de neurones temporels avec une représentation externe du temps. L'information temporelle est transformée en une information spatiale, le problème à résoudre s'agit de reconnaître une forme. Nous mentionnons ici l'architecture NETtalk [SEG 86], l'architecture TDNN [SEG 89], etc. Des retards permettent la mémorisation des données d'entrée pendant une certaine durée τ_i . Dans ce cas, le vecteur d'entrée comporte les données à l'instant t et ceux des instants $t-\tau_i$ (avec $i=1, \dots, N$). Ceci provoque l'obtention d'une fenêtre temporelle de taille $N+1$. L'avantage principal de cette technique est l'aptitude d'utiliser les architectures de réseaux de neurones statiques sans aucune modification.
- Représentation dynamique du temps : Ce sont des réseaux qui traitent le temps à l'intérieur du réseau. Les informations peuvent se mémoriser d'une manière implicite par la récurrence des connexions, tels que l'architecture de Moakes [SEG 94], l'architecture de M.W. Mak [MAK 95], l'architecture RRFR [ZEM 03], etc. L'inconvénient majeur de ces réseaux récurrents est la lenteur ainsi que la difficulté d'apprentissage. Les informations peuvent se mémoriser, aussi, d'une manière explicite par le biais d'implantation des retards au niveau des connexions tel que l'architecture ATDNN [DAY 93].

2.3.3- Synthèse et choix des outils de surveillance adéquats pour notre approche

Les outils et les méthodes de surveillance industrielle sont très variés. En effet, nous avons

distingué deux grandes classes qui sont la surveillance avec modèle et la surveillance sans modèle. La différence entre les deux est qu'on effectue, pour la première classe, la comparaison entre l'état estimé et l'état théorique du système. Par contre, pour la deuxième classe, on compare les paramètres estimés aux paramètres théoriques du système. Ces deux approches sont en réalité complémentaires. L'approche sans modèle, de type reconnaissance des formes (surveillance sans modèle), peut s'appuyer sur des paramètres mesurés par des capteurs mais aussi sur des paramètres estimés par une méthode à base de modèle.

Vu que les systèmes de production sont généralement complexes et que leurs représentations par un modèle mathématique est difficile, les méthodes de surveillance sans modèle sont les plus efficaces pour les systèmes industriels.

En ce qui concerne notre architecture « Surveillance indirecte suite à une commande robuste », notre objectif pour la surveillance est de pronostiquer, suivre et détecter les états jugés anormaux du système. En effet, les deux fonctions objectives sont le pronostic et le suivi-détection.

Notre architecture concerne les SPA qui présentent une grande classe des systèmes industriels. Ces derniers sont généralement complexes. Pour ceci, nous proposons d'exploiter des méthodes de surveillance sans modèle. En effet, les outils que nous allons utiliser sont :

- Pour la fonction pronostic : les réseaux de neurones qui présentent un outil très performant pour la prédiction des états des systèmes complexes dont la modélisation mathématique est difficile vu l'impossibilité de prendre en compte tous les paramètres du système et leurs interactions. Partant de la littérature, nous pouvons distinguer deux architectures efficaces pour la prédiction qui sont : les RFR et les PMC. En effet, les RFR ont montré une performance plus appréciée vu leurs capacités de classer les entrées avant de prédire les états. Nous avons vu aussi que les récurrences donnent aux réseaux de neurones l'aptitude d'avoir une mémoire dynamique interne. Pour ceci, nous allons exploiter dans la suite de cette thèse, pour la fonction pronostic, l'outil RFR. Nous allons essayer d'améliorer l'architecture (RRFR), proposée dans [ZER 03], qui utilise les récurrences au niveau de la couche d'entrée.
- Pour la fonction suivi-détection : les Réseaux de Petri Temporels à Objets qui présentent un outil performant pour le suivi et la détection des symptômes de défaillance. Cet outil permet de satisfaire nos besoins en terme de modélisation des paramètres temporels et non temporels en vue de suivre en premier lieu les paramètres de la loi de commande estimés par la fonction pronostic et de détecter les anomalies du système en deuxième lieu. Ce type de RdP a montré son efficacité à résoudre la problématique de suivi-détection [COM 91], [ZAM 97] [DAS 03], [MIN 04], etc.

2.4- Conclusion

Dans ce chapitre, nous nous sommes intéressés au choix des outils et des méthodes de surveillance et de modélisation de commande performants pour notre architecture.

Nous avons commencé par une étude bibliographique concernant les principaux outils de modélisation des lois de commande des systèmes industriels. Nous avons distingué, à partir de la littérature, deux types des outils de modélisation qui peuvent résoudre notre

problématique de commande : Automate à états finis et RdP. Vu la performance de l'outil des RdP en termes de modélisation des synchronisations, des parallélismes, des conflits et des partages de ressources, cet outil a été choisi pour être utilisé dans la suite de cette thèse. L'étude comparative entre les différents types des RdP justifie notre choix qui est les RdP à Intervalles. En effet, les RdP à Intervalles ont la capacité de modéliser des contraintes temporelles et non temporelles. Ceci permet une modélisation adéquate et complète de la commande robuste hybride (passive et active) bi-critères (flux et qualité) de notre architecture.

Dans un second lieu, nous avons concentré notre intérêt sur le choix des outils et des méthodes qui vont servir les fonctions du modèle surveillance. Nous avons distingué deux classes des outils et des méthodes pour la surveillance des systèmes industriels : avec modèle et sans modèle. Vu la complexité des systèmes industriels qui ne peuvent pas, généralement, être modélisé mathématiquement vu l'impossibilité de prendre en compte tous les paramètres du système et leurs interactions, nous avons choisi les méthodes sans modèle. En effet pour la fonction pronostic, notre choix consiste à exploiter, dans la suite de cette thèse, les réseaux de neurones RFR qui présentent un outil très approprié pour la prédiction des états des systèmes complexes. Concernant la fonction suivi-détection, les Réseaux de Petri Temporels à Objets sont efficaces à résoudre la problématique de suivi-détection [COM 91], [ZAM 97] [DAS 03], [MIN 04]. Cet outil répond aux obligations imposées par notre architecture en termes de modélisation des paramètres temporels (flux) et non temporels (qualité).

Dans le chapitre suivant, nous allons s'intéresser au développement de notre approche de modélisation de la commande robuste hybride (passive et active) bi-critères (flux et qualité) de notre architecture.

Chapitre 3

Modélisation et analyse d'une commande robuste multicritères

Chapitre 3

Modélisation et analyse d'une commande robuste multicritères

Table des matières du chapitre 3

3.1-	Introduction.....	81
3.2-	Définition d'une terminologie.....	81
3.3-	Modèles RdP de commande pour la génération de robustesse : approche monocritère.....	82
3.3.1-	Mise en situation	82
3.3.2-	Modélisation d'un paramètre	83
3.3.2.1-	Modélisation d'un paramètre flux.....	83
3.3.2.2-	Modélisation d'un paramètre qualité	83
3.3.3-	Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop	84
3.3.3.1-	Modélisation d'un circuit élémentaire	84
3.3.3.1.1-	Modélisation d'un circuit élémentaire de flux Cp	84
3.3.3.1.1.1-	Cas où $b_{ai} \geq a_{pi+1}$	84
3.3.3.1.1.2-	Cas d'un goulot d'étranglement : $b_{ai} < a_{p.i+1}$	86
3.3.3.1.2-	Modélisation d'un circuit élémentaire de qualité Cq	88
3.3.4-	Aspect formel pour le modèle de robustesse hybride monocritère	90
3.3.4.1-	Aspect formel pour les circuits élémentaires	90
3.3.4.2-	Aspect formel pour un système de production.....	94
3.4-	Modèles RdP de commande pour la génération de robustesse d'une ressource : approche bi-critère	97
3.4.1-	Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop.....	97
3.4.1.1-	Cas des variables qualitatives unitaires en parallèle	97
3.4.1.2-	Cas des variables qualitatives unitaires en série	99
3.4.2-	Aspect formel pour le modèle de robustesse hybride bi-critère d'une ressource	101
3.4.2.1-	Aspect formel pour le cas des variables qualitatives unitaires en parallèle	101
3.4.2.2-	Aspect formel pour le cas des variables qualitatives unitaires en série.....	102
3.5-	Modèles RdP de commande pour la génération de robustesse d'un Système de production: approche bi-critère	102
3.5.1-	Modélisation d'une commande robuste hybride bi-critère d'un système de production de type flow-shop	102
3.5.2-	Aspect formel pour le cas des variables qualitatives unitaires en série.....	104
3.5.3-	Application.....	105
3.6-	Conclusion	108

3.1- Introduction

Dans la littérature, les travaux portant sur la modélisation, par Réseaux de Petri, des commandes robustes se sont intéressés à un seul type de robustesse (passive ou active). Ils n'ont pas intervenu un modèle hybride qui génère à la fois la robustesse passive et active. C'est dans ce cadre que se situe notre travail dans ce chapitre.

Comme il est annoncé précédemment, l'outil qu'on va exploiter est les réseaux de Petri à Intervalles.

L'objectif de ce chapitre est de développer un modèle de commande hybride modélisant, à la fois, la robustesse passive et active et permettant l'observation de la robustesse des ressources et l'interprétation de type de la robustesse globale des systèmes de production manufacturière face à des perturbations temporelles (flux) et non temporelles (qualité). L'idée sous-jacente est de définir des modèles locaux hybrides, relatifs aux ressources, permettant la conservation des propriétés spécifiées de sous-systèmes par l'exploitation de la redondance des marges de la robustesse entre passive et active. A partir de ces modèles locaux, nous pouvons observer, généraliser et constater la robustesse globale de système.

Dans une première étape, nous présenterons une terminologie relative à notre contexte. Nous présenterons, dans la seconde étape, le modèle flux et le modèle qualité qui traitent, chacun, à la fois la robustesse active et passive. Dans la troisième partie, nous nous intéresserons au modèle de commande robuste d'une ressource. La quatrième partie, présente le modèle finale pour un système de production.

3.2- Définition d'une terminologie

Définition 3.1 [TEL 08a]

« Un circuit élémentaire de production » C_p est défini comme un ensemble des machines ordonnées influant, directement ou indirectement, par la variation de leurs temps de production, une des spécificités du système opérant.

Définition 3.2 [TEL 08a]

Un paramètre qualité est nommé « variable expliquée » si est seulement s'il dépend de variations des autres paramètres mesurés en amont.

Définition 3.3 [TEL 08a]

Un paramètre qualité est nommé « variable explicative » si est seulement s'il participe avec des autres paramètres, par sa variation, à influencer une variable expliquée.

Définition 3.4 [TEL 08a]

Un paramètre qualité est nommé « variable expliquée composée » si est seulement s'il dépend au moins de la variation d'une autre variable expliquée mesurée en amont.

On note que :

- une variable expliquée peut être explicative pour des autres variables mesurées en aval,
- si nous sommes dans un environnement certain, la relation entre la variable expliquée et celles explicatives serait déterminée par une formule mathématique,

- si nous sommes dans un environnement incertain, la relation entre la variable expliquée et celles explicatives serait déterminée par une relation floue (logique floue) ou par utilisation des outils d'estimation statistique (régression linéaire, régression non linéaire, etc.).

Nous considérons un système de production manufacturière de type flow-shop dont nous tendons à maintenir n paramètres qualité. Le but est que le système conserve les propriétés de qualité du produit face à des perturbations prévues ou imprévues.

Définition 3.5 [TEL 08a]

Un paramètre de qualité forme un «circuit élémentaire de qualité » Cq si et seulement si il est une variable expliquée.

Nous désignons par RM la « robustesse modulaire » d'un circuit élémentaire de type flux ou qualité.

Définition 3.6 [TEL 08b]

La robustesse modulaire est définie comme la capacité à maintenir localement les propriétés spécifiques d'un circuit élémentaire en présence de variations ou d'incertitudes prévues ou imprévues dues à des perturbations internes ou externes afin de conserver la robustesse globale de système de production.

3.3- Modèles RdP de commande pour la génération de robustesse : approche monocritère

Les systèmes concernés sont les systèmes flexibles de production Manufacturière de type flow-shop. Dans ce paragraphe, nous nous intéressons à la problématique de robustesse de flux et celle de qualité chacune à part.

3.3.1- Mise en situation

Considérons un système de production manufacturière S constitué de N ressources de transformation de matière. Chaque ressource est caractérisée par le temps de production élémentaire (Tm_i); avec $i \in \mathbb{N}$ et $i \in \{1, 2, \dots, N\}$. A la sortie de chaque ressource, n_i paramètres qualité seront mesurés. S est constitué de $H.Cq$ et $K.Cp$. Nous définissons l'intervalle de temps $Ip_i = [a_{pi}, b_{pi}]$ (respectivement $Ia_i = [a_{ai}, a_{pi}] \cup [b_{pi}, b_{ai}]$) comme la marge de robustesse passive (respectivement de robustesse active) relative au temps de production élémentaire de la ressource R_i ; avec $a_{pi} \geq a_{ai}$ et $b_{pi} \leq b_{ai}$. Vu que la fonction temps est monotone croissante et que nous ne nous intéressons pas à la corrélation entre la qualité et le flux nous pouvons suggérer que $Ia_i = [b_{pi}, b_{ai}]$.

De même, nous attribuons l'intervalle $Ip_{ij} = [a_{pi,j}, b_{pi,j}]$ (respectivement $Ia_{ij} = [a_{ai,j}, a_{pi,j}]$ et $Ia_{s_{ij}} = [b_{pi,j}, b_{ai,j}]$) comme la marge de robustesse passive (respectivement de robustesse active) de la variable $V_{i,j}$; où, $V_{i,j}$ est la variable explicative présentant le $j^{\text{ème}}$ paramètre qualité de la $i^{\text{ème}}$ ressource de S . avec $a_{pi,j} \geq a_{ai,j}$ et $b_{pi,j} \leq b_{ai,j}$; $j \in \mathbb{N}$ et $j \in \{1, 2, \dots, n_i\}$.

Hypothèse 3.7

$\forall i$ et j , $V_{i,j}(t)$ est croissante non définie.

Hypothèse 3.8

Les variations des variables temporelles ou non temporelles ne dépassent pas les marges de robustesse active.

3.3.2- Modélisation d'un paramètre

La modélisation d'un paramètre flux ou qualité par un RdP à Intervalles dépend du nombre des intervalles de robustesse attribués à chaque paramètre. Chaque intervalle de robustesse (passive ou active) devrait avoir une place qui le modélise. En effet, Nous définissons à chaque type des paramètres une structure de parallélisme capable à prendre en compte les marges de robustesse active et passive.

3.3.2.1- Modélisation d'un paramètre flux

Chaque paramètre flux se modélise par une structure de parallélisme composée d'une transition et de deux places relatives respectivement à la robustesse passive et active [TEL 08b], [TEL 09a].

Les intervalles alloués aux deux places de la structure de parallélisme se repartissent comme suit (fig. 3.1) :

$I'p_i = [a_{pi}, b_{pi}]$: intervalle attribué à la place qui modélise la robustesse passive de R_i ,

$I'as_i = [a_{ai}, b_{ai}]$: intervalle attribué à la place qui modélise la robustesse active de R_i .

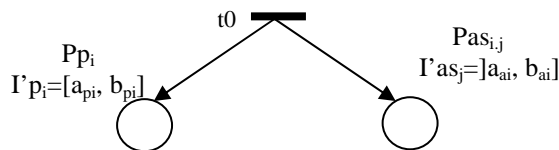


Fig. 3.1- Éléments d'une structure de parallélisme de modèle flux

3.3.2.2- Modélisation d'un paramètre qualité

Similairement au modèle d'un paramètre flux, le nombre des places de la structure de parallélisme du paramètre qualité dépend du nombre des intervalles de robustesse. En effet, chaque paramètre qualité se modélise par une structure de parallélisme composée d'une transition et de trois places relatives respectivement à la robustesse active inférieure, passive et active supérieure [TEL 08a], [TEL 09a].

Les intervalles alloués aux trois places de la structure de parallélisme se repartissent comme suit (fig. 3.2) :

$I'ai_{ij} = [a_{ai,j}, a_{pi,j}]$: intervalle attribué à la place qui modélise la robustesse active inférieure de $V_{i,j}$,

$I'p_{ij} = [a_{ai,j}, b_{pi,j}]$: intervalle attribué à la place qui modélise la robustesse passive de $V_{i,j}$,

$I'as_{ij} = [a_{ai,j}, b_{ai,j}]$: intervalle attribué à la place qui modélise la robustesse active supérieure de $V_{i,j}$.

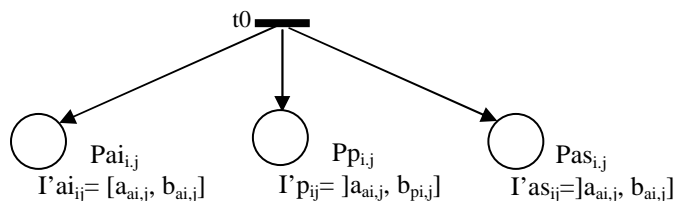


Fig. 3.2- Éléments d'une structure de parallélisme de modèle flux

3.3.3- Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop

Notre objectif, dans ce paragraphe, est de développer un concept méthodologique permettant la modélisation d'une loi de commande robuste prenant en compte les marges de robustesse passive et active d'un système de type flow-shop (au cas limite, un atelier d'assemblage). Après la mise en place des structures de parallélisme affectées pour les paramètres flux et qualité, dans le paragraphe précédent, nous commençons, le présent, par la définition d'un concept de modélisation exploitant les RDP à Intervalles, pour les circuits élémentaires de flux et de qualité. Puis, nous cumulons par la mise au point d'une méthodologie formelle (théorique) qui permet la synthèse de la nature de robustesse monocritère. Des exemples applicatifs seront illustrés.

3.3.3.1- Modélisation d'un circuit élémentaire

La modélisation d'une loi de commande robuste monocritère avec la prise en compte d'un seul type de robustesse (passive ou active) consiste à simuler chaque paramètre temporel (flux) [KAN 97], [COL 97], [BON 00], [JER 06] ou non temporel (qualité) [DHO 05a] par une seule place portant, selon le type de paramètre, l'intervalle attribué à la marge de robustesse (passive ou active). Notre objectif, dans ce paragraphe, est de présenter une modélisation performante des circuits élémentaires (de flux ou de qualité) permettant la mise en interaction des marges de robustesse passive et active. Ceci permet l'observation de type de robustesse au niveau des paramètres et des ressources d'où, par la mise d'un cadre formel, la constatation de type de la robustesse modulaire ; puis, dans une deuxième étape, la robustesse globale. En effet, les structures de parallélisme définies dans le paragraphe précédent, seront exploitées dans notre nouveau modèle hybride (passive et active) pour remplacer les places.

3.3.3.1.1- Modélisation d'un circuit élémentaire de flux Cp

Nous distinguons deux cas à traiter. Le premier s'agit d'une structure normale où quelles que soient R_i et R_{i+1} la condition suivante est satisfaite : $b_{ai} \geq a_{pi+1}$. Le deuxième répond au cas d'un goulot d'étranglement : $b_{ai} < a_{pi+1}$.

3.3.3.1.1.1- Cas où $b_{ai} \geq a_{pi+1}$

Pour une série des ressources répondant à la condition suivante : $b_{ai} \geq a_{pi+1}$, La modélisation d'une loi de commande modulaire robuste monocritère relative au $k^{\text{ème}}$ Cp_k permettant la prise en compte de la redondance de robustesse entre passive et active peut être présentée sous une forme primaire pyramidale ; $k \in \mathbb{N}$ et $k \in \{1, 2, \dots, K\}$. Le squelette d'un tel modèle de flux est formulé de Nsp_k structures de parallélisme [TEL 08b].

Avec :

$$Nsp_k = 2^{R_k} - 1 \quad (1.3)$$

Où :

k, r : indice de la $r^{\text{ème}}$ ressource de Cp_k , $R_{k,r}$; $r \in \mathbb{N}$ et $r \in \{1, 2, \dots, R_k\}$.

Exemple 3.1

La figure 3.3 illustre la modélisation d'une loi de commande modulaire robuste, sous la forme

pyramidale, relative à un circuit élémentaire de production composé de deux ressources dont les variations de temps de traitement de produit influent la grandeur spécifique Tce_1 d'un système opérant avec $b_{a_{i,j}} \leq a_{i,j+1}$.

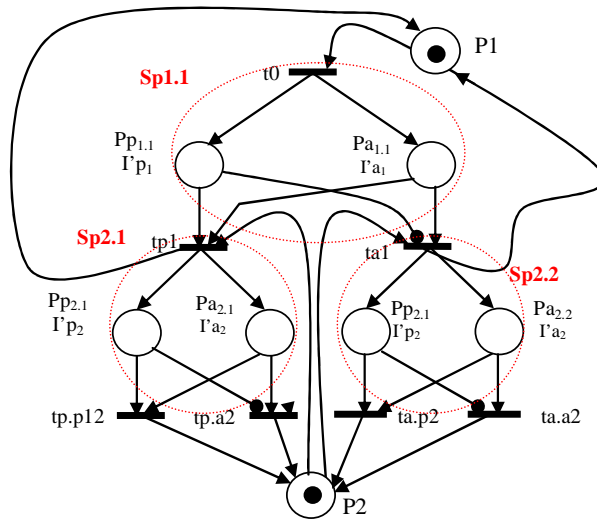


Fig. 3.3- Redondance de la robustesse (structure pyramidale) ; $b_{ai} \geq a_{i+1}$

où :

P1 : place modélisant la disponibilité de la ressource $R_{1,1}$,

P2: place modélisant la disponibilité de la ressource $R_{1,2}$.

Le principe d'évolution de ce modèle consiste à assurer, en premier lieu, la robustesse passive de la ressource. En effet, Le principe de fonctionnement du Réseau de Petri à Intervalles de la figure 3.3 se résume comme suit :

Suite au franchissement de transition $t0$, les places $Pp_{1,1}$ et $Pa_{1,1}$ se marquent chacune par un jeton. Si à $t \leq b_{p1,1}$ la transition $tp1.1$ s'est franchie, les places $Pp_{2,1}$ et $Pa_{2,1}$ seraient marquées et la ressource RI serait considérée robuste passivement. Si non, à partir de $t > b_{p1,1}$, seulement la place $Pa_{1,1}$ serait marquée. Le franchissement de la transition $ta1.1$ à $t \leq b_{a1,1}$ signifie que la ressource RI est robuste activement. $Sp2.1$ et $Sp2.2$ ont le même principe de fonctionnement.

Cette structure s'aperçoit être plus complexe lorsque le nombre des ressources d'un circuit élémentaire est grand (pour $R_k=6$; $Nsp_k=32$). Ceci implique des modèles compliqués à construire et à comprendre.

La structure parallèle consiste à fusionner les 2^r structures de parallélisme de la $r^{ème}$ ressource de Cp_k . En effet, sauf la première qui prend la structure de parallélisme précédemment définit, chaque ressource sera modélisée par une seule structure de parallélisme à deux transitions d'entrée [TEL 08b]. La figure 3.4 présente cette structure.

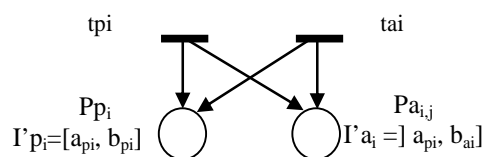


Fig. 3.4- Éléments d'une structure de parallélisme à deux transitions d'entrée (modèle flux)

Exemple 3.2

La figure 3.5 propose un modèle à une structure parallèle simplifiée et équivalente à la structure pyramidale.

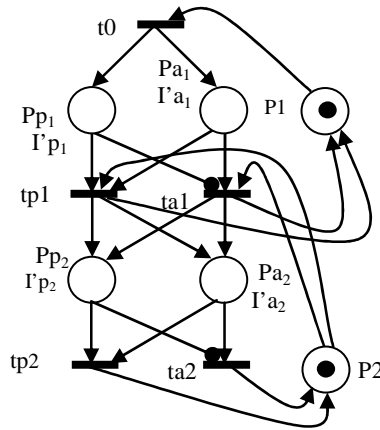


Fig. 3.5- Redondance de la robustesse de flux (structure parallèle) : $b_{ai} \geq a_{pi+1}$

Le principe de fonctionnement de cette structure parallèle est le même que celui de la structure pyramidale, sauf que le franchissement d'une de deux transitions de sortie de R_i provoque le marquage des places de la structure de parallélisme modélisant R_{i+1} .

3.3.3.1.1.2- Cas d'un goulot d'étranglement : $b_{ai} < a_{pi+1}$

Si nous admettons le même modèle du sous paragraphe précédent, dans le cas d'un goulot d'étranglement ($b_{ai} < a_{pi+1}$), un blocage de fonctionnement se manifesterait. Pour Ceci, nous proposons le lemme suivant :

Lemme 3.10 [TEL 08b]

Soient deux ressources chronologiquement successives R_i et R_{i+1} appartenant au circuit élémentaire Cp_k . Pour une structure pyramidale, Si ($b_{ai} < a_{pi+1}$) alors (chaque place des structures de parallélisme modélisant la robustesse passive de R_i (respectivement la robustesse active de R_i)) devrait être obligatoirement suivie par une place $Psp_{i,v}$ (respectivement $Psa_{i,v}$) indiquant le stock des produits intermédiaire en attente. L'intervalle statique associé à $Psp_{i,v}$ et $Psa_{i,v}$ est $Is_i = [0, +\infty[$; où l'indice i,v est associé à la $v^{ème}$ structure de parallélisme de la $i^{ème}$ ressource de Cp_k . Pour le cas d'une structure parallèle en enlève l'indice v .

Preuve :

Pour la structure pyramidale : Deux cas peuvent se présenter.

- *Premier cas : $b_{ai} \geq a_{pi+1}$*

Quelque soit l'âge de marquage t_i de $Pa_{i,v}$ nous ne rencontrons jamais le cas où une marque serait morte.

- *deuxième cas : $b_{ai} < a_{pi+1}$*

Il existe des cas où une marque de $Pa_{i,v}$ serait morte à l'instant $t_i = b_{ai}$ parce que la place $Pa_{i+1,v}$ est encore marquée. Alors que les commandes de $t = Pp_{i+1,v}^0$ ou de $t = Pa_{i+1,v}^0$ ne seraient

validées que si les marques des places modélisant la disponibilité de la ressource $R_{i,j+1}$ sont marquées.

Pour la structure parallèle : on admet le même raisonnement sauf qu'on enlève l'indice v .

Exemple 3.3

La figure 3.6 (respectivement la figure 3.7) présente la modélisation d'une loi de commande modulaire robuste, sous la forme pyramidale (respectivement sous la forme parallèle), relative à un circuit élémentaire de production composé des deux ressources dont les variations de temps de traitement de produit influent la grandeur spécifique Tce_1 d'un système opérant avec $b_{ai} < a_{i+1}$.

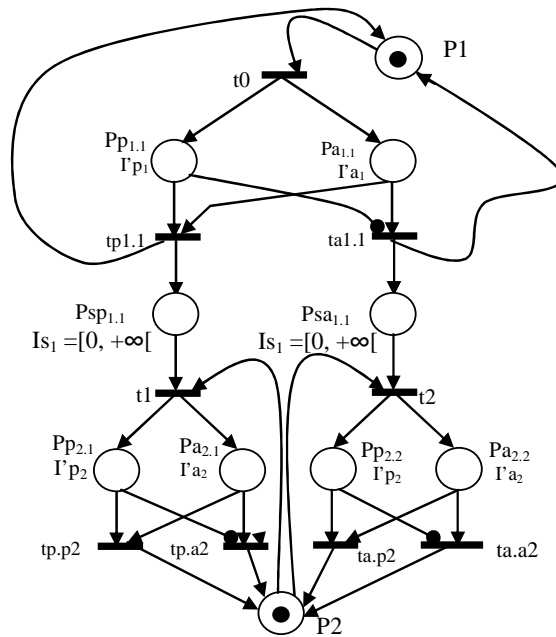


Fig. 3.6- Redondance de la robustesse de flux (structure pyramidale) ; $b_{ai} < a_{i+1}$

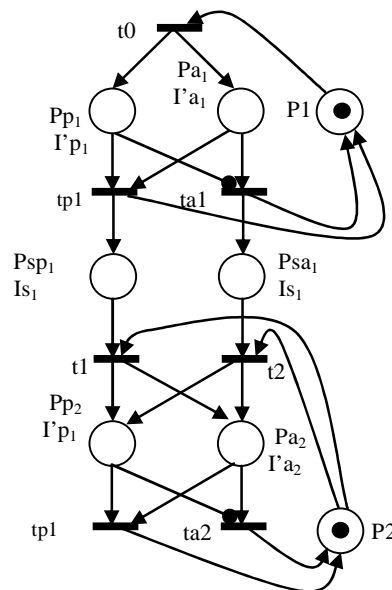


Fig. 3.7- Redondance de la robustesse de flux (structure parallèle) ; $b_{ai} < a_{i+1}$

3.3.3.1.2- Modélisation d'un circuit élémentaire de qualité Cq

Exploitant le même principe de modélisation d'un Cp_k , une loi de commande modulaire robuste hybride monocritère (robustesse active et passive) relative au $h^{ème} Cq_h$ peut être présentée sous une forme primaire pyramidale ; $h \in \mathbb{N}$ et $h \in \{1, 2, \dots, H\}$. Le squelette d'un tel modèle de qualité est formulé de Nsp_h structures de parallélisme [TEL 08a].

$$Nsp_h = \frac{3^{S_h} - 1}{2} \tag{2.3}$$

Où :

h, s : l'indice de la $s^{ème}$ variable explicative de $Cq_h, V_{h,s}$; $s \in \mathbb{N}$ et $s \in \{1, 2, \dots, S_h\}$.

Exemple 3.4

La figure 3.8 présente la modélisation d'une loi de commande modulaire robuste, sous la forme pyramidale, relative à un circuit élémentaire de qualité composé de deux ressources dont chacune comprend une seule variable explicative. La variation de chacune influe la variable expliquée spécifiant Cq_1 .

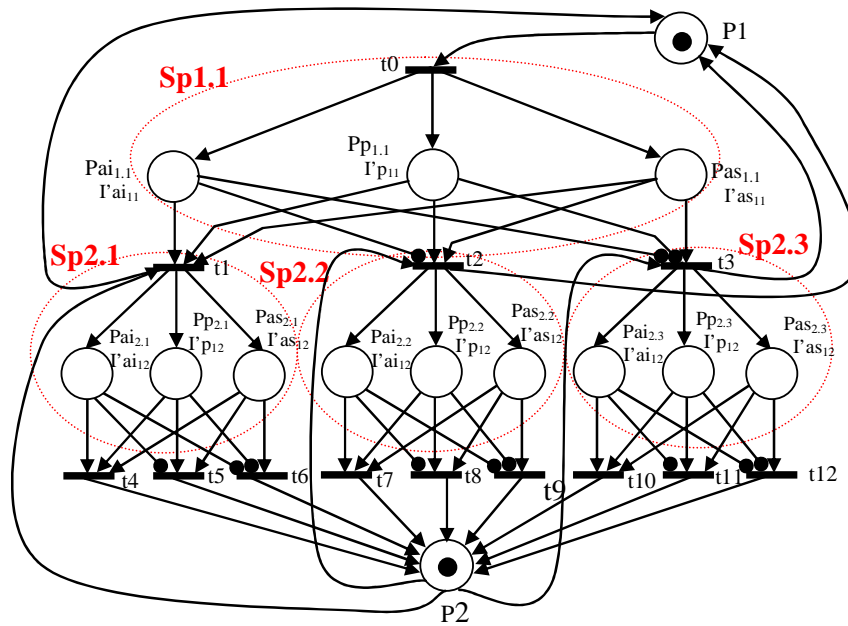


Fig. 3.8- Redondance de la robustesse de qualité (structure pyramidale)

où :

$P1$: place modélisant la disponibilité de la première ressource,

$P2$: place modélisant la disponibilité de la deuxième ressource,

$\left. \begin{matrix} \Gamma ai_{hs} : \\ \Gamma as_{hs} : \\ \Gamma p_{hs} : \end{matrix} \right\} h, s$ est l'indice de la $s^{ème}$ variable explicative de Cq_h .

Remarque : chaque intervalle portant l'indice hs correspond à un indice ij défini au départ.

Le principe de fonctionnement du modèle de la figure 3.8 est le même du modèle flux à structure pyramidale.

Similairement au modèle flux, nous proposons une structure parallèle qui permet la réduction de la complexité du RdP à Intervalles modélisant la loi de commande robuste hybride monocritère de qualité (pour $R_h=6$; $Ns_{q_h}=364$).

La structure parallèle consiste à associer les 3^s structures de parallélisme de la $r^{ème}$ variable explicative de Cq_h en une seule appelée structure de parallélisme croisée. Seule la structure d'entrée, modélisant la première ressource prend la structure de parallélisme classique définie pour le modèle pyramidale. Pour les autres, chacune sera modélisée par une seule structure de parallélisme croisée à trois transitions d'entrée [TEL 08a]. La figure 3.9 présente cette structure.

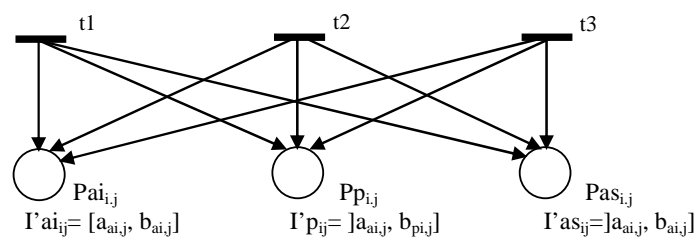


Fig. 3.9- Éléments d'une structure de parallélisme à trois transitions d'entrée (modèle qualité)

Exemple 3.5

La figure 3.10 présente le modèle parallèle simplifiée équivalent à la structure pyramidale de la figure 3.8.

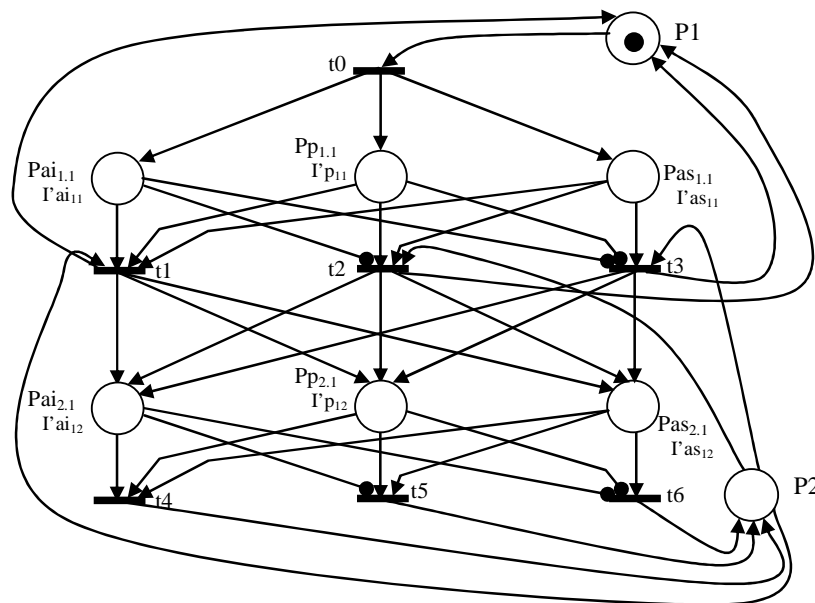


Fig. 3.10- Redondance de la robustesse (structure parallèle) : qualité

Le principe de fonctionnement de cette structure parallèle de qualité est le même que celui de la structure parallèle de flux, sauf que le franchissement d'une des trois transitions de sortie de

$V_{h,s}$ provoque le marquage des places de la structure de parallélisme modélisant $V_{h,s+1}$.

Remarque : il arrive des cas où une ressource comprend plusieurs variables explicatives qui participent à la variation d'une variable expliquée d'un Cq_h . Dans ce cas, la disponibilité de cette ressource est modélisée par trois arcs sortant chacune d'une transition de sortie de la structure modélisant la dernière variable expliquée de cette ressource pour être liés par la suite à la place de disponibilité. Ensuite, trois autres arcs lient cette place aux transitions d'entrée de la première variable expliquée de cette ressource.

Exemple 3.6

La figure 3.11 présente la modélisation de cette situation pour une ressource R_i qui comprend deux variables explicatives participant à la variation d'une variable expliquée.

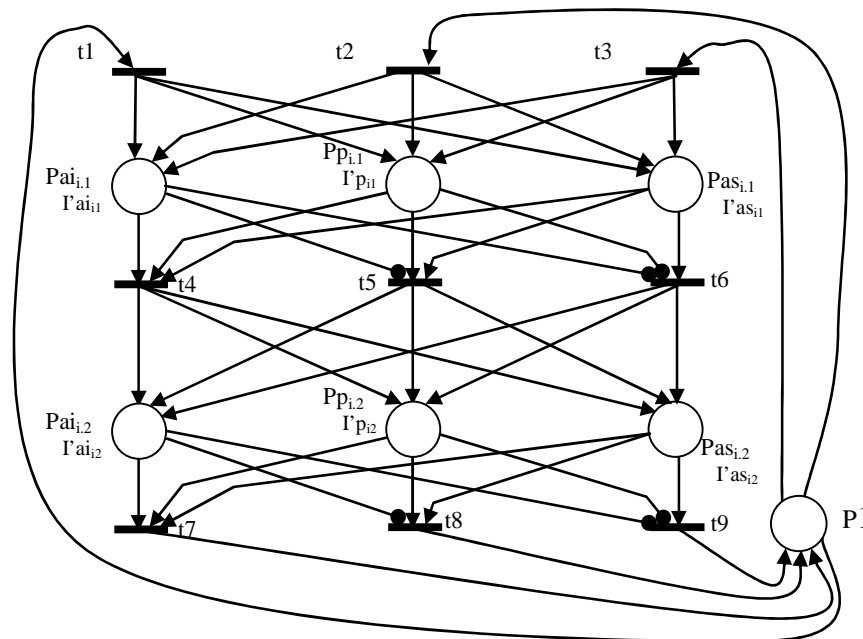


Fig. 3.11- Modélisation de la disponibilité pour une ressource à deux variables explicatives participant à la variation d'une variable expliquée

3.3.4- Aspect formel pour le modèle de robustesse hybride monocritère

Après la mise d'un concept de modélisation dans le sous paragraphe précédent, nous présentons, dans le présent, l'aspect formel qui nous permet de mettre en évidence la théorie relative aux modélisations de la robustesse hybride monocritère des circuits élémentaires déjà présentées. Pour ceci, nous présentons, dans la suite, un ensemble des définitions, des lemmes et théorèmes qui vont servir pour l'observation de la robustesse au niveau des paramètres et ressources et la constatation de nature de robustesse modulaire.

3.3.4.1- Aspect formel pour les circuits élémentaires

Définition 3.11 [TEL 08a], [TEL 08b]

Nous désignons par P^- (respectivement P^+) la fonction déterminant les places modélisant la robustesse passive (respectivement active) des variables temporelles ou non temporelles

appartenant au réseau spécifiant un circuit élémentaire de flux ou de qualité.

P^- et P^+ sont définies comme suit :

Modèle flux :

$$P^- : P \rightarrow EP^-$$

$$p_{h,e} \mapsto P^-(p_{h,e}) = Pp_{h,e}$$

$$P^+ : P \rightarrow EP^+$$

$$p_{h,e} \mapsto P^+(p_{h,e}) = Pa_{h,e}$$

Modèle qualité :

$$P^- : P \rightarrow EP^-$$

$$p_{k,d} \mapsto P^-(p_{k,d}) = Pp_{k,d}$$

$$P^+ : P \rightarrow EP^+$$

$$p_{k,d} \mapsto P^+(p_{k,d}) = Pa_{k,d}$$

Où :

P : Ensemble des places du réseau.

EP^- : Ensemble des places du réseau modélisant la robustesse passive des ressources,

EP^+ : Ensemble des places du réseau modélisant la robustesse active des ressources,

k, e : désigne l'indice de la $e^{\text{ème}}$ structure de parallélisme de la $k^{\text{ème}}$ ressource de Cp ,

h, d : désigne l'indice de la $d^{\text{ème}}$ structure de parallélisme de la $h^{\text{ème}}$ variable explicative de Cq .

Définition 3.12 [TEL 08a], [TEL 08b], [NAB 08]

Pc est nommé parcours de commande. Il est défini comme un chemin orienté qui relie les transitions et les places marquées modélisant la nature de la robustesse dans l'ordre, tout en commençant de celle modélisant l'entrée d'un circuit élémentaire vers celles de sortie. Si, durant l'évolution du réseau modélisant un circuit élémentaire, toutes les places de Pc sont marquées et les transitions de sortie relatives sont franchies, Pc serait appelé « marqué ».

Note : Au cas de synchronisation des jetons (séjournés aux places modélisant un paramètre qualité ou flux), seule la place qui a la plus petite borne supérieure de l'intervalle alloué serait hypothétiquement considérée marquée.

Nous désignons par :

$Pc_{k,p}$: parcours de commande numéro p de Cp_k ; $p \in \mathbb{N}$ et $p \in \{1, \dots, 2^{R_k}\}$,

$Pc_{h,q}$: parcours de commande numéro q de Cq_h ; $q \in \mathbb{N}$ et $q \in \{1, \dots, 3^{S_h}\}$,

$Pc_{P_{k,p}}$: l'ensemble des places subordonnées de $Pc_{k,p}$ et appartenant à EP^- ,

$Pc_{A_{k,q}}$: l'ensemble des places subordonnées de $Pc_{k,p}$ et appartenant à EP^+ ,

$Pc_{P_{h,q}}$: l'ensemble des places subordonnées de $Pc_{h,q}$ et appartenant à EP^- ,

$Pc_{A_{h,q}}$: l'ensemble des places subordonnées de $Pc_{h,q}$ et appartenant à EP^+ .

Lemme 3.13 [TEL 08a], [TEL 08b], [NAB 08]

Soit $P_{C_{k,p}}$ (respectivement $P_{C_{h,q}}$) un parcours de commande marqué d'un circuit élémentaire C_{p_k} (respectivement C_{q_h}). Si ($P_{C_{A_{k,p}}} = \emptyset$ (respectivement $P_{C_{A_{h,q}}} = \emptyset$)) alors (C_{p_k} (respectivement C_{q_h}) est robuste passivement).

Preuve

Tant que :

- toutes les variables explicatives de C_{p_k} (respectivement C_{q_h}) ont conservé leurs spécificités lors des opérations de transformation tout en respectant les marges de robustesse passive,
- la marge de robustesse passive de chaque variable temporelle (respectivement non temporelle) est choisie au départ sous des contraintes globales traduisant la robustesse passive de la variable expliquée de C_{p_k} (respectivement C_{q_h}).

La définition 1.2 est applicable.

Lemme 3.14 [TEL 08a], [TEL 08b]

Soit $P_{C_{k,p}}$ (respectivement $P_{C_{h,p}}$) un parcours de commande marqué d'un circuit élémentaire C_{p_k} (respectivement C_{p_h}). Si ($P_{C_{A_{k,p}}} \neq \emptyset$ (respectivement $P_{C_{A_{h,p}}} \neq \emptyset$)) alors (C_{p_k} (respectivement C_{p_h}) est robuste activement).

Preuve :

Tant qu'une (au moins) des variables explicatives de $P_{C_{k,p}}$ (respectivement $P_{C_{h,p}}$) est robuste activement alors que les autres sont robustes passivement, les propriétés spécifiées de $P_{C_{k,p}}$ (respectivement $P_{C_{h,q}}$) ne peuvent être maintenues qu'au prix d'un calcul total ou partiel de la conduite.

la définition 1.3 est applicable.

Théorème 3.15 [TEL 08a], [TEL 08b]

Il n'existe qu'un seul parcours de commande $P_{C_k^-}$ (respectivement $P_{C_h^-}$) accédant à la robustesse passive de C_{p_k} (respectivement de C_{p_h}). Les autres accèdent à la robustesse active.

Définition 3.16 [TEL 08a], [TEL 08b], [NAB 08]

$P_{C_k^+}$ (respectivement $P_{C_h^+}$) est nommé ensemble des parcours de commande robuste activement. Pour une structure modélisant C_{p_k} (respectivement C_{p_h}) remembrant K (respectivement H) ressources, l'ensemble $P_{C_k^+}$ (respectivement $P_{C_h^+}$) assemble $NP_{C_k^+}$ (respectivement $NP_{C_h^+}$) parcours avec :

$$NP_{C_k^+} = 2^{R_k} - 1 \quad (3.3)$$

$$NP_{C_h^+} = 3^{S_h} - 1 \quad (3.4)$$

Exemple 3.7

Dans cet exemple nous présentons l'application de ces définitions, ces lemmes et ce théorème sur les modélisations présentées dans les exemples : exemple 3.1 (modèle flux) et exemple 3.8 (modèle qualité).

- **Modèle flux : (exemple 3.1 ; figure 3.3)**

Ensemble des places du réseau modélisant la robustesse passive des ressources :

$$EP_i^- = \{Pp_{1,1}, Pp_{2,1}, Pp_{2,2}\},$$

Ensemble des places du réseau modélisant la robustesse active des ressources :

$$EP_i^+ = \{Pa_{1,1}, Pa_{2,1}, Pa_{2,2}\}.$$

Parcours de commande :

$$Pc_{1,1} = (t0, Pp_{1,1}, tp1.1, Pp_{2,1}, tp.p1.2),$$

$$Pc_{1,2} = (t0, Pp_{1,1}, tp1.1, Pa_{2,1}, tp.a1.2),$$

$$Pc_{1,3} = (t0, Pa_{1,1}, ta1.1, Pp_{2,2}, ta.p1.2),$$

$$Pc_{1,4} = (t0, Pa_{1,1}, ta1.1, Pa_{2,2}, ta.a1.2).$$

Nombre des parcours de commande active :

$$NPC_i^+ = 2^2 - 1 = 3$$

Parcours de commande passive :

$$Pc_i^- = \{Pc_{1,1}\}$$

Parcours de commande active :

$$Pc_i^+ = \{Pc_{1,2}, Pc_{1,3}, Pc_{1,4}\}$$

Le circuit élémentaire du réseau modélisé par la figure 3.3 ne peut être robuste passivement que si et seulement si le parcours de commande passive $Pc_i^- = \{Pc_{1,1}\}$ est marqué.

- **Modèle qualité : (exemple 3.4 ; figure 3.8)**

Ensemble des places du réseau modélisant la robustesse passive des ressources :

$$EP_i^- = \{Pp_{1,1}, Pp_{2,1}, Pp_{2,2}, Pp_{2,3}\}.$$

Ensemble des places du réseau modélisant la robustesse active des ressources :

$$EP_i^+ = \{Pai_{1,1}, Pas_{1,1}, Pai_{2,1}, Pas_{2,1}, Pai_{2,2}, Pas_{2,2}, Pai_{2,3}, Pas_{2,3}\}.$$

Parcours de commande :

$$Pc_{1,1} = (t0, Pai_{1,1}, t1, Pai_{2,1}, t4),$$

$$Pc_{1,2} = (t0, Pai_{1,1}, t1, Pp_{2,1}, t5),$$

$$Pc_{1,3} = (t0, Pai_{1,1}, t1, Pas_{2,1}, t6),$$

$$Pc_{1,4} = (t0, Pp_{1,1}, t2, Pai_{2,2}, t7),$$

$$Pc_{1,5} = (t0, Pp_{1,1}, t2, Pp_{2,2}, t8),$$

$$Pc_{1,6} = (t0, Pp_{1,1}, t2, Pas_{2,2}, t9),$$

$$Pc_{1,7} = (t0, Pas_{1,1}, t3, Pai_{2,3}, t10),$$

$$Pc_{1,8} = (t0, Pas_{1,1}, t3, Pp_{2,3}, t12),$$

$$Pc_{1,9} = (t0, Pas_{1,1}, t3, Pas_{2,3}, t11).$$

Nombre des parcours de commande active :

$$NPc_i^+ = 3^2 - 1 = 8$$

Parcours de commande passive :

$$Pc_i^- = \{Pc_{1,5}\}.$$

Parcours de commande active :

$$Pc_i^+ = \{Pc_{1,1}, Pc_{1,2}, Pc_{1,3}, Pc_{1,4}, Pc_{1,6}, Pc_{1,7}, Pc_{1,8}, Pc_{1,9}\}.$$

Le circuit élémentaire du réseau modélisé par la figure 3.3 ne peut être robuste passivement que si et seulement si le parcours de commande passive $Pc_i^- = \{Pc_{1,5}\}$ est marqué.

3.3.4.2- Aspect formel pour un système de production

L'extension des propriétés et des principes de modélisation développés, nous amène à s'étendre pour dégager les lemmes suivants :

Lemme 3.17 [TEL 08a], [TEL 08b], [NAB 08]

Soit un système automatisé de production S constitué de K Cp (respectivement H Cq). Si ($\forall k \in \{1, 2, \dots, K\}$) (respectivement $h \in \{1, 2, \dots, H\}$), $Pc_{Ak,p} = \emptyset$ (respectivement $Pc_{Ah,p} = \emptyset$) alors (S est robuste passivement).

Preuve

Même raisonnement de la preuve de lemme 3.13.

Lemme 3.18 [TEL 08a], [TEL 08b], [NAB 08]

Soit un système automatisé de production S constitué de K Cp (respectivement H Cq). Si (il \exists pour $k \in \{1, 2, \dots, K\}$) (respectivement $h \in \{1, 2, \dots, H\}$), au moins ($Pc_{Ak,p} \neq \emptyset$ (respectivement $Pc_{Ah,p} \neq \emptyset$)) alors (S est robuste activement).

Preuve

Même raisonnement de la preuve de lemme 3.14.

On peut déduire des lemmes 3.17 et 3.18 le théorème suivant :

Théorème 3.19 [TEL 08a], [TEL 08b], [NAB 08]

Un système de production S est robuste passivement si et seulement si tous les circuits élémentaires sont robustes passivement. Si non, il est robuste activement.

Exemple 3.8 (cas flux)

Cas d'un atelier d'assemblage :

Nous traitons ici le cas d'un système d'assemblage qui hérite la caractéristique d'un atelier flow-shop : chaque poste de travail est chargé d'une seule transformation de matière.

Considérons un système automatisé de production composé de trois circuits élémentaires C_1 , C_2 et C_3 . C_1 sert à fabriquer un premier produit semi-fini A. C_2 permet la production d'un deuxième produit semi-fini B. C_3 permet dans un premier lieu d'assembler A et B par une

ressource $R_{3,1}$, puis d'effectuer une transformation mécanique par $R_{3,2}$. C_1 est composé de deux ressources de transformation en séries $R_{1,1}, R_{1,2}$ dont $b_{a1,2} > a_{p1,1}$. C_2 est composée de deux ressources de transformation successives $R_{2,1}, R_{2,2}$ dont $a_{p2,1} \geq b_{a2,2}$. Finalement nous indiquons que $a_{p3,1} \geq b_{a3,2}$. La figure 3.12 illustre la modélisation de cet exemple.

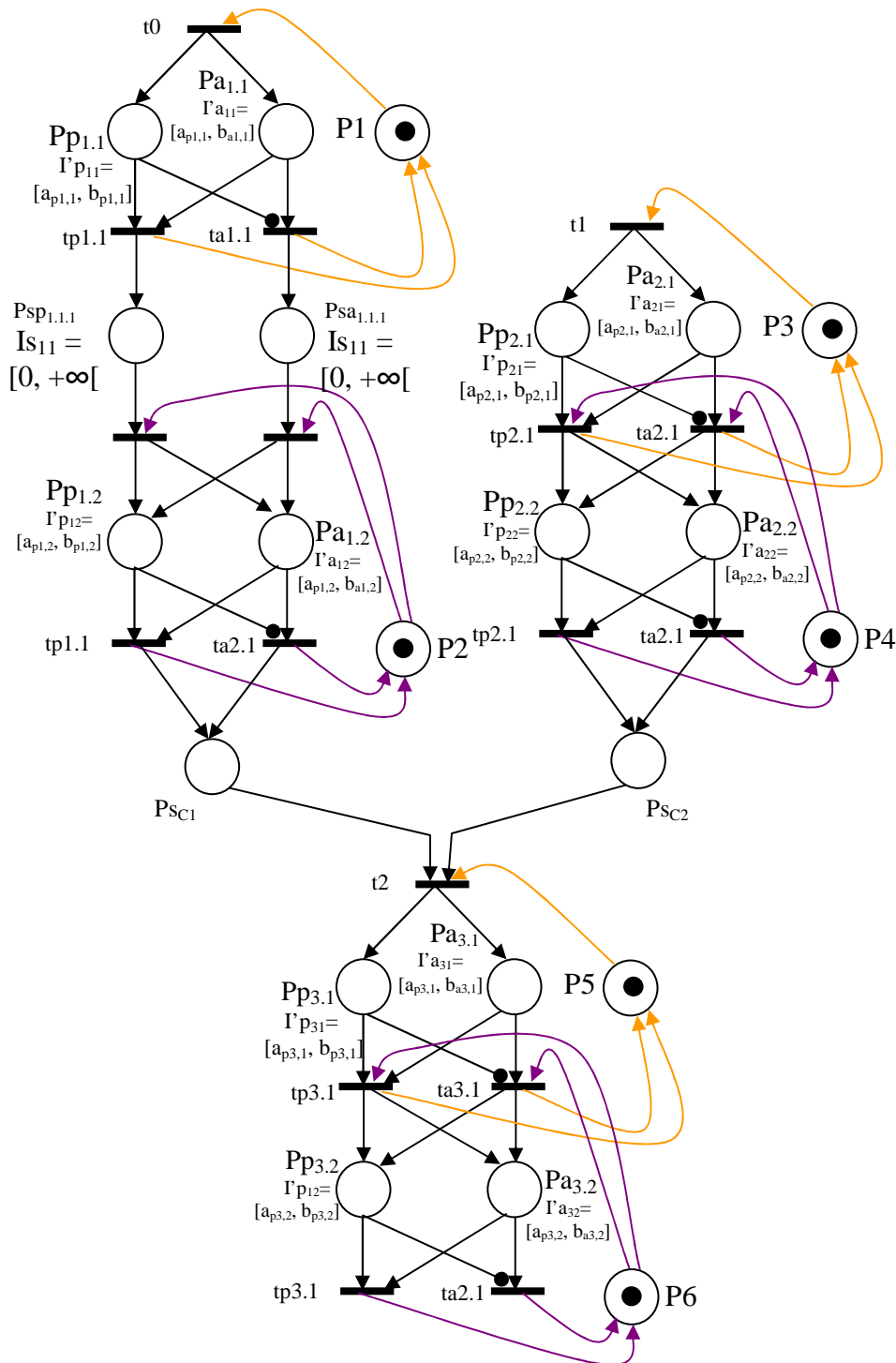


Fig. 3.12- Redondance de la robustesse d'un système de production (cas flux : atelier d'assemblage)

Le découplage de deux flux de A et B est garanti par une structure de synchronisation constituée de deux places PSC1 et PSC2 modélisant respectivement les deux stocks des produits

A et B et d'une transition $t2=P^{\circ}SC1=P^{\circ}SC2$. Pour que S soit robuste passivement, il faut que les parcours P_{Ck1} , P_{Ck2} et P_{Ck3} soient marqués.

Exemple 3.9 (cas qualité)

On opte, dans cet exemple, à présenter la modélisation d'une loi de commande robuste hybride de qualité d'un système de production S de type flow-shop. On considère que le système S comprend trois ressources dont chacune est caractérisée par un seul paramètre qualité (variable explicative). Les variables de la première ressource et de la deuxième forment un circuit élémentaire de qualité. Aussi, les variables de la deuxième et de la troisième forment le second circuit élémentaire de qualité de S. La figure 3.13 illustre la modélisation de cet exemple.

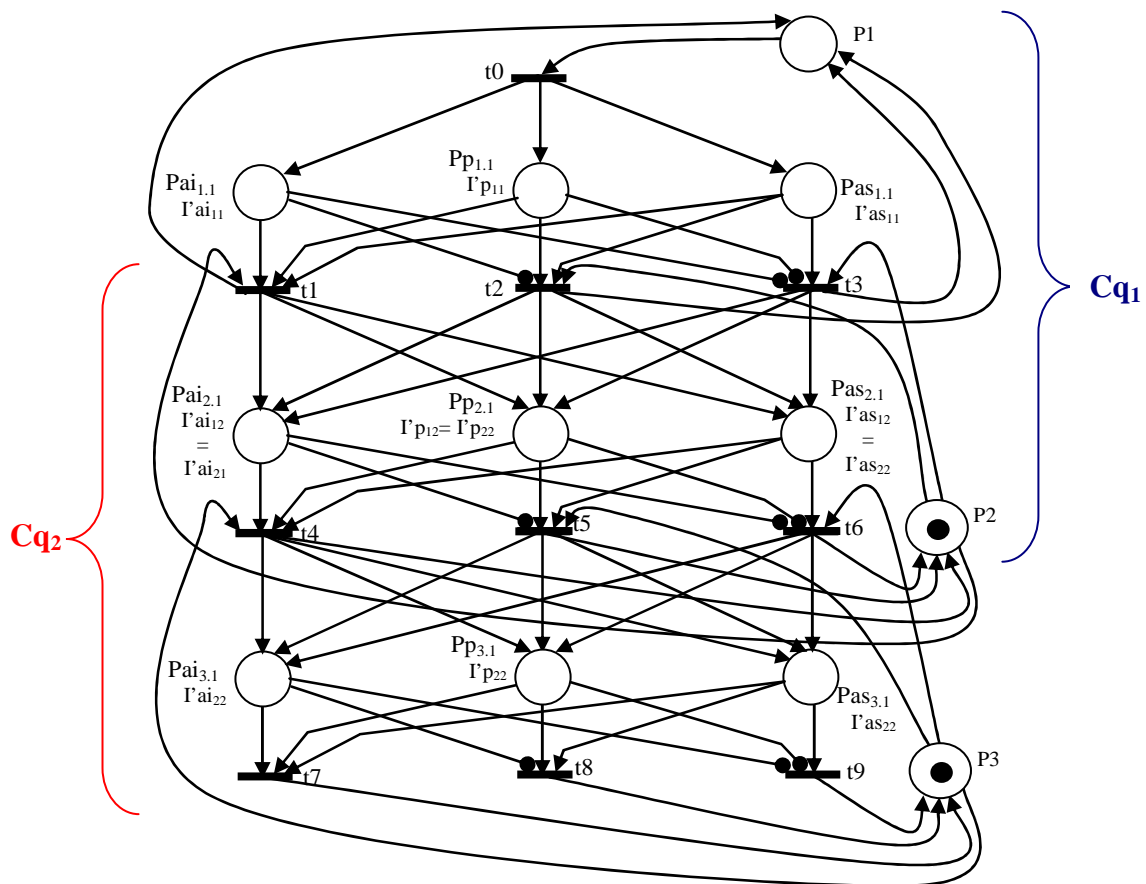


Fig. 3.12- Redondance de la robustesse d'un système de production (cas qualité)

On remarque que la variable de la deuxième ressource appartient aux deux circuits élémentaires Cq_1 et Cq_2 . Ceci implique la fusion des intervalles de robustesse de la deuxième ressource de Cq_1 et la première de Cq_2 : $I'_{ai12}=I'_{ai21}$, $I'_{p12}=I'_{p22}$ et $I'_{as12}=I'_{as22}$.

Le système ne peut être robuste passivement que si est seulement si Cq_1 et Cq_2 sont robustes passivement. Si non, c'est-à-dire si au moins un seul circuit est robuste activement, alors le système est robuste activement.

Autrement, Pour que S soit robuste passivement, il faut que les parcours P_{Ch1} et P_{Ch2} soient marqués.

S ne peut être robuste passivement que si et seulement si toutes les ressources sont robustes passivement.

3.4- Modèles RdP de commande pour la génération de robustesse d'une ressource : approche bi-critère

Comme nous avons précisé, une ressource R_i est caractérisée par une seule variable temporelle Tm_i et une ou plusieurs variables qualité. Dans ce qui suit, notre enchaînement méthodologique sera fondé sur la distinction entre deux cas qui sont :

- Cas des variables qualitatives unitaires en parallèle,
- Cas des variables qualitatives unitaires en série,

Définition 3.20 [TEL 09a]

Pour une ressource de production, on définit par « variables qualitatives unitaire en parallèle » l'ensemble des variables qualitatives interdépendantes instantanément qui influent par leurs variations la commandabilité des transitions de sortie des places modélisant la robustesse d'une variable temporelle.

Définition 3.21 [TEL 09a]

Pour une ressource de production, on définit par « variables qualitatives unitaire en série » l'ensemble des variables qualitatives successivement dépendantes, chacune de la précédente, et qui influent par leurs variations la commandabilité des transitions de sortie des places modélisant la robustesse d'une variable temporelle.

3.4.1- Modélisation d'une commande robuste monocritère d'un système de production de type flow-shop

3.4.1.1- Cas des variables qualitatives unitaires en parallèle

La modélisation de ce cas est présentée sous la forme d'une structure formée de n_{i+1} places d'entrée. La première est suivie d'une structure de parallélisme modélisant le paramètre flux (temporel). Les autres sont suivies chacune par une structure de parallélisme modélisant une variable qualité (non temporelle) [TEL 09a].

La synchronisation entre la structure de parallélisme modélisant la robustesse passive et active du paramètre temporel et l'ensemble des celles modélisant les robustesses des paramètres non temporelles (les variables qualité) est assurée à travers des canaux (places) d'échange d'informations [TEL 09a]. L'exemple suivant explique bien comment on modélise ce cas.

Exemple 3.10

Soit une ressource R_I d'un système de production S . R_I est caractérisée par le temps de production élémentaire Tm_I (variable temporelle), et deux variables qualitatives unitaires en parallèle $V_{I,1}$ et $V_{I,2}$.

Nous illustrons, par la figure 3.13, la modélisation de la loi de commande robuste hybride bi-

critère (flux-qualité) de la ressource R_l .

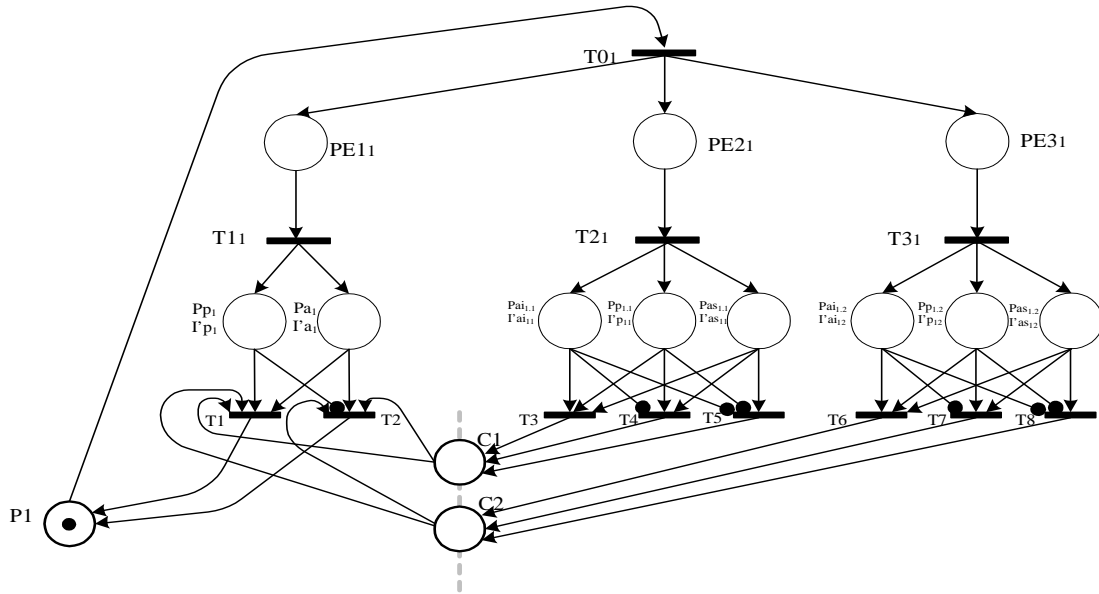


Fig. 3.13- Modèle d'une loi de commande robuste hybride bi-critère (flux-qualité) de la ressource R_l : cas des variables qualitatives unitaires en parallèle

Où :

$C1$ et $C2$: Place d'échange d'informations, elle présente un canal de communication entre la les places modélisant $V_{l,1}$ et $V_{l,2}$ et celles spécifiant l'aspect temporel de R_l .

$P1$: places modélisant la disponibilité de R_l .

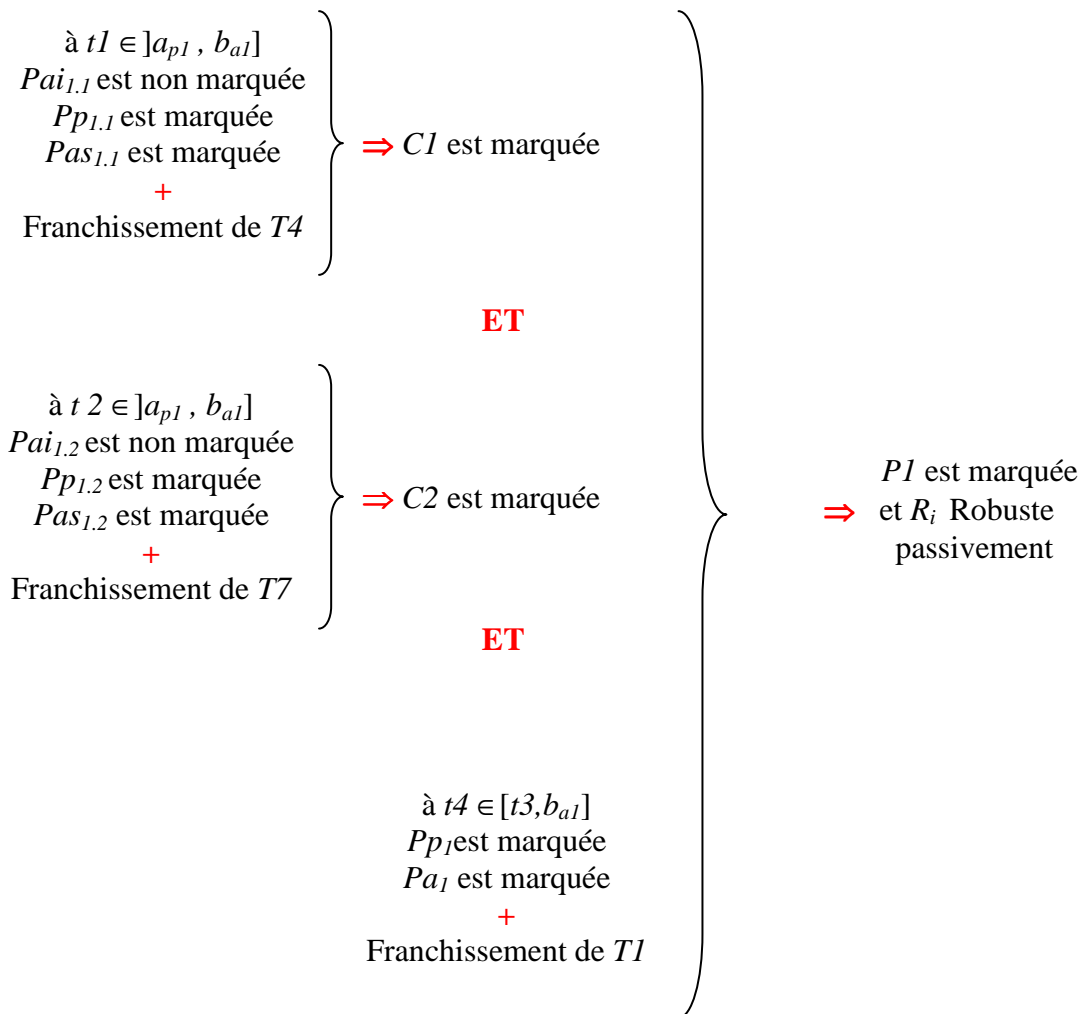
Le principe de fonctionnement de ce modèle est expliqué par le tableau 3.1 qui représente les différents cas d'évolution de ce modèle.

À	$t1 \in]a_{p1}, b_{a1}]$				$t2 \in]a_{p1}, b_{a1}]$				$t3 \in]\max\{t1,t2\}, b_{a1}]$			$t4 \in [t3, b_{a1}]$			
Cas	$P_{ai,1,1}$	$P_{pi,1,1}$	$P_{as,1,1}$	T.F	$P_{ai,1,2}$	$P_{pi,1,2}$	$P_{as,1,2}$	T.F	C1	C2	P_{p1}	P_{p2}	T.F	P1	Rob. Ri
1	1	1	1	T3	1	1	1	T6	1	1	1	1	T1	1	Active
2	1	1	1	T3	0	1	1	T7	1	1	1	1	T1	1	Active
3	1	1	1	T3	0	0	1	T8	1	1	1	1	T1	1	Active
4	0	1	1	T4	1	1	1	T6	1	1	1	1	T1	1	Active
5	0	1	1	T4	0	1	1	T7	1	1	1	1	T1	1	Passive
6	0	1	1	T4	0	0	1	T8	1	1	1	1	T1	1	Active
7	0	0	1	T5	1	1	1	T6	1	1	1	1	T1	1	Active
8	0	0	1	T5	0	1	1	T7	1	1	1	1	T1	1	Active
9	0	0	1	T5	0	0	1	T8	1	1	1	1	T1	1	Active
10	1	1	1	T3	1	1	1	T6	1	1	0	1	T2	1	Active
11	1	1	1	T3	0	1	1	T7	1	1	0	1	T2	1	Active
12	1	1	1	T3	0	0	1	T8	1	1	0	1	T2	1	Active
13	0	1	1	T4	1	1	1	T6	1	1	0	1	T2	1	Active
14	0	1	1	T4	0	1	1	T7	1	1	0	1	T2	1	Active
15	0	1	1	T4	0	0	1	T8	1	1	0	1	T2	1	Active
16	0	0	1	T5	1	1	1	T6	1	1	0	1	T2	1	Active
17	0	0	1	T5	0	1	1	T7	1	1	0	1	T2	1	Active
18	0	0	1	T5	0	0	1	T8	1	1	0	1	T2	1	Active

Tab. 3.1- Cas d'évolution du modèle de la figure 3.13

On lit le tableau 3.1 de la façon suivante :

Pour le cas n°5,



3.4.1.2- Cas des variables qualitatives unitaires en série

La modélisation que nous proposons pour ce cas se présente sous la forme d'une structure formée de deux places d'entrée. La première est suivie d'une structure de parallélisme modélisant le paramètre flux (temporel). L'autre est suivie par n_i structures de parallélisme ordonnées en série dont chacune modélise une variable qualité [TEL 09a].

la synchronisation entre la structure modélisant la variable temporelle et la structure modélisant la dernière variable qualitative de la ressource concernée est assurée à travers des canaux (places) de transmission d'informations [TEL 09a].

L'exemple 3.11 explique bien comment on modélise ce cas.

Exemple 3.11

Nous reprendrons le même exemple du sous-paragraphe précédent (3.3.1.1-). Seulement, les deux variables qualitatives unitaires $V_{1,1}$ et $V_{1,2}$ seront considérées cette fois en série.

Nous illustrons, par la figure 3.14, une loi de commande robuste hybride bi-critère (flux-qualité) de la ressource R_1 .

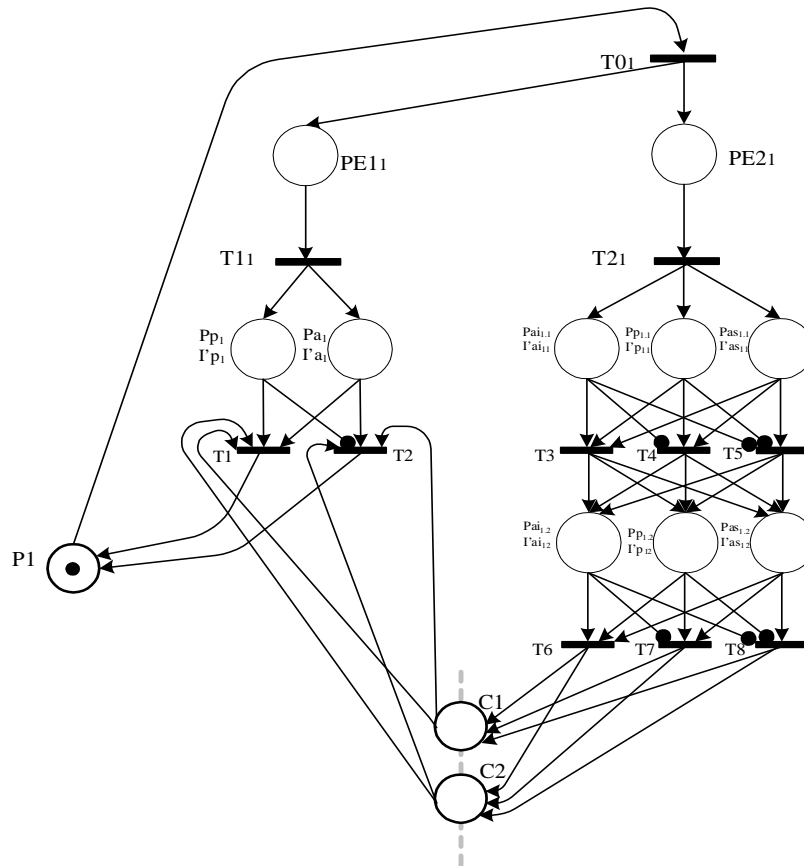


Fig. 3.13- Modèle d'une loi de commande robuste hybride bi-critère (flux-qualité) de la ressource R_l : cas des variables qualitatives unitaires en série

Le tableau 3.2 explique le principe de fonctionnement de ce modèle.

À	$t1 \in]a_{p1}, b_{a1}]$				$t2 \in]t1, b_{a1}]$				$t3 \in]t2, b_{a1}]$		$t4 \in [t3, b_{a1}]$				
Cas	Pai _{1,1}	Pp _{1,1}	Pas _{1,1}	T.F	Pai _{1,2}	Pp _{1,2}	Pas _{1,2}	T.F	C1	C2	Pp ₁	Pp ₂	T.F	P1	Rob. Ri
1	1	1	1	T3	1	1	1	T6	1	1	1	1	T1	1	Active
2	1	1	1	T3	0	1	1	T7	1	1	1	1	T1	1	Active
3	1	1	1	T3	0	0	1	T8	1	1	1	1	T1	1	Active
4	0	1	1	T4	1	1	1	T6	1	1	1	1	T1	1	Active
5	0	1	1	T4	0	1	1	T7	1	1	1	1	T1	1	Passive
6	0	1	1	T4	0	0	1	T8	1	1	1	1	T1	1	Active
7	0	0	1	T5	1	1	1	T6	1	1	1	1	T1	1	Active
8	0	0	1	T5	0	1	1	T7	1	1	1	1	T1	1	Active
9	0	0	1	T5	0	0	1	T8	1	1	1	1	T1	1	Active
10	1	1	1	T3	1	1	1	T6	1	1	0	1	T2	1	Active
11	1	1	1	T3	0	1	1	T7	1	1	0	1	T2	1	Active
12	1	1	1	T3	0	0	1	T8	1	1	0	1	T2	1	Active
13	0	1	1	T4	1	1	1	T6	1	1	0	1	T2	1	Active
14	0	1	1	T4	0	1	1	T7	1	1	0	1	T2	1	Active
15	0	1	1	T4	0	0	1	T8	1	1	0	1	T2	1	Active
16	0	0	1	T5	1	1	1	T6	1	1	0	1	T2	1	Active
17	0	0	1	T5	0	1	1	T7	1	1	0	1	T2	1	Active
18	0	0	1	T5	0	0	1	T8	1	1	0	1	T2	1	Active

Tab. 3.2- Cas d'évolution du modèle de la figure 3.14

La seule différence entre le tableau 3.1 et le tableau 3.2 réside dans les intervalles attribués à t_2 et t_3 .

3.4.2- Aspect formel pour le modèle de robustesse hybride bi-critère d'une ressource

Lemme 3.22 [TEL 08d], [TEL 08e], [TEL 09a]

Soit une ressource R_i de S . Si (au moins une des marges de robustesse passive des variables qualitatives ou temporelles n'était pas respectée), Alors (R_i est robuste activement).

Preuve

Si une seule marge de robustesse passive n'a pas été respectée (la marge de robustesse active est respectée), Les propriétés spécifiées de R_i n'auraient pu être maintenues qu'au prix d'un calcul total ou partiel de la conduite.

La définition 3 est applicable.

Nous allons présenter dans la suite l'aspect formel que nous proposons pour chaque cas :

- Cas des variables qualitatives unitaires en parallèle,
- Cas des variables qualitatives unitaires en série.

3.4.2.1- Aspect formel pour le cas des variables qualitatives unitaires en parallèle

Pour le cas des variables qualitatives unitaires en parallèle nous proposons le théorème suivant :

Théorème 3.23 [TEL 08d], [TEL 08e], [TEL 09a]

Une ressource R_i est robuste passivement si et seulement si les deux conditions suivantes sont satisfaites :

Condition 1 : à $t-\varepsilon$, $t=\min \{SP_{p_{i,j}}(n) \mid j \in \{1, 2, \dots, n_i\}\}$, les places modélisant la robustesse passive de R_i sont toutes marquées.

$$\left. \begin{array}{l} \text{card}(m(P_{p_i}))=1 \\ \forall j, \text{card}(m(P_{p_{i,j}}))=1 \end{array} \right\} \Rightarrow \text{card}(m(P_{p_i})) + \sum_{k=1}^{n_i} \text{card}(m(P_{p_{i,j}})) = n_i + 1 \quad (3.5)$$

Où : ε est réel positif négligeable par rapport au temps (t) ; $\varepsilon < t$

Condition 2 : à $t=SP_{p_i}^\circ(n) \in]a_{p_i}, b_{p_i}]$, les transitions de sorties relatives aux places modélisant la robustesse passive sont franchies.

$$\forall j, SP_{p_i}^\circ(n) \geq SP_{p_{i,j}}^\circ(n) \quad (3.6)$$

Preuve

Les jetons séjournés dans les places modélisent réellement l'état temporel et qualitatif d'un seul produit. Ceci implique qu'un jeton ne peut être pas passé pour séjournier dans la place en aval modélisant l'état de produit dans une autre ressource que si chaque structure de

parallélisme , de ressource concernée, contient au moins un jeton capable d'être tirée (chaque place ne peut comporter qu'un seul jeton).

D'autre part, tant que les marges de robustesse passive de toutes les variables temporelles et non temporelles sont choisies au départ sous des contraintes globales traduisant la robustesse passive de chaque ressource, une ressource ne peut être robuste passivement que si et seulement si les marges de robustesse passive de toutes les variables sont respectées.

La définition 2 est applicable.

3.4.2.2- Aspect formel pour le cas des variables qualitatives unitaires en série

Pour le cas des variables qualitatives unitaires en série nous proposons le théorème suivant :

Théorème 3.24 [TEL 09a]

Une ressource R_i est robuste passivement si et seulement si les trois conditions suivantes sont satisfaites :

Condition 1 : les marges de robustesse passive de toutes les variables qualitatives de R_i qui précèdent $V_{i,ni}$ ont été respectées.

$$\forall j \in \{1, 2, \dots, n_i - 1\}, \forall q(V_{i,j}) \in I_{p_{ij}} = [a_{p_{ij}}, b_{p_{ij}}] \quad (3.7)$$

Condition 2 : à $t - \varepsilon$, $t = SP_{p_{i,n_j}}(n)$, la place modélisant la robustesse passive de $V_{i,ni}$ est marquée.

$$\left. \begin{array}{l} \text{card}(m(P_{p_i})) = 1 \\ \text{card}(m(P_{p_{i,ni}})) = 1 \end{array} \right\} \Rightarrow \text{card}(m(P_{p_i})) + \text{card}(m(P_{p_{i,ni}})) = 2 \quad (3.8)$$

Où : ε est réel positif négligeable par rapport au temps (t) ; $\varepsilon \ll t$

Condition 3 : à $t = SP_{p_i}^\circ(n) \in]a_{p_i}, b_{p_i}]$,

$$SP_{p_i}^\circ(n) \geq \sum_j^{n_i} SP_{p_{i,j}}^\circ(n) \quad (3.9)$$

Preuve

Même raisonnement la preuve de théorème 3.23.

3.5- Modèles RdP de commande pour la génération de robustesse d'un Système de production: approche bi-critère

Après la mise en place du concept de base de modélisation des paramètres et ressources, nous nous intéressons, dans ce paragraphe, à la concrétisation de notre travail par la définition d'une méthode pour la modélisation complète d'un système de production.

3.5.1- Modélisation d'une commande robuste hybride bi-critère d'un système de production de type flow-shop

Pour, la conception et la modélisation d'une loi de commande robuste hybride bi-critère d'un système de production S . On propose les étapes ordonnées suivantes [NAB 08], [TEL 09a]:

(1) définir les variables flux-qualité spécifiant l'état de chaque ressource,

- (2) modéliser les paramètres (variables) chacun à part,
- (3) modéliser les ressources,
- (4) lier les modèles ressources.

Les trois premières étapes sont bien présentées et expliquées dans les paragraphes précédents. Par l'exemple 3.12, nous allons présenter notre méthode de modélisation et de synchronisation d'une série des ressources.

Exemple 3.12

Soit deux ressources successives R_i et R_{i+1} . R_i est caractérisée par un paramètre temporel et deux variables qualitatives en parallèle. R_{i+1} est caractérisée par un paramètre temporel et une seule variable qualité. Avec $a_{p_i} \geq b_{a_{i+1}}$ et $a_{p_{i+1}} < b_{a_{i+2}}$.

La figure 5 présente la modélisation de synchronisation de la loi de commande robuste hybride bi-critère.

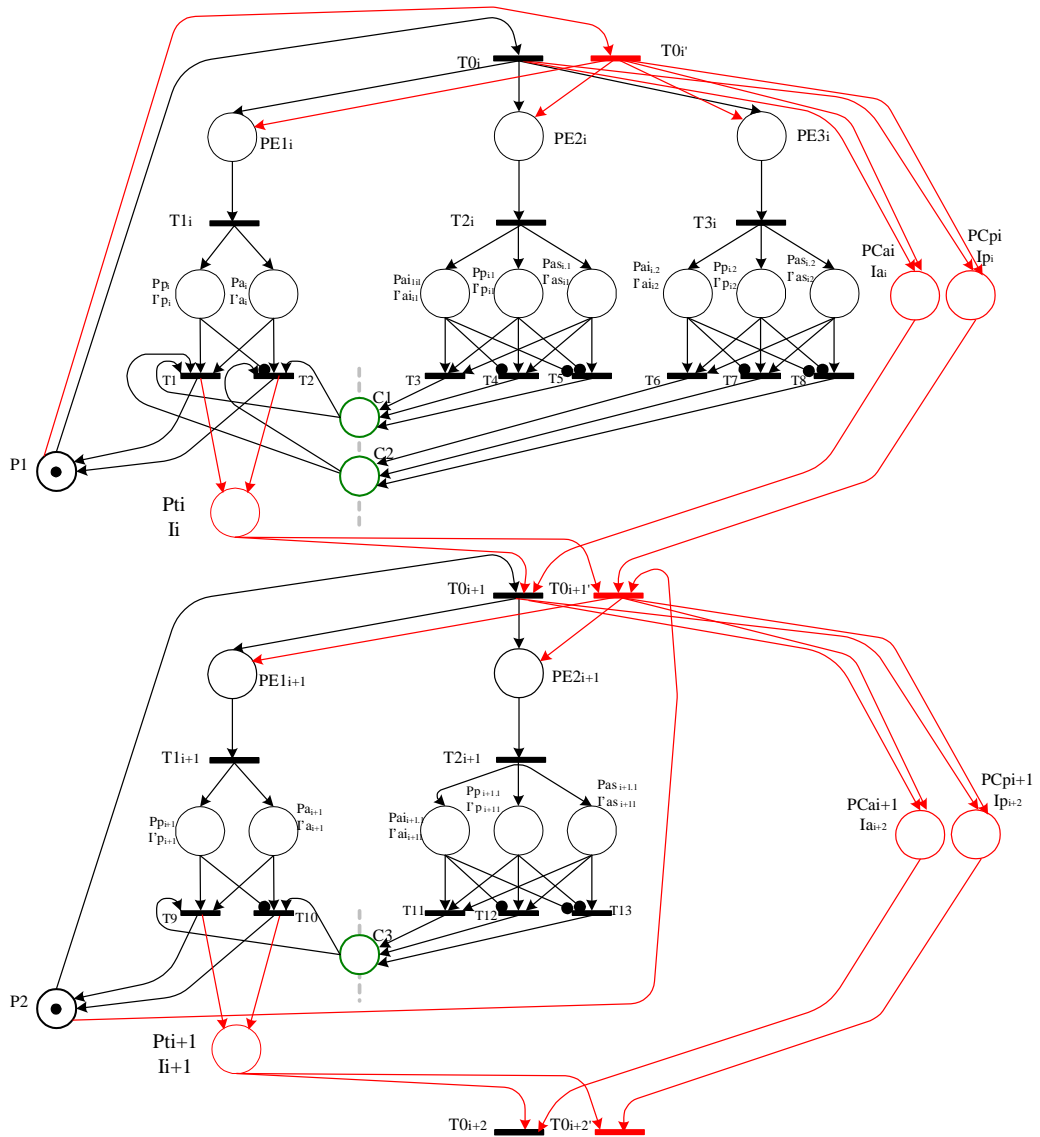


Fig. 3.14- Modèle d'une loi de commande robuste bi-critère (flux-qualité) pour la synchronisation de deux ressources R_i et R_{i+1}

Où :

PC_{pi} et PC_{ai} : places modélisant la robustesse temporelle de R_i tout en prenant compte du stock intermédiaire entre R_i ,

Ic_{pi} et Ic_{ai} : Intervalles statiques associés aux deux places PC_{pi} et PC_{ai} ,

PI_i : place modélisant, au cas où $a_{pi,j} < b_{ai,j+1}$, le stock intermédiaire entre R_i et R_{i+1} ,

$I_i = [0, +\infty]$: Intervalle statique alloué à PI_i , modélise le paramètre temps.

Les bornes des intervalles Ic_{pi} et Ic_{ai} seront définies dans le sous paragraphe suivant.

3.5.2- Aspect formel pour le cas des variables qualitatives unitaires en série

Dans ce sous paragraphe, nous nous intéressons, dans un premier lieu, à la mise en place d'un formalisme permettant de définir comment lier les modèles ressources. Dans un second lieu, nous allons présenter l'aspect formel permettant l'observation et la constatation de la robustesse globale du système de production.

Lemme 3.25 [TEL 09a]

Soit deux ressources successives R_i et R_{i+1} d'un système automatisé de production S . Les intervalles statiques associés aux deux places PC_{pi} et PC_{ai} modélisant la robustesse temporelle de R_i tout en prenant compte de la place modélisant le stock intermédiaire entre R_i et R_{i+1} sont :

$$Ic_{pi} = \begin{cases} Ip_i; & \text{si } a_{pi} \geq b_{ai+1} \\ Ip_{i+1}; & \text{si } a_{pi} < b_{ai+1} \end{cases} \quad (3.10)$$

$$Ic_{ai} = \begin{cases} Ia_i; & \text{si } a_{pi} \geq b_{ai+1} \\ Ia_{i+1}; & \text{si } a_{pi} < b_{ai+1} \end{cases} \quad (3.11)$$

Théorème 3.26 [TEL 09a]

Soit un système automatisé de production S composé de N ressources de Production $R_1 \rightarrow \dots \rightarrow R_i \rightarrow R_{i+1} \rightarrow \dots \rightarrow R_N$ qui se succèdent chronologiquement. Le système S est robuste passivement si et seulement si :

$$\text{Si } a_{pi} \geq b_{ai+1} : \forall i \in \{1, 2, \dots, N\}, SPI_i^\circ(n) + S^\circ PI_i(n) \in Ip_i = [a_{pi}, b_{pi}] \quad (3.12)$$

$$\text{Si } a_{pi} < b_{ai+1} : \forall i \in \{1, 2, \dots, N\}, SPI_i^\circ(n) + S^\circ PI_i(n) \in Ip_{i+1} = [a_{pi+1}, b_{pi+1}] \quad (3.13)$$

Preuve

Deux cas peuvent apparaître :

- Premier cas : $a_{pi} \geq b_{ai+1}$

La ressource R_{i+1} ne peut présenter en aucun cas un goulot d'étranglement. Si $SPI_i^\circ(n) + S^\circ PI_i(n) > b_{pi}$ (c.à.d. un jeton a séjourné une durée supérieure à 0 dans PI_i), R_i perd ces propriétés temporelles spécifiant sa robustesse passive.

- Deuxième cas : $a_{pi} < b_{ai+1}$

La ressource R_{i+1} peut présenter un goulot d'étranglement. Si $SPI_i^\circ(n) + S^\circ PI_i(n) > b_{pi+1}$, R_{i+1} perd ces propriétés temporelles spécifiant sa robustesse passive.

La définition 2 est applicable

Lemme 27 [NAB 08], [TEL 08e], [TEL 09a]

Soit un système automatisé de production S remembrant N ressources. Si (il \exists pour $i \in \{1, 2, \dots, N\}$, au moins R_i robuste activement), alors (S est robuste activement).

Preuve

Même raisonnement de la preuve de lemme 3.13.

3.5.3- Application

L'atelier considéré correspond à celui étudié dans [DHO 05a], [TEL 08b], [TEL 09a]. En effet le système étudié est une unité de manufacture de confection des cigarettes. Le contrôle du poids des unités fabriquées conduit à l'amélioration du taux de production, la qualité et le coût, en limitant les rebuts.

L'atelier est constitué de quatre ressources de production : tapis d'entraînement du tabac, brosses d'écêtage, machine de formation du boudin et machine de coupe des cigarettes.

Pour une cigarette, le poids est fonction de trois paramètres : le module (m), la longueur (l) et la densité (d). L'objectif est d'examiner le respect de la tolérance sur le poids via les trois paramètres m , l et d qui doivent respecter les marges de conformité.

En effet, on peut assimiler une cigarette à un cylindre de diamètre m et de longueur l . Le poids (P) de la cigarette est donné alors par la relation suivante :

$$P = k.m^2.l.d \quad (3.14)$$

Où :

k est une constante ($k = \frac{\pi}{4}$).

Après un développement de Taylor au premier ordre et le remplacement de l'accroissement de cette fonction (3.14) par l'accroissement d'une forme linéaire la relation devient :

$$\Delta P \leq K_1 \Delta d + K_2 \Delta m + K_3 \Delta l \quad (3.15)$$

Avec :

$$\begin{aligned} \Delta l &= [l_{\min}, l_{\max}] ; & K_1 &= klm^2 ; \\ \Delta d &= [d_{\min}, d_{\max}] ; & \text{et } K_2 &= 2klmd ; \\ \Delta m &= [m_{\min}, m_{\max}] ; & K_3 &= kdm^2. \\ \Delta P &= [P_{\min}, P_{\max}] ; \end{aligned}$$

Ces intervalles couvrent les marges de robustesse active et passive.

D'autre part, la densité d du tabac est une fonction de trois paramètres qui sont : la compacité du tabac (c), le niveau du tabac (n) et l'hauteur du tabac écêté (e).

d est donnée alors par la relation suivante :

$$d = \alpha.c.n.e \quad (3.16)$$

Où, α un coefficient constant exprimé en mm^{-2} .

Après développement limité de la relation non linéaire (4.16) au premier ordre on obtient un accroissement d'une forme linéaire :

$$\Delta d \leq \alpha (ne\Delta c + ce\Delta n + nc\Delta e) \tag{3.17}$$

Avec :

$$\Delta c = [c_{min}, c_{max}];$$

$$\Delta l = [e_{min}, e_{max}];$$

$$\Delta n = [n_{min}, n_{max}];$$

$$\Delta d = [d_{min}, d_{max}].$$

Ces intervalles couvrent les marges de robustesse active et passive.

On résume les paramètres temporels et les variables qualité à contrôler pour chaque ressource dans le tableau 3.3.

Ressources	Paramètres temporelles	Paramètres qualité	
		Explicatives	Expliquées
R ₁ : Tapis d'entraînement du tabac	<i>Tm1</i>	<i>c</i> et <i>n</i>	<i>d</i> <i>P</i>
R ₂ : Brosses d'écrotage	<i>Tm2</i>	<i>e</i>	
R ₃ : Machine de formation du boudin	<i>Tm3</i>	<i>m</i>	
R ₄ : Machine de coupe des cigarettes	<i>Tm4</i>	<i>l</i>	

Tab. 3.3- Paramètres temporels et qualité à contrôler

où :

c et *n* : deux variables qualitatives en parallèle,

Tmi : temps de production élémentaire de ressource *R_i*.

Les marges de robustesse de flux (temporelles) sont allouées à base unitaire d'une cigarette. La chaîne de production est bien équilibrée : $a_{p_i} \geq b_{a_{i+1}}$; $Ic_{p_i} = Ip_i$ et $Ic_{a_i} = Ia_i$. Le tableau 3.4, présente les valeurs de tolérance pour chaque variable.

Paramètre	Unité	Valeur /active	Valeur /passive
<i>c_{min}</i>	g/mm ³	4.5	5
<i>c_{max}</i>	g/mm ³	6	5.5
<i>n_{min}</i>	mm	100	230
<i>n_{max}</i>	mm	500	370
<i>e_{min}</i>	mm	7.8	7.9
<i>e_{max}</i>	mm	8.1	8
<i>d_{min}</i>	g/mm ³	0.2	
<i>d_{max}</i>	g/mm ³	0.23	
<i>m_{min}</i>	mm	7.77	7.85
<i>m_{max}</i>	mm	8.03	8.95
<i>l_{min}</i>	mm	63.7	63.9
<i>l_{max}</i>	mm	64.3	64.1
<i>P_{min}</i>	mg	745	
<i>P_{max}</i>	mg	805	

Tab. 3.4- Valeurs des marges de robustesse de paramètres de qualité

Note : Les mages de la robustesse passive de *d* et *p* se présentent implicitement car *d* et *p* sont deux variables expliquées.

La figure 3.15 illustre la modélisation hybride de la robustesse de système de production.

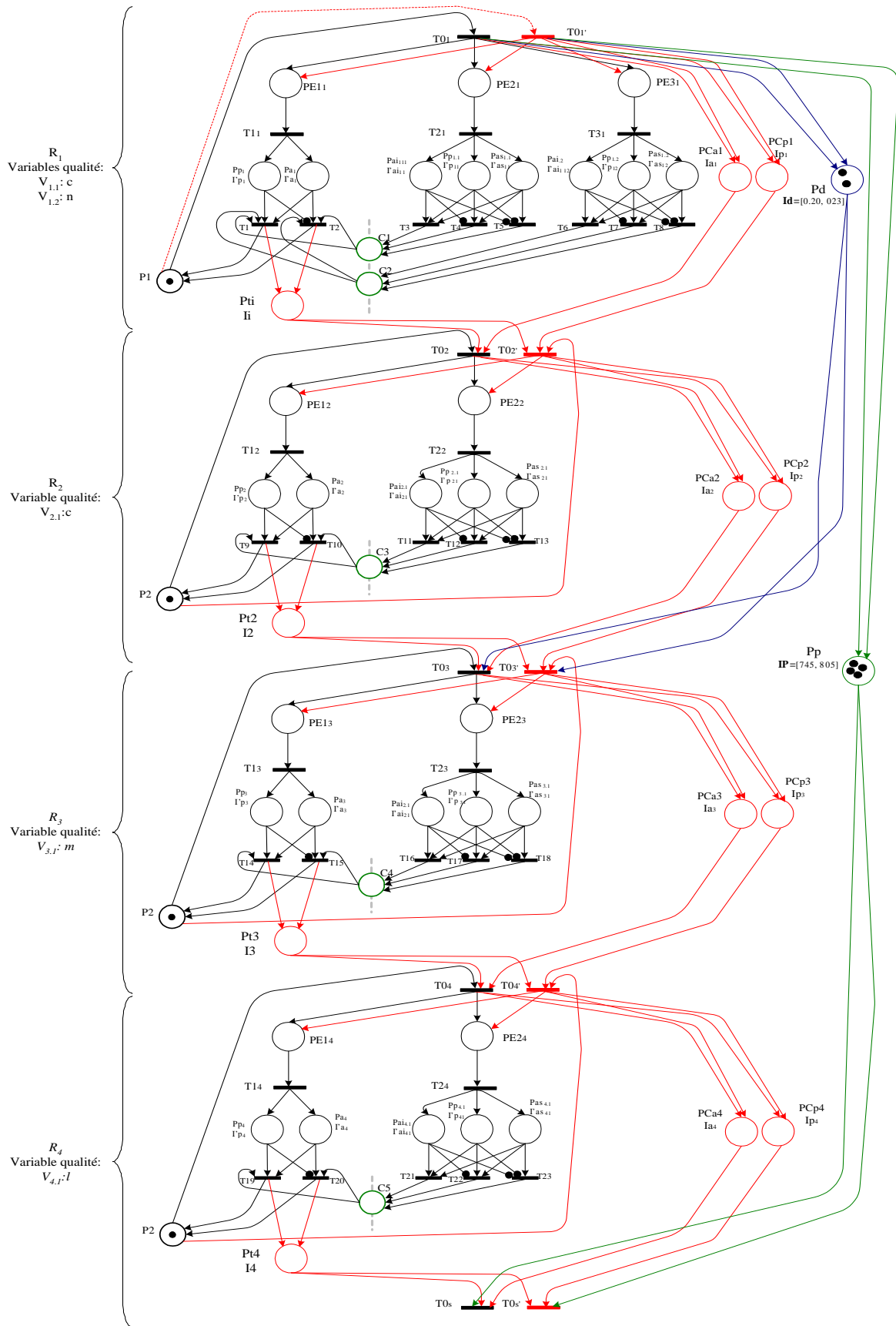


Fig. 3.15- Modèle d'une loi de commande robuste bi-critère (flux-qualité) de la manufacture de confection des cigarettes

où :

P_d et P_p : places modélisant les marges de robustesse active respectivement de densité (d) et de Poids (P).

3.6- Conclusion

Dans ce chapitre, nous nous sommes intéressés à développer une approche de modélisation de la commande robuste de notre architecture de Surveillance-Commande proposée : « Surveillance Indirecte suite à une Commande Robuste ». L'outil de modélisation exploité est les réseaux de Petri à Intervalles.

En effet, la commande robuste proposée a porté deux contributions par rapport à la modélisation classique située dans : [COL 97], [KHA 97], [BON 00], [COL 04], [DHO 05a], [JER 06]. La première contribution consiste à la modélisation, au même temps et dans le même modèle, de la robustesse active et passive d'un seul critère : flux ou qualité. La seconde a porté sur la modélisation d'une loi de commande bi-critère qui précise à la fois la robustesse de deux critères : flux et qualité. Le résultat de ces deux contributions est un modèle d'une commande robuste hybride (passive et active) bi-critère (flux et qualité).

L'objectif de ces deux contributions est le développement d'un concept qui permet l'observation de la nature de robustesse des différents composants de système de production opérant (paramètres, circuits élémentaires et ressources) tout au long de fonctionnement du système de production. Aussi, elles nous permettent de constater la nature de robustesse :

- d'un circuit élémentaire ou d'une ressource à partir de l'observation de la robustesse de leurs paramètres relatifs,
- de la nature de robustesse du système de production à partir de l'observation de la robustesse des circuits élémentaires ou des ressources.

Nous avons cumulé chaque contribution de modélisation par un développement théorique qui permet la synthèse de la nature de robustesse. Des exemples applicatifs sont illustrés.

Chapitre 4
Surveillance Indirecte : approche
multicritères

Chapitre 4

Surveillance Indirecte : approche multicritères

Table des matières du chapitre 4

4.1-	Introduction.....	111
4.2-	Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales (RR ² FR) pour le pronostic ...	111
4.2.1-	Réseaux de neurones récurrents à fonctions de base radiales (RRFR).....	112
4.2.2-	Proposition d'une architecture améliorée des RFR : Le RR ² FR [TEL 09b].....	115
4.2.2.1-	Topologie de l'architecture RR ² FR	115
4.2.2.2-	Analyse de comportement de mémorisation et de corrélation d'un RR ² FR	116
4.2.2.3-	Apprentissage des RR ² FR.....	120
4.2.2.3.1-	Phase 1 : initialisation les paramètres de neurones bouclés	120
4.2.2.3.2-	Phase 2 : détermination des centres et rayons d'influence des neurones gaussiens	120
4.2.2.3.3-	Phase 3 : calcul des poids des connexions de sortie	122
4.2.2.4-	Application des RR ² FR.....	124
4.2.2.4.1-	Application des RR ² FR à la Série temporelle <i>MackeyGlass</i> [TEL 09b].....	124
4.2.2.4.1.1-	Simulation	124
4.2.2.4.1.2-	Interprétations.....	127
4.2.2.4.2-	Application des RR ² FR à la prédiction d'une concentration de sortie en CO ² d'un four à gaz.....	127
4.2.3-	Interprétation générale et apport de l'architecture RR ² FR	131
4.3-	Détection des symptômes par suivi des paramètres de loi de commande à travers les RdPTàO : approche qualité-flux	132
4.3.1-	Modèle de Combacau pour la détection des symptômes temporels [COM 91]	132
4.3.2-	Modèle de Suivi-Détection multicritères : Qualité-flux [TEL 08c]	134
4.3.3-	Application	137
4.3.4-	Interprétation générale et apport de Modèle de Suivi-Détection proposé	141
4.4-	Synchronisation entre le bloc Pronostic et le bloc Suivi-Détection	141
4.4.1-	Principe de communication	141
4.4.2-	Application	142
4.5-	Conclusion	144

4.1- Introduction

La surveillance d'un système industriel avec des méthodes qui se basent sur l'outil mathématique nécessite une maîtrise des paramètres du système ainsi que leurs interactions. Ceci est presque impossible pour un système industriel. Notre choix annoncé dans le chapitre deux consiste à utiliser des méthodes et outils de surveillance qui sont plus adaptables pour un système industriel. Bien sûr, les objectifs et les contraintes de notre approche sont prises en considération et présentent l'élément le plus important qui explique notre choix. En effet, nous avons prédestiné l'outil Réseaux de neurones à Fonctions de base Radiales RFR pour le Pronostic et l'outil Réseau de Petri Temporels à Objets pour la fonction Suivi-Détection.

Le présent chapitre est consacré à la présentation de notre méthodologie de surveillance. En effet, notre approche de surveillance, intégrée dans notre architecture, consiste à surveiller deux critères différents, à la fois : le flux et la qualité. En se référant à la littérature [LI 99], [NAB 00], [DHO 05a], etc., la notion de surveillance indirecte avec la prise en compte de deux critères présente une des contributions de cette thèse.

En effet, nous nous intéressons, dans un premier lieu, à la fonction pronostic tout en essayant de procurer des améliorations à l'architecture RFR, proposée dans [ZEM 03]. Nous allons présenter une nouvelle architecture neuronale destinée à la prédiction des états : Le Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales (RR²FR). Le RR²FR proposé s'excite d'une part par le bouclage local des neurones d'entrée et d'autre part par le rebouclage des signaux de sortie sur l'entrée du réseau. Ceci permet la prise en compte de la dynamique des données d'entrée et des résultats estimés par rapport à l'état réel du système. Nous proposons aussi un algorithme d'apprentissage non supervisé des neurones à base des fonctions radiales.

Dans un second lieu, la fonction Suivi-Détection, occupera notre intérêt. En effet, le modèle de Suivi-Détection proposé par [COM 91] sera la base de notre contribution. A partir de ce travail exploitant l'outil Réseau de Petri Temporels à Objets, nous allons développer notre modèle qui va intervenir la notion de suivi des paramètres temporels (flux) et non-temporels (qualité) au même temps afin de détecter les symptômes de défaillances.

Nous terminerons par l'explication du principe de synchronisation entre Le RR²FR proposé pour le pronostic et le modèle Réseau de Petri Temporels à Objets du bloc Suivi-Détection.

4.2- Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales (RR²FR) pour le pronostic

L'objectif de cette section est de proposer un outil permettant le pronostic des valeurs futures des paramètres flux et qualité de produit intégrés dans une la loi de commande.

Les réseaux de neurones récurrents sont les plus efficaces, parmi l'ensemble des représentations temporelles, pour des applications de surveillance dynamique. En effet, ils permettent à travers les récurrences de graver une mémoire dynamique interne. Ce type de réseaux de neurones se montre efficace pour servir trois types d'applications en surveillance dynamique : la prédiction temporelle, la reproduction de séquences temporelles et la reconnaissance de séquences temporelles. L'apprentissage de ces réseaux est souvent très complexe et lourd à implanter et notamment coûteux en temps de calcul. La façon la simple

pour avoir une mémoire dynamique interne est d'introduire des récurrences locales au niveau du neurone lui-même. Ce type d'architectures est appelée architecture LRGF : Locally Recurrent Globally Feedforward ou architecture Localement Récurrente Globalement Feedforward. Tout au long de ce paragraphe, nous allons présenter notre architecture RR²FR, qui introduit des récurrences locales au niveau des neurones d'entrée et de sortie d'un RFR classique.

4.2.1- Réseaux de neurones récurrents à fonctions de base radiales (RRFR)

Proposés par [ZEM 03], les RRFR considèrent le temps comme une représentation interne au réseau. En effet, l'architecture des RFR est maintenue, seulement la structure de la couche d'entrée est modifiée (Fig. 4.1).

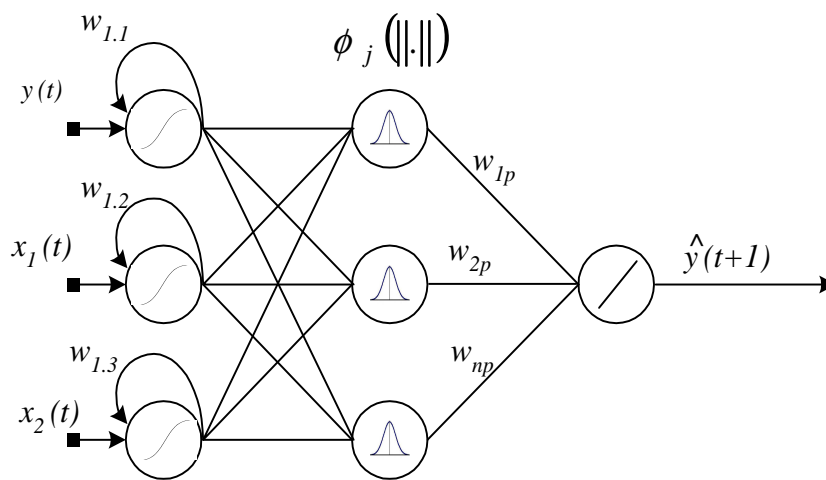


Fig. 4.1- Architecture d'un RRFR.

L'architecture RRFR est composée de trois couches : une couche d'entrée qui a les spécificités d'une mémoire dynamique, une couche cachée qui permet de mémoriser le prototype donc elle s'agit d'une mémoire statique et une couche de décision (la couche de sortie).

Par rapport aux RFR classiques, deux modifications ont été apportées à la couche d'entrée : la récurrence locale des connexions des neurones et la fonction d'activation qui devient une sigmoïde :

$$f(x) = \frac{1 - \exp(-kx)}{1 + \exp(-kx)} \quad (4.1)$$

Les modifications apportées par [ZEM 03], sur la couche d'entrée ont l'effet de greffer une trace du passé des entrées.

Le bouclage au niveau des neurones de la couche d'entrée est consacré pour que le signal d'entrée ne se propage pas seulement de la couche d'entrée vers la couche de sortie comme

dans les réseaux de neurones à propagation avant. Cette boucle fermée permet au réseau de garder en mémoire une trace interne d'un signal d'entrée.

On distingue dans la littérature trois types de LRGF [ZEM 03]:

- Les architectures à retour local synaptique (Local Synapse Feedback),
- Les architectures à retour local de l'activation (Local Activation Feedback),
- Les architectures à retour local de la sortie du neurone (Local Output Feedback).

[ZEM 03] a montré que les architectures à retour local de la sortie du neurone sont plus performantes. En effet, il a exploité le modèle de Frasconi-Gori-Soda pour l'architecture RRFR (Fig. 4.2).

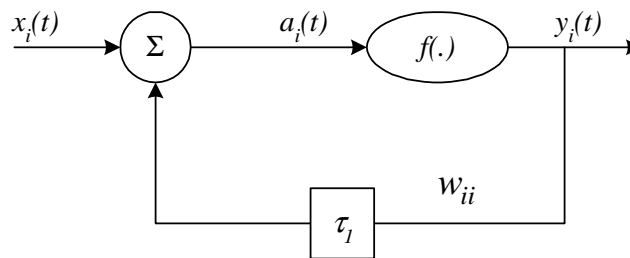


Fig. 4.2- Architecture, d'un réseau LRGF, simplifiée avec retour de la sortie.

Chaque neurone de la couche d'entrée effectue une sommation à l'instant t entre son entrée I_i et sa sortie de l'instant précédent $(t-1)$ pondérée par le poids de l'autoconnexion w_{ii} . Le résultat est la fonction d'activation suivante :

$$a_i(t) = w_{ii}y_i(t-1) + x_i(t) \quad (4.2)$$

La sortie du neurone i à l'instant t est la suivante :

$$y_i(t) = f(a_i(t)) \quad (4.3)$$

Afin d'évaluer la performance de ce bouclage au niveau de neurone d'entrée en termes de sa capacité de créer une mémoire au réseau, généralement, on considère, sur l'entrée, les conditions initiales suivantes : $x_i(t_0) = 0$ et $y_i(t_0) = 1$. Puis, on évolue le neurone en faisant abstenir les excitations provenant de l'entrée extérieure [FRA 95], [BER 96], [ZER 03]. La sortie du neurone a donc l'expression suivante :

$$y(t) = \frac{1 - \exp(-kw_{ii}y(t-1))}{1 + \exp(-kw_{ii}y(t-1))} \quad (4.4)$$

Lorsque aucune excitation extérieure ne se présente, l'activation du neurone bouclé s'exprime par l'équation suivante :

$$a_i(t) = w_{ii}f(a_i(t-1)) \quad (4.5)$$

La recherche des racines de la fonction g (équation 4.6) permet de déterminer les points d'équilibre (c.à.d. lorsque : $a^* = w_{ii} f(a^*)$).

$$g(a) = w_{ii} f(a) - a \quad (4.6)$$

La valeur $a_0=0$ est une solution évidente de $g(a) = 0$. Les autres points d'équilibre, s'ils existent, peuvent être déterminés en faisant résoudre la fonction dérivée de $g(a)$. Les racines de $g'(a) = 0$ sont les autres points d'équilibre.

$$g'(a) = -\frac{(\exp^{-ka})^2 + 2 \exp^{-ka} (1 - kw_{ii}) + 1}{(1 + \exp^{-ka})^2} \quad (4.7)$$

Par la figure 4.3 nous présentons l'évolution de la sortie du neurone bouclé à chaque instant.

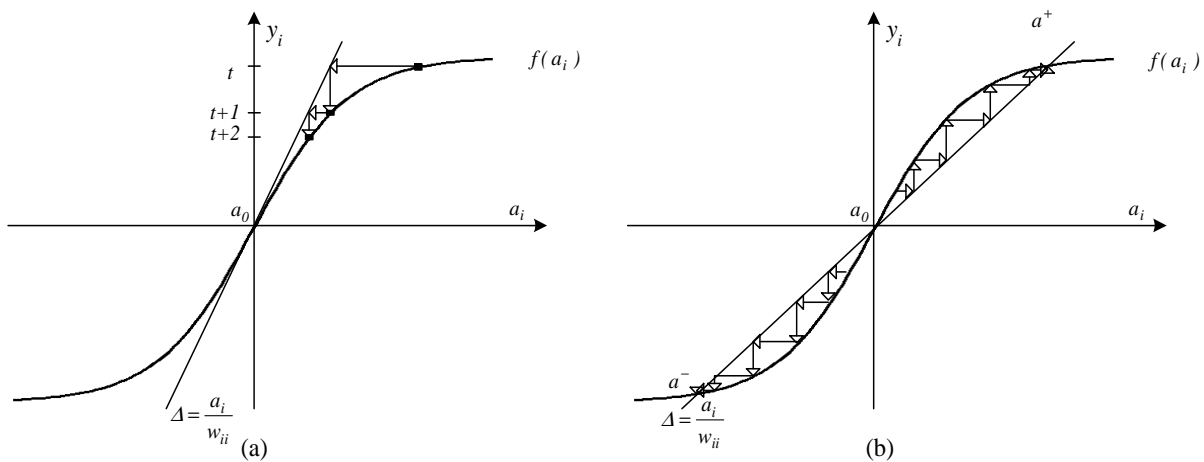


Fig. 4.3- Évolution de la sortie et points d'équilibre du neurone bouclé :

- a - comportement d'oubli ($kw_{ii} \leq 2$).
- b - comportement de mémorisation temporaire ($kw_{ii} > 2$)

On remarque bien que l'évolution de la sortie du neurone dépend de la pente de la droite Δ . Ceci implique que le poids de la connexion w_{ii} et la valeur du paramètre k de la fonction d'activation présentent les variables explicatives de la sortie d'un neurone bouclé.

Selon la valeur du produit kw_{ii} le comportement de neurone bouclé varie (Fig.4) :

- Si $kw_{ii} \leq 2$, le neurone possède un seul point d'équilibre a_0 . Ceci implique que le neurone a un comportement d'oubli.
- Si $kw_{ii} > 2$, le neurone possède trois points d'équilibres : a_0, a^-, a^+ . Ceci implique que le neurone a un comportement de mémorisation temporaire.

Le réseau RRBF a été testé avec succès, par [ZEM 03], sur le problème de prédiction des états de la série temporelle de Mackey-Glasse et d'un système non linéaire (un four à gaz). Ce réseau donne des résultats plus performants que les RFR classiques en termes de l'erreur de prédiction.

L'intérêt de l'architecture RRFR réside dans deux points. Le premier apport est la topologie du RFR qui permet de prendre en compte l'aspect dynamique des données. Le second apport

touche l'algorithme d'apprentissage non supervisé qui a été amélioré. Une technique améliorée de l'algorithme non supervisé K-moyens a permis de déterminer d'une manière plus stable et efficace les paramètres des neurones cachés.

Dans les lignes qui suivent, nous allons présenter la topologie ainsi que les caractéristiques de notre architecture : Réseau de Neurones doublement Récurrent à Base de Fonctions Radiales.

4.2.2- Proposition d'une architecture améliorée des RFR : Le RR²FR [TEL 09b]

4.2.2.1- Topologie de l'architecture RR²FR

La fonction d'activation ainsi que la récurrence au niveau des neurones d'entrée ont apporté une amélioration de la performance de prédiction par rapport aux RFR classiques [ZEM 03]. Dans ce paragraphe, une nouvelle architecture est proposée : Réseau de neurones Doublement Récurrent à base de Fonctions Radiales (RR²FR). Le RR²FR considère le temps comme une représentation interne au réseau. Le RR²FR proposé consiste à introduire à l'architecture RRFR des récurrences au niveau des neurones de sortie et de mettre en interaction la sortie estimée $\hat{y}(t+1)$ avec la sortie réelle $y(t+1)$ du paramètre sujet à la prédiction afin de prédire $\hat{y}(t+2)$.

Nous considérons toujours pour les RR²FR que les neurones d'entrée de chaque RRFR ont la fonction d'activation sigmoïde et que les neurones de la couche cachée (couche de classification) ont la fonction d'activation gaussienne [TEL 09b]. La figure 4.4 présente l'architecture du RR²FR.

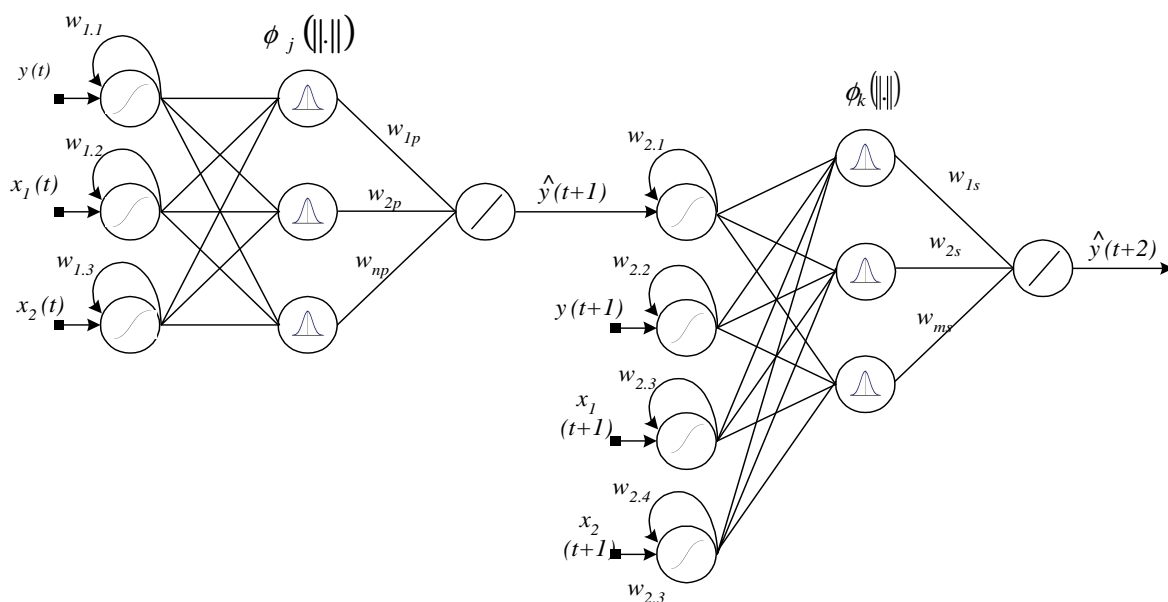


Fig. 4.4- Architecture du RR²FR.

Cette architecture est composée de deux RRFR [TEL 09b] :

- Estimateur de premier ordre : c'est un RRFR classique [ZEM 03]. Il permet de prédire la valeur d'une variable de sortie à un instant $t+1$ à partir des valeurs connues (à un instant t) d'un ensemble $\Theta(t) = \{x_1(t), x_2(t), \dots, x_n(t), y(t)\}$ des variables d'états conçues pour la prédiction.

- Estimateur de second ordre : il permet la prédiction de la valeur de variable de sortie à un instant $t+i$. Les entrées de ce deuxième RRFR sont $\hat{y}(t+1)$ et les valeurs des autres variables d'entrée à l'instant $t+1$ qui sont $\mathcal{O}(t+1)=\{x_1(t+1), x_2(t+1), \dots, x_n(t+1), y(t+1)\}$.

Le bouclage au niveau des neurones d'entrée de chaque RRFR permet de prendre en considération une partie importante de l'historique de variation de chaque variable. Donc pour cette nouvelle architecture des réseaux des neurones, deux types des variables sont mémorisés : les variables d'états ou les variables d'entrées de chaque RRFR et la variable de sortie du premier RRFR. Une telle architecture permet d'inférer, à travers la mémoire dynamique créée par les récurrences des neurones d'entrée du premier et de second RRFR, les successions des informations d'entrée et de sortie.

Dans le sous-paragraphe suivant, nous nous intéresserons à démontrer la capacité de RR²FR à mémoriser les informations introduites par l'entrée du réseau et les résultats estimés. Nous mettrons l'accent, aussi, sur la capacité des RR²FR à prendre en considération la corrélation entre les états estimés par le premier RRFR et les états réels du système qui vont être introduits au deuxième RRFR.

4.2.2.2- Analyse de comportement de mémorisation et de corrélation d'un RR²FR

On nome par $E_1(t)$ (respectivement $E_2(t+1)$) les vecteurs d'entrée du premier RRFR (respectivement du deuxième RRFR). Aussi, on nomme par $\hat{y}(t+1)$ (respectivement $\hat{y}(t+i)$) la sortie du premier RRFR (respectivement du deuxième RRFR) ; où i est l'horizon d'estimation du RR²FR. La dimension du vecteur d'entrée est $n+1$ pour le premier RRFR et $n+2$ pour le second RRFR.

$$E_1(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \\ y(t) \end{bmatrix} ; \quad E_2(t+1) = \begin{bmatrix} x_1(t+1) \\ x_2(t+1) \\ \vdots \\ x_n(t+1) \\ y(t+1) \\ \hat{y}(t+1) \end{bmatrix}$$

Le bouclage effectué au niveau de la couche d'entrée d'un RRFR permet de mémoriser l'historique des variables d'entrée. Ceci implique que chaque variable d'entrée du deuxième RRFR simule un vecteur de dimension m ; avec :

$$X_i(t+1) = \begin{bmatrix} x_i(t+1) \\ x_i(t) \\ x_i(t-1) \\ \vdots \\ x_i(t-m+1) \end{bmatrix} ; \quad Y(t+1) = \begin{bmatrix} y(t+1) \\ y(t) \\ y(t-1) \\ \vdots \\ y(t-m+1) \end{bmatrix} ; \quad \hat{Y}(t+1) = \begin{bmatrix} \hat{y}(t+1) \\ \hat{y}(t) \\ \hat{y}(t-1) \\ \vdots \\ \hat{y}(t-m+1) \end{bmatrix}$$

Où m est la taille de la mémoire obtenue par le bouclage des neurones et $i \in \{1, 2, \dots, n\}$.

Ceci signifie que la dimension de vecteur d'entrée du premier RRFR peut être simulé par un vecteur de dimension $(n+1)m$. Le vecteur d'entrée du deuxième RRFR est simulé par vecteur de dimension $(n+2)m$. Les deux vecteurs E_1 et E_2 sont équivalant aux :

$$E_1(t) = \begin{bmatrix} X_1(t) \\ X_2(t) \\ \vdots \\ X_n(t) \\ Y(t) \end{bmatrix}; \quad E_2(t+1) = \begin{bmatrix} X_1(t+1) \\ X_2(t+1) \\ \vdots \\ X_n(t+1) \\ Y(t+1) \\ \hat{Y}(t+1) \end{bmatrix}$$

Nous présentons, dans les lignes qui suivent, nos démonstrations affirmant la capacité des RR^2FR à prendre en considération la corrélation entre $\hat{Y}(t+1)$ et $Y(t+1)$.

Démonstration 4.1 [TEL 09b]

Considérons un RR^2FR qui ne possède aucune variable d'entrée autre que $y(t)$. $y(t)$ présente aussi le seul paramètre à estimer. L'objectif est de prédire sa valeur à un instant $(t+1)$ dans une première phase par le premier $RRFR$ et sa valeur à un instant $(t+i)$ dans une deuxième phase par le second $RRFR$.

Afin d'atteindre notre objectif, nous commençons notre démonstration par considérer que la fonction d'activation de neurone d'entrée de chaque $RRFR$ est linéaire.

Le vecteur d'entrée du deuxième $RRFR$ de RR^2FR est le suivant :

$$E_2(t+1) = \begin{bmatrix} Y(t+1) \\ \hat{Y}(t+1) \end{bmatrix}$$

Les expressions de la covariance $cov(Y(t+1), \hat{Y}(t+1))$ et du coefficient de corrélation $\rho(Y(t+1), \hat{Y}(t+1))$ s'écrivent comme suit :

$$cov(Y(t+1), \hat{Y}(t+1)) = \frac{\sum_{k=t-m+1}^{t+1} (y(k) - \mu_Y) - (\hat{y}(k) - \mu_{\hat{Y}})}{m} \quad (4.8)$$

$$\rho(Y(t+1), \hat{Y}(t+1)) = \frac{cov(Y(t), \hat{Y}(t))}{\sigma_Y \sigma_{\hat{Y}}} \quad (4.9)$$

L'équation (2.32) s'écrit de la façon suivante :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}}^2} \left(\begin{pmatrix} Y(t+1) \\ \hat{Y}(t+1) \end{pmatrix} - \begin{pmatrix} \mu_Y \\ \mu_{\hat{Y}} \end{pmatrix} \right) \left(\begin{pmatrix} Y(t+1) \\ \hat{Y}(t+1) \end{pmatrix} - \begin{pmatrix} \mu_Y \\ \mu_{\hat{Y}} \end{pmatrix} \right)^t \right) \quad (4.9)$$

D'où :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}}^2} \sum_{k=t-m+1}^{t+1} (y(k) - \mu_Y)^2 + (\hat{y}(k) - \mu_{\hat{Y}})^2 \right) \quad (4.10)$$

Cette équation est équivalente à :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}Y}^2} \sum_{k=t-m+1}^{t+1} \left((y(k) - \mu_Y)^2 + (\hat{y}(k) - \mu_{\hat{Y}})^2 + (y(k) - \mu_Y) \right) \right) \quad (4.11)$$

D'où :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}Y}^2} \sum_{k=t-m+1}^{t+1} \left((y(k) - \mu_Y)(y(k) - \mu_Y - 1) - (\hat{y}(k) - \mu_{\hat{Y}})(\hat{y}(k) - \mu_{\hat{Y}} + 1) + (y(k) - \mu_Y) - (\hat{y}(k) - \mu_{\hat{Y}}) \right) \right) \quad (4.12)$$

Il en résulte :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}Y}^2} \left(\sum_{k=t-m+1}^{t+1} \left((y(k) - \mu_Y)(y(k) - \mu_Y - 1) - (\hat{y}(k) - \mu_{\hat{Y}})(\hat{y}(k) - \mu_{\hat{Y}} + 1) + (y(k) - \mu_Y) - (\hat{y}(k) - \mu_{\hat{Y}}) \right) \right) \right) \quad (4.13)$$

En utilisant l'équation (4.8), nous obtenons :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}Y}^2} \left(\text{cov}(Y(t+1), \hat{Y}(t+1))m + \sum_{k=t-m+1}^{t+1} (y(k) - \mu_Y)(y(k) - \mu_Y - 1) - (\hat{y}(k) - \mu_{\hat{Y}})(\hat{y}(k) - \mu_{\hat{Y}} + 1) \right) \right) \quad (4.14)$$

Finalement, la prise en considération de l'équation (4.9), nous mène à conclure que l'expression de $\phi(E_2(t+1))$ dépend du coefficient de corrélation $\rho(Y(t+1), \hat{Y}(t+1))$:

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{\hat{Y}Y}^2} \left(m\sigma_Y\sigma_{\hat{Y}}\rho(Y(t+1), \hat{Y}(t+1)) + \sum_{k=t-m+1}^{t+1} (y(k) - \mu_Y)(y(k) - \mu_Y - 1) - (\hat{y}(k) - \mu_{\hat{Y}})(\hat{y}(k) - \mu_{\hat{Y}} + 1) \right) \right) \quad (4.15)$$

L'équation (4.15) montre bien que le coefficient de corrélation entre $\hat{Y}(t+1)$ et $Y(t+1)$ est pris en considération par le deuxième RRFR du RR²FR lorsque la fonction d'activation est linéaire.

On montre, dans les lignes qui suivent, que la corrélation est prise implicitement en compte aussi lorsque la fonction d'activation est une sigmoïde.

Démonstration 4.2[TEL 09b]

Lorsque la fonction sigmoïde est prise comme fonction d'activation des neurones de la couche d'entrées, le vecteur d'entrée de la couche cachée du deuxième RRFR s'écrit comme suit :

$$S_2(t+1) = \begin{bmatrix} h_{Y_2}(t+1) \\ h_{\hat{Y}_2}(t+1) \end{bmatrix}$$

Avec :

$$h_{Y_2}(t+1) = \begin{bmatrix} h_{y_2}(t+1) \\ h_{y_2}(t) \\ h_{y_2}(t-1) \\ \vdots \\ h_{y_2}(t-m+1) \end{bmatrix} \quad \text{et} \quad h_{\hat{Y}_2}(t+1) = \begin{bmatrix} h_{\hat{y}_2}(t+1) \\ h_{\hat{y}_2}(t) \\ h_{\hat{y}_2}(t-1) \\ \vdots \\ h_{\hat{y}_2}(t-m+1) \end{bmatrix}$$

Où :

$$h_{y_2}(t+1) = \frac{1 - \exp(-kw_{ii}y(t))}{1 + \exp(-kw_{ii}y(t))} \tag{4.16}$$

En suivant la même démarche de la démonstration 4.1, nous obtenons l'équation suivante :

$$\phi(E_2(t+1)) = \exp \left(-\frac{1}{2\sigma_{Y\hat{Y}}^2} \left(m\sigma_Y\sigma_{\hat{Y}}\rho(h_{Y_2}(t+1), h_{\hat{Y}_2}(t+1)) + \sum_{k=t-m+1}^{t+1} \left((h_{y_2}(k) - \mu_{h_{y_2}})(h_{y_2}(k) - \mu_{h_{y_2}} - 1) - (h_{\hat{y}_2}(k) - \mu_{h_{\hat{y}_2}})(h_{\hat{y}_2}(k) - \mu_{h_{\hat{y}_2}} + 1) \right) \right) \right) \tag{4.17}$$

Cette équation montre bien que le coefficient de corrélation entre $h_{Y_2}(t+1)$ et $h_{\hat{Y}_2}(t+1)$ est pris en considération par le deuxième RRFR du RR²FR lorsque la fonction d'activation est la fonction sigmoïde. Ceci affirme que la corrélation entre $\hat{Y}(t+1)$ et $Y(t+1)$ est soutenue implicitement par les neurones gaussienne du second RRFR à travers $h_{Y_2}(t+1)$ et $h_{\hat{Y}_2}(t+1)$.

Nous démontrons, dans la suite, comment un RR²FR prendre en considération l'erreur de prédiction e_1 du premier RRFR comme une entrée implicite au deuxième RRFR.

Démonstration 4.3

L'erreur de prédiction $e(t+1)$ du premier RRFR s'écrit comme suit :

$$e_1(t+1) = y(t+1) - \hat{y}(t+1) \tag{4.18}$$

En faisant remplacer $y(t+1) - \hat{y}(t+1)$ par $e(t+1)$ dans l'équation (4.8), nous obtenons l'équation suivante :

$$\text{cov}(Y(t+1), \hat{Y}(t+1)) = \frac{\sum_{k=t-m+1}^{t+1} e_k(t+1) - \mu_Y + \mu_{\hat{Y}}}{m} \quad (4.19)$$

Le coefficient de corrélation est une grandeur qui dépend de la covariance. Par utilisation de l'équation (4.9), Il en résulte :

$$\rho(Y(t+1), \hat{Y}(t+1)) = \frac{\sum_{k=t-m+1}^{t+1} e_k(t+1) - \mu_Y + \mu_{\hat{Y}}}{m\sigma_Y\sigma_{\hat{Y}}} \quad (4.20)$$

Dans la démonstration 4.1, nous avons montré que $\rho(Y(t+1), \hat{Y}(t+1))$ est prise en compte implicitement par le deuxième RFR du RR²FR. L'équation (4.20) montre que $\rho(Y(t+1), \hat{Y}(t+1))$ est en fonction de $e_k(t+1) = y(t+1) - \hat{y}(t+1)$. On peut conclure donc que l'erreur de prédiction du premier RFR est une entrée implicite au deuxième RFR.

Dans le sous-paragraphe suivant, nous nous intéresserons à l'apprentissage des RR²FR. Nous allons présenter les différentes phases d'apprentissage tout mettant l'accent sur la phase d'apprentissage non supervisé. En effet, nous allons proposer un nouvel algorithme d'apprentissage non supervisé.

4.2.2.3- Apprentissage des RR²FR

Normalement l'apprentissage des RFR s'effectue en deux phases : une phase de détermination des centres et rayons d'influence des neurones gaussiens et une phase de calcul des poids des connexions de sortie entre les neurones gaussiens et les neurones de sortie. Pour les RFR, une autre phase localisée en premier ordre est nécessaire pour initialiser les paramètres de neurones bouclés [ZEM 03]. Vu que le RR²FR est une combinaison de deux RFR, les mêmes phases, techniques et méthodes d'apprentissage des RFR seront utilisées.

4.2.2.3.1- Phase 1 : initialisation des paramètres de neurones bouclés

Le travail [ZEM 03] a montré que la plus longue mémoire est obtenue pour $kw_{ii}=2$. Avec cette valeur le neurone bouclé est capable de graver une mémoire d'occurrence d'un événement donné pendant plus de 200 pas de simulation. Il suffit donc, dans la suite, de choisir les valeurs de k et de w_{ii} .

4.2.2.3.2- Phase 2 : détermination des centres et rayons d'influence des neurones gaussiens

Pour la détermination des centres et rayons d'influence des neurones gaussiens, on utilise les vecteurs d'entrée après la phase de bouclage. L'apprentissage est non supervisé. Dans la littérature, l'algorithme de segmentation K-moyens est le plus utilisé pour cette phase. Il permet la minimisation de la somme des erreurs quadratiques entre les vecteurs d'entrée et les centres des prototypes. D'autres heuristiques sont introduites dans la littérature telles que :

l'algorithme RCE (Restricted Coulomb Energy) introduit par [HUD 92] et l'algorithme DDA (Dynamic Decay Adjustment) par [BER 95]. Ces techniques optent à ajuster les rayons d'influence des prototypes en fonctions d'un seuil θ . La différence entre les deux heuristiques est que dans l'heuristique DDA un deuxième seuil θ^+ est ajouté afin de réduire les conflits entre les prototypes. Pour cette phase, l'algorithme K-moyens présente un outil d'apprentissage très efficace et facile.

Le principe de fonctionnement de l'algorithme K-moyens se résume par les étapes suivantes :

Algorithme 4.1

- i. initialisation du nombre de centres k ,
 - ii. initialisation aléatoire des k centres $\{\mu_1, \mu_2, \dots, \mu_k\}$,
 - iii. faire jusqu'à pas de changements :
 - i.1. affectation de chaque point $x \in \chi$ au centre μ_i le plus proche,
 - i.2. calcul des nouveaux centres $\{\mu_1, \mu_2, \dots, \mu_k\}$ de chaque nuage de points,
 - iv. calcul du rayon d'influence de chaque centre avec l'équation (2.32).
-

Cet algorithme présente des inconvénients qui sont :

- la détermination du nombre adéquat de centres ou de prototypes est aléatoire,
- l'initialisation des centres est aléatoire,
- des situations où un nuage de points soit vide peuvent être apparues; dans ce cas, le rayon d'influence de ce centre ne peut être calculé.

Dans nos travaux de thèse, l'algorithme K-moyens sera utilisé. Afin d'éviter ses inconvénients, nous proposons une version améliorée de l'algorithme K-moyens. En effet, l'amélioration consiste à appliquer, en premier lieu, l'heuristique RCE. Puis, on initialise l'algorithme K-moyens par le nombre des classes et les centres résultat du l'heuristique RCE [TEL 09b]. L'heuristique RCE est résumée par l'algorithme suivant :

Algorithme 4.2

```
//Mise à zéro des poids :
pour tout prototype  $k$  de classe  $i$   $\mu_k^i$  faire
     $w_k^i = 0$ 
fin
//Itération d'apprentissage  $x$  de classe  $c$  faire :
    si  $\exists \mu_k^c : \phi_k^c(x) \geq \theta$  alors
         $w_k^c += 1.0$ 
    sinon
        //création d'un nouveau prototype
        ajouter un nouveau prototype  $\mu_{m_c+1}^c$  avec:
```

$$\begin{aligned} \mu_{m_c+1}^c &= x \\ \sigma_{m_c+1}^c &= \max_{i \neq c \wedge 1 \leq j \leq m_i} \{ \sigma : \phi_{m_c+1}^c(\mu_j^i) < \theta \} \\ w_{m_c+1}^c &= 1.0 \\ m_c &+= 1 \end{aligned}$$

fin

//ajuster les zones de conflits

pour tout $i \neq c, 1 \leq j \leq m_i$ **faire**

$$\sigma_j^i = \max \{ \sigma : \phi_i^j(x) < \theta \}$$

fin

fin

4.2.2.3.3- Phase 3 : calcul des poids des connexions de sortie

Cette dernière phase d'apprentissage a le but de calculer les poids des connexions de la couche cachée vers la couche de sortie (apprentissage supervisé). Ces poids sont déterminés soit par renforcement (maximum d'appartenance) dans le cas d'une application en classification, soit par un algorithme des moindres carrés en minimisant une fonction de risque empirique. Dans nos travaux de thèse, nous nous intéressons à la prédiction. Dans la littérature plusieurs méthodes sont présentées telles que la méthode de descente de gradient, la méthode régression linéaire multiple (MLR), la méthode moyenne mobile intégrée autorégressive (ARIMA), etc., [YAO 06].

Dans notre travail, nous avons choisi l'utilisation de la méthode Régression Linéaire Multiple (MLR). La MLR, consiste à résoudre le système suivant :

$$Y = \Phi W + U \tag{4.21}$$

Où :

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix}; W = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_q \end{bmatrix}; \Phi = \begin{bmatrix} 1 & \phi_{11} & \phi_{12} & \cdots & \phi_{1q} \\ 1 & \phi_{21} & \phi_{22} & \cdots & \phi_{2q} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \phi_{p1} & \phi_{p2} & \cdots & \phi_{pq} \end{bmatrix}; U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{bmatrix}$$

Avec :

Y : vecteur des sorties prescrites,

W : vecteur des poids,

U : Vecteur d'erreurs,

q : nombre des classes ou nombre des neurones à base de fonctions radiales,

p : taille d'échantillon d'apprentissage.

Le modèle MLR d'estimation est le suivant :

$$\hat{Y} = \Phi \hat{W} \quad (4.22)$$

Où :

$$\hat{W} = \begin{bmatrix} \hat{w}_0 \\ \hat{w}_1 \\ \hat{w}_2 \\ \vdots \\ \hat{w}_q \end{bmatrix}$$

L'objectif est de minimiser l'erreur :

$$E = Y - \hat{Y} \quad (4.23)$$

Avec :

$$E = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_p \end{bmatrix}$$

E est la valeur d'estimation de U .

Pour calculer \hat{W} , il suffit de résoudre l'équation suivante :

$$\min \sum_{k=1}^p e_i^2 = EE^t = (Y - \Phi \hat{W})^t (Y - \Phi \hat{W}) \quad (4.24)$$

Le minimum de $\sum_{k=1}^p e_i^2$ est atteint lorsque :

$$\frac{\partial(EE^t)}{\partial \hat{W}} = -2\Phi^t Y + 2\Phi^t \Phi \hat{W} = 0 \quad (4.25)$$

Ainsi, \hat{W} peut être résolu par :

$$\hat{W} = (\Phi^t \Phi)^{-1} \Phi^t Y \quad (4.26)$$

4.2.2.4- Application des RR²FR

Après avoir proposé, dans une première étape, notre architecture RR²FR, nous avons montré en seconde étape, l'apport des récurrences au niveau de neurones d'entrée du deuxième RRFR en termes de la prise en considération de la corrélation entre $\hat{Y}(t+1)$ et $Y(t+1)$. Nous avons aussi vu les différentes phases d'apprentissage proposées pour les RR²FR. Dans ce sous-paragraphe, nous allons tester les propriétés dynamiques du Réseau RR²FR à travers deux applications :

- prédiction des valeurs de la série temporelle *MackeyGlass*,
- prédiction d'une concentration de sortie en CO₂ d'un four à gaz.

L'objectif de ces deux applications est l'affirmation de l'apport de notre architecture au niveau de minimisation de l'erreur absolue de prédiction.

4.2.2.4.1- Application des RR²FR à la Série temporelle *MackeyGlass* [TEL 09b]

La série temporelle *MackeyGlass* est caractérisée par l'équation différentielle suivante :

$$\frac{dx(t)}{dt} = -0.1x(t) + \frac{0.2x(t-\Delta)}{1+(x(t-\Delta))^{10}} \quad (4.27)$$

La base de données de cette série comprend 1200 points. Elle est disponible sur le site Web suivant : <http://neural.cs.nthu.edu.tw/jang/benchmark>.

Cette base de données est obtenue par l'utilisation de la méthode de Runge-Kutta d'ordre 4 de l'équation (4.27). Les données ont été obtenues avec les conditions initiales suivantes : $\Delta=17$, $x(0)=1.2$ et $x(t-\Delta)=0$ pour $0 \leq t < \Delta$ avec une fréquence d'échantillonnage de 1 (Zemouri et al., 2003).

4.2.2.4.1.1- Simulation

L'apprentissage ainsi que le test de prédiction du réseau de neurones sont effectués sur un horizon d'un pas : $y(t+1)$ pour le premier RRFR et $y(t+2)$ pour le deuxième RRFR. La population d'apprentissage comprend 500 points à partir du point 118. La population de test comprend le reste des points (du point 619 jusqu'au point 1200).

Les valeurs optimales des paramètres d'apprentissage pour les neurones bouclés de la couche d'entrée sont : $w_{ii}=40$ et $K=0.05$ [ZEM 03].

Comme il a été vu, les résultats de l'algorithme RCE permettent la détermination des centres initiaux et du nombre des classes de l'algorithme K-moyens. Afin de constater la valeur optimale de θ , une application de RRFR est faite tout en utilisant seulement l'algorithme RCE pour la classification. Une variation du paramètre θ entre 0 et 1 avec un pas de 0.01 permet de constater la valeur optimale de θ . Cette dernière permet au RRFR (le premier ou le deuxième) de prédire avec le minimum d'erreur. La figure 4.5 résume la variation de l'erreur de prédiction en fonction de θ . Le minimum de 3,83 % d'erreur sur les deux populations (Apprentissage + Test) correspond à une valeur optimale de $\theta=0,74$.

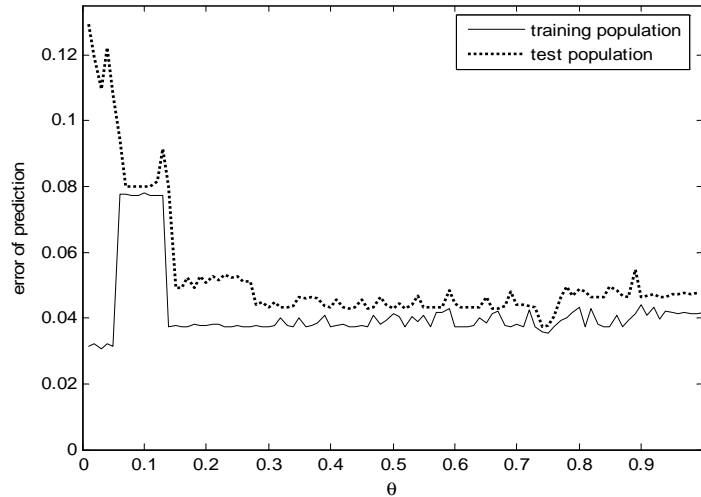


Fig. 4.5- Erreur moyenne relative de prédiction (algorithme RCE) : $f(\theta) = \text{erreur moyenne}$

La figure 4.6 présente le nombre des classes en fonction de θ . La valeur qui correspond à $\theta=0,74$ est $K=7$. Donc, durant la simulation de l'application, l'algorithme K-moyens amélioré sera initialisé par $\theta=0,74$.

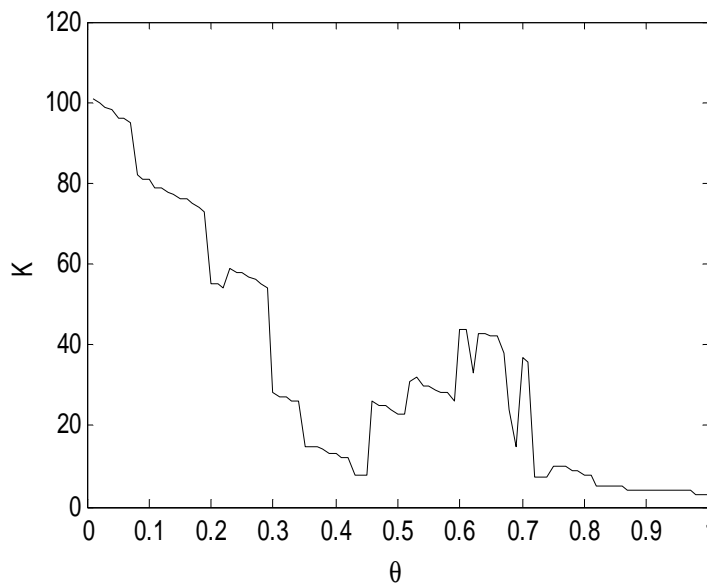


Fig. 4.6- Nombre de neurones cachés (centres) en fonction du paramètre θ

Afin de bien constater l'apport de l'architecture RR^2FR , ainsi que la proposition de l'amélioration de l'algorithme K-moyens, trois simulations ont été réalisées :

- la première est une prédiction par un $RRFR$ utilisant l'algorithme K-moyens pour la phase d'apprentissage non supervisé,
- la deuxième s'agit d'une prédiction par un RR^2FR avec le même algorithme,
- La dernière est une prédiction par un RR^2FR dont l'algorithme d'apprentissage non supervisé est K-moyens amélioré.

Les résultats des simulations sont donnés par les figures 4.7, 4.8 et 4.9 et le tableau 4.1.

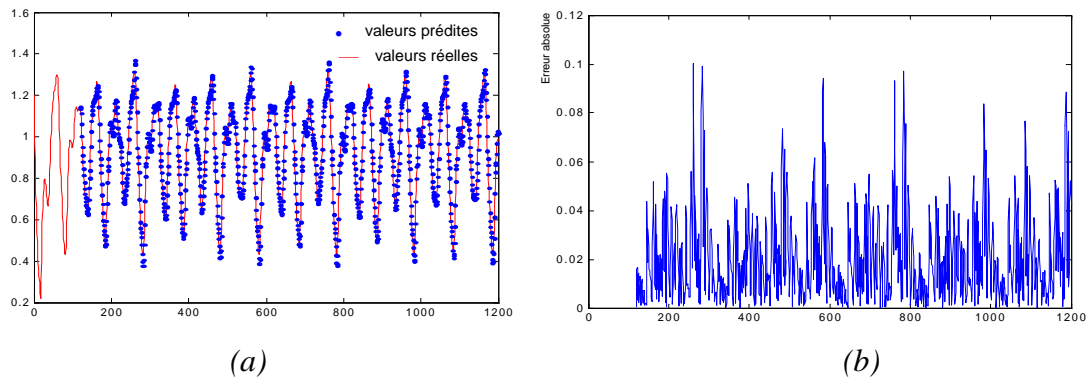


Fig. 4.7- RRFR + K-moyens appliqué à la série temporelle *MackeyGlass*

(a) résultat de la prédiction obtenu
(b) erreur absolue de prédiction

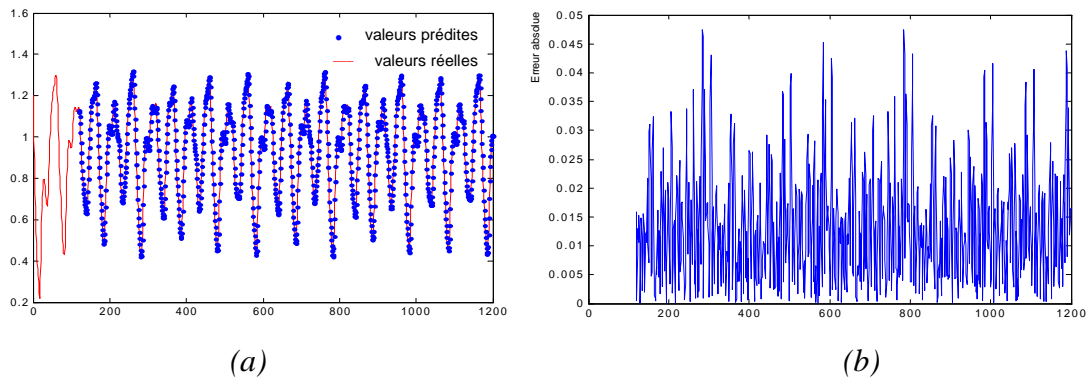


Fig. 4.8- RR²FR + K-moyens appliqué à la série temporelle *MackeyGlass*

(a) résultat de la prédiction obtenu
(b) erreur absolue de prédiction

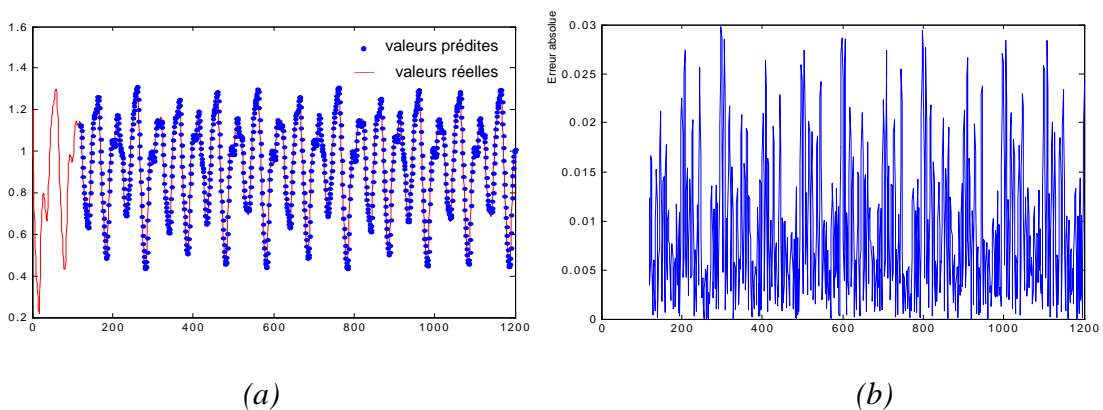


Fig. 4.9- RR²FR+K-moyens amélioré appliqué à la série temporelle *MackeyGlass*

(a) résultat de la prédiction obtenu
(b) erreur absolue de prédiction

Le tableau 4.1 résume les résultats obtenus ainsi que les améliorations acquises.

	Moy ₁	% E ₁	Moy ₂	% E ₂	T. app. (secondes)
RRFR+ K-moyens	0.0334	4.05	0.0337	4.10	0.03121 [*]
RR ² FR+ K-moyens	0.0139	1.64	0.014	1.65	~0.03187 ^{*,†}
RR ² FR+ K-moyens amélioré	0.0084	0.86	0.0086	0.88	~0.03272 ^{*,‡}

Tab. 4.1- Résumé des résultats des simulations pour la série temporelle *MackeyGlass*

Avec :

Moy₁ : la moyenne des erreurs absolues de la population d'apprentissage,

% E₁ : la moyenne de pourcentages des erreurs absolues de la population d'apprentissage,

Moy₂ : la moyenne des erreurs absolues de toute la population (test+apprentissage),

% E₂ : la moyenne de pourcentages des erreurs absolues de toute la population (test+apprentissage),

T. app. : temps d'apprentissage.

4.2.2.4.1.2- Interprétations

En utilisant le RRFR et l'algorithme K-moyens, la moyenne des erreurs absolues est à l'ordre de 0,0337 pour toute la population. L'application de la nouvelle architecture, tout en utilisant le même algorithme d'apprentissage non supervisé, porte une amélioration au niveau de la performance de prédiction. En effet, la moyenne des erreurs absolues s'améliore pour atteindre 0,014. Les résultats des simulations montrent l'efficacité de l'architecture proposée. L'utilisation du bouclage au niveau des neurones de sortie porte une amélioration en terme de l'erreur. Ceci est dû à la corrélation qui existe au niveau de deuxième RRFR du RR²FR entre le résultat obtenu par le premier RRFR pour un instant $t+1$ et l'état réel du système à l'instant $t+1$. D'autre part, l'utilisation de l'algorithme K-moyens amélioré porte une amélioration à l'ordre de 0,054 de la moyenne des erreurs absolues pour toute la population soit 0,77% de la moyenne de pourcentages des erreurs absolues.

L'initialisation de l'algorithme K-moyens par les résultats de l'heuristique RCE, a montré une efficacité énorme. L'approche proposée n'a pas seulement résolu les inconvénients de l'algorithme K-moyens, mais elle a aussi apporté une diminution de l'erreur de prédiction.

Les temps d'apprentissage sont presque identiques pour les trois simulations. Ceci affirme encore l'efficacité des RR²FR.

En conclusion, à travers cette application, un RR²FR est capable d'améliorer les propriétés intrinsèques des RFR classiques.

4.2.2.4.2- Application des RR²FR à la prédiction d'une concentration de sortie en CO₂ d'un four à gaz

Après avoir appliqué notre RR²FR à la série temporelle *MackeyGlass*, nous présentons, dans ce sous-paragraphe, une application de cette version améliorée des RFR au pronostic d'un

* Traitement effectué sur un processeur de 1.77 GHz de fréquence d'horloge.

† Temps d'apprentissage de chaque RRFR de RR²FR

‡ Temps d'apprentissage de chaque RRFR de RR²FR, non incluant le temps d'initialisation par RCE.

système industriel. Le système concerné est le benchmark d'un four à gaz. Cette application est schématisée par la figure 4.10.

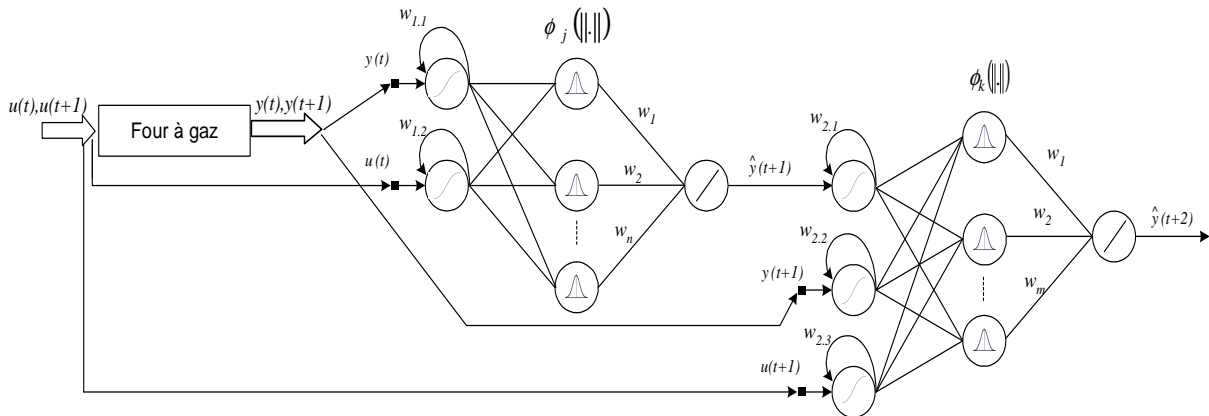


Fig. 4. 10- RR²FR appliqué à un benchmark d'un four à gaz

avec :

$u(t)$: le débit de gaz en entrée à l'instant t

$y(t)$: concentration de sortie en CO₂ à l'instant t

La base de données de cette série comprend 290 points. Elle est disponible sur le site Web suivant : <http://neural.cs.nthu.edu.tw/jang/benchmark>.

Le but de cette application est de prédire la concentration de sortie en CO₂ : $y(t+2)$ à partir de la sortie y aux instants t et $t+1$ et du débit de gaz en entrée u aux instant t et $t+1$.

4.2.2.4.2.1- Simulation

La population d'apprentissage comprend les 145 premiers points. La population de test comprend le reste des points.

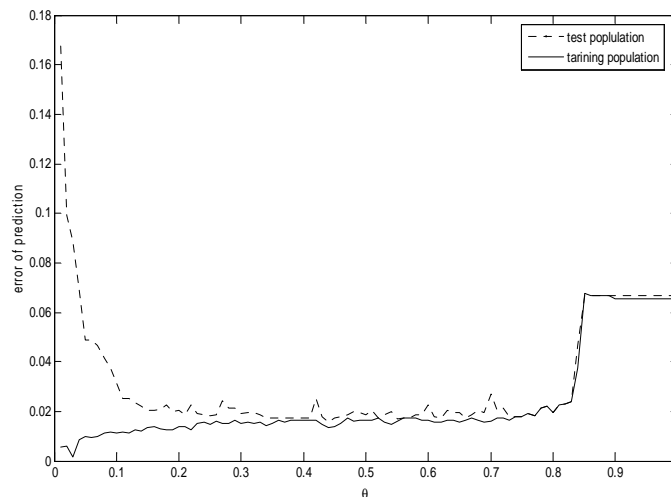


Fig. 4.11- Erreur moyenne relative de prédiction (algorithme RCE) : $f(\theta) = \text{erreur moyenne}$

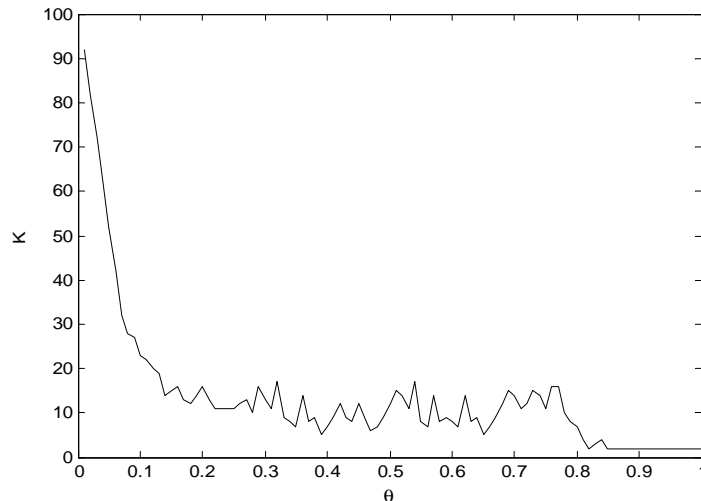


Fig. 4.12- Nombre de neurones cachés (centres) en fonction du paramètre θ

Pour cette application les valeurs optimales des paramètres d'apprentissage sont :

- $w_{ii}=400$ et $K=0.05$ pour les neurones bouclés de la couche d'entrée,
- par variation du paramètre θ entre 0 et 1 avec un pas de 0.01, nous pouvons constater la valeur optimale de θ pour cette application. La figure 4.11 résume la variation de l'erreur de prédiction en fonction de θ . Le minimum de 1,43 % d'erreur sur les deux populations (Apprentissage + Test) correspond à une valeur optimale de $\theta=0,44$. La figure 4.12 présente le nombre des classes en fonction de θ . La valeur qui correspond à $\theta=0,44$ est $K=8$. Donc, durant la simulation de cette application, l'algorithme K-moyens amélioré sera initialisé par $\theta=0,44$.

Comme l'application précédente trois simulations ont été réalisées pour la prédiction de la concentration de sortie en CO_2 : un RRFR utilisant l'algorithme K-moyens pour la phase d'apprentissage non supervisé, un RR^2FR avec le même algorithme et un RR^2FR dont l'algorithme d'apprentissage non supervisé est K-moyens amélioré.

Les figures (Fig. 4.13, Fig. 4.14 et Fig. 4.15) et le tableau 4.2 illustrent les résultats des simulations de cette application.

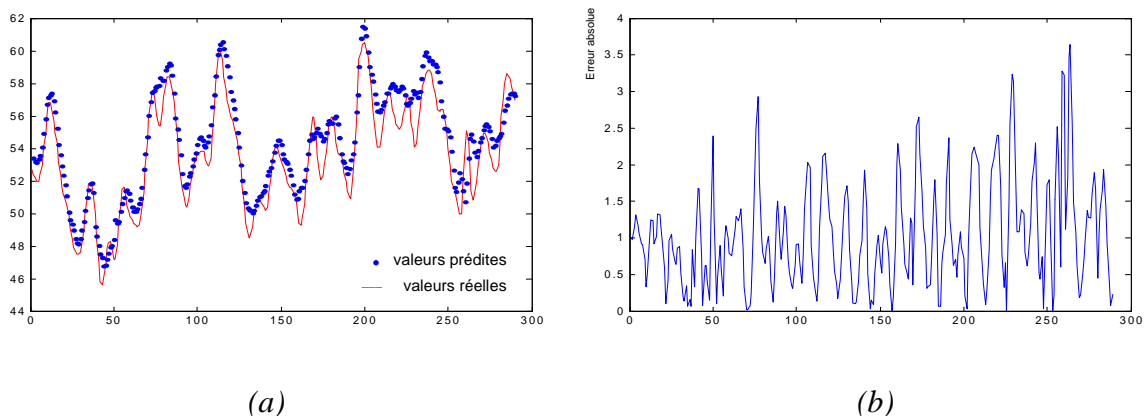
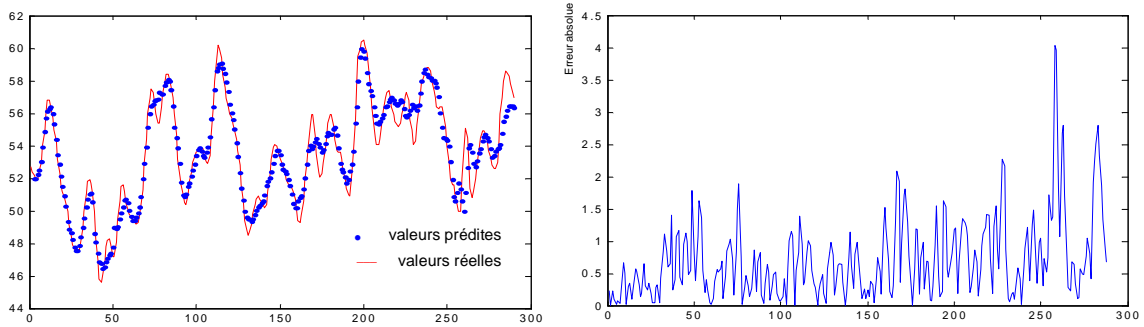


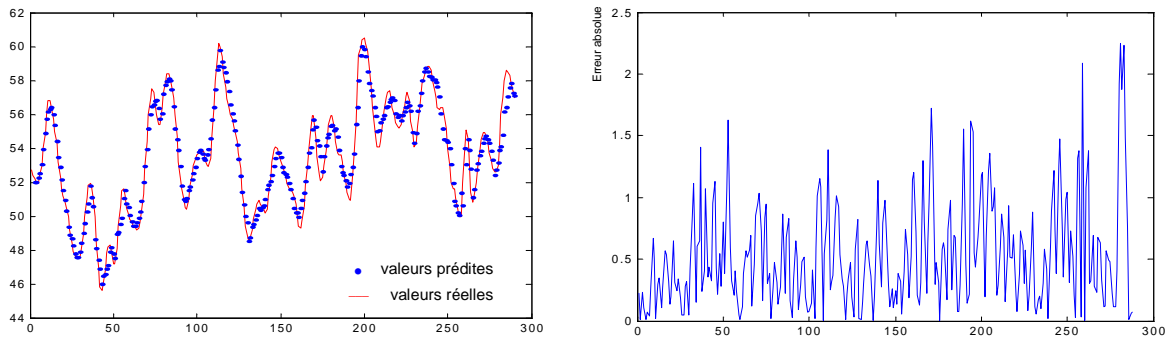
Fig. 4.13- RRFR + K-moyens appliqué à un benchmark d'un four à gaz

(a) Résultat de la prédiction obtenu par le réseau
(b) erreur absolue



(a) (b)
Fig. 4.14- RR²FR+ K-moyens appliqué à un benchmark d'un four à gaz

(a) Résultat de la prédiction obtenu par le réseau
(b) erreur absolue



(a) (b)
Fig. 4.15- RR²FR + K-moyens amélioré appliqué à un benchmark d'un four à gaz

(a) Résultat de la prédiction obtenu par le réseau
(b) erreur absolue

	Moy ₁	% E ₁	Moy ₂	% E ₂	T. app. (secondes)
RRFR+ K-moyens	0.892	1.73	1.052	1.98	0.02091 [§]
RR ² FR+ K-moyens	0.615	1.2	0.809	1.32	~0.02111 ^{§,**}
RR ² FR+ K-moyens amélioré	0.386	0.79	0.490	0.89	~0.02151 ^{§,††}

Tab. 4.2- Résumé des résultats des simulations pour la prédiction de la concentration de sortie en CO₂ d'un four à gaz

Avec :

[§] Traitement effectué sur un processeur de 1.77 GHz de fréquence d'horloge.

^{**} Temps d'apprentissage de chaque RRFR de RR²FR

^{††} Temps d'apprentissage de chaque RRFR de RR²FR, non incluant le temps d'initialisation par RCE.

Moy_1 : la moyenne des erreurs absolues de la population d'apprentissage,

$\% E_1$: la moyenne de pourcentages des erreurs absolues de la population d'apprentissage,

Moy_2 : la moyenne des erreurs absolues de toute la population (test+apprentissage),

$\% E_2$: la moyenne de pourcentages des erreurs absolues de toute la population (test+apprentissage),

$T. app.$: temps d'apprentissage.

4.2.2.4.2.- Interprétations

Les résultats de simulation de prédiction de la concentration de sortie en CO₂ d'un four à gaz par un RR²FR ont amplifié les opportunités de succès de cette architecture. Une amélioration concrète est observée tout en passant d'une simulation à une autre : partant d'un RRFR utilisant l'algorithme K-moyens, puis un RR²FR avec le même algorithme et finalement un RR²FR dont l'algorithme d'apprentissage non supervisé est K-moyens amélioré.

En effet, en utilisant le RRFR et l'algorithme K-moyens, la moyenne des erreurs absolues est à l'ordre de 1,052 pour toute la population. Ensuite, la nouvelle architecture RR²FR est appliquée en utilisant l'algorithme K-moyens classique, une amélioration au niveau de la performance de prédiction est bien remarquée. Par conséquent, la moyenne des erreurs absolues s'améliore pour atteindre 0,809. Ceci montre que le bouclage des neurones d'entrée du deuxième RRFR de RR²FR a entraîné la prise en compte de la corrélation entre $\hat{y}(t+1)$ et $y(t+1)$ par le deuxième RRFR. Ce résultat porte une affirmation de l'apport des RR²FR. L'algorithme K-moyens amélioré appliqué aux RR²FR a provoqué une amélioration à l'ordre de 0,43% de la moyenne de pourcentages des erreurs absolues.

Comme l'application précédente, les temps d'apprentissage sont à-peu-près semblables pour les trois simulations. Ceci montre encore l'efficacité des RR²FR en terme de rapidité de traitement par ordinateurs.

4.2.3- Interprétation générale et apport de l'architecture RR²FR

La fonction pronostic de notre architecture Surveillance-Commande proposée « Surveillance indirecte suite à une commande robuste » a le rôle de prédire les valeurs futures des variables de la commande. Ceci nécessite un outil facile et performant capable de prendre en compte la complexité des systèmes industriels. Le réseau RR²FR a montré sa performance à résoudre une telle problématique. En effet, plus que son architecture non complexe, le temps d'apprentissage d'un tel réseau est à-peu-près identique à celui des RFR et RRFR.

En conclusion, le réseau RR²FR présenté a été testé avec succès sur deux applications. La première est la prédiction des valeurs de la série temporelle *MackeyGlass*. La seconde sur le problème de prédiction d'un système non linéaire : *un benchmark d'un four à gaz*.

Aussi, ce réseau a montré, à travers les deux applications présentées, sa capacité à minimiser l'erreur de prédiction par rapport aux RFR classiques et aux RRFR proposé par [ZEM 03]. Ceci est du :

- aux neurones bouclés au niveau des entrées de chaque RRFR de notre architecture qui jouent le rôle d'une mémoire dynamique,

- à la corrélation entre $\hat{y}(t+1)$ et $y(t+1)$ qui est prise implicitement en considération et qui permet un traitement dynamique de l'erreur $e(t+1) = \hat{y}(t+1) - y(t+1)$. cette erreur lui-même est prise en considération comme une entrée auxiliaire,
- à l'algorithme d'apprentissage non supervisé K-moyens amélioré qui a pu traiter les inconvénients de l'algorithme K-moyens classique.

4.3- Détection des symptômes par suivi des paramètres de loi de commande à travers les RdPTàO : approche qualité-flux

Les RPTàO se caractérisent par le fait que les jetons qui forment le marquage des places ne sont pas des entités atomiques et indistinctes. Néanmoins, ils peuvent être distingués les uns des autres et portent une valeur. En partant de l'hypothèse que les défaillances d'une ressource peuvent être préconçues à partir des déviations temporelles et que ces dernières sont tabulées, cet outil est largement utilisé pour la modélisation des systèmes à événements discrets afin d'assurer la détection des symptômes des défaillances [ZAM 97], [BOU 03], [Minca et al 03], [DAS 03a], [DAS 03b], [MHA 06].

Notre objectif, dans cette section, est de développer une méthode multicritères de détection des symptômes des défaillances par l'exploitation de l'outil Réseaux de Petri Temporels à Objets (RdPTàO). L'idée sous-jacente est de développer le modèle de détection approuvé par M. Combacau [COM 91] afin de l'évoluer pour qu'il soit apte à détecter non seulement les dérives temporelles (flux) mais aussi celles non-temporelles (qualité).

4.3.1- Modèle de Combacau pour la détection des symptômes temporels [COM 91]

Le modèle de [COM 91] pour la détection des symptômes temporels fait appel à deux Réseaux de Petri Temporels à Objets (RdPTàO). Le premier est le réseau de référence, il modélise les états de fonctionnement correct du système. Ce réseau sert de référence pour le modèle de commande modélisé par un second RdPTàO appelé réseau de commande.

Les fonctions suivi et détection sont assurées par la coopération et la synchronisation existantes entre le modèle de référence et celui de commande. Ces derniers évoluent alors en parallèle. Une communication directe se réalise entre le modèle de commande et les capteurs et/ou les actionneurs installés sur les points de référence de début/fin des activités.

En fonctionnement normal avant chaque requête, la commande consulte le modèle de référence afin d'en examiner la convenance vis-à-vis des contraintes du procédé. Après la transmission d'un ordre, elle se met alors en attente d'un compte-rendu d'exécution. En cas de réception convenable (vis-à-vis d'une fenêtre temporelle) de ce compte-rendu, l'information est envoyée vers le modèle de référence pour la mise à jour de l'image du procédé. Si un symptôme est détecté, seul le modèle de référence est mis à jour. La détection des symptômes se réalise par l'implantation de mécanisme de chien de grade (CG) qui permet d'apparaître les déviations temporelles.

En effet, les RdPTàO permettent une modélisation simple et efficace du mécanisme de chien de garde. La figure 4.16 présente la modélisation du mécanisme de chien de garde qui permet la détection d'un symptôme temporel par l'extraction du jeton d'une séquence normale s'il séjourne plus de τ_i unités de temps dans la place P_i .

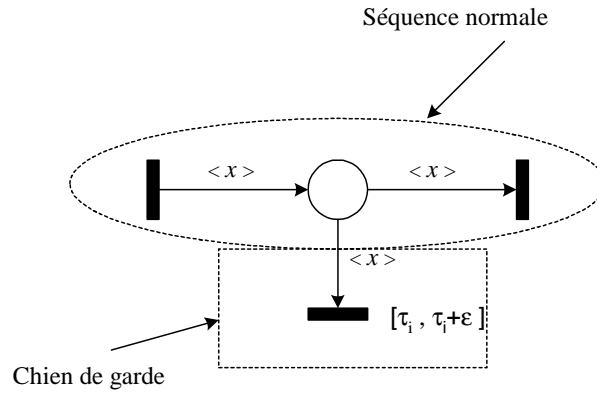


Fig. 4.16- Spécification du mécanisme d'un chien de garde

La figure (Fig. 4.17) présente la structuration de deux RdPTàO (référence et commande) ainsi que les liens entre eux. Les places C_i ($i \in \mathbb{N}$) présentent les canaux de communication entre le deux réseaux. Pour le modèle commande, l'entrée de chaque activité est modélisée par deux transitions Tdc_1 et Tdc_2 . Lorsque Tdc_1 est franchie, ceci signifie que l'activité A_i assurée par la ressource R_i est prête à être déclenchée. Le mécanisme CGd permet la détection d'un symptôme de type 1 : *date de début au plus tard* est dépassée. Si aucun symptôme n'est détecté, les transitions Tdr et Tdc_2 seraient franchies d'où le début de l'activité A_i . A ce niveau un autre mécanisme de chien de garde est localisé sur la place A_i du modèle référence. Il permet de détecter un symptôme de type 2 : *date de fin au plus tard* est dépassée. Si aucun symptôme n'est détecté, la transition Tfc_1 et immédiatement Tfr seraient franchies. Ceci signifie que l'activité A_i de la ressource courante est terminée.

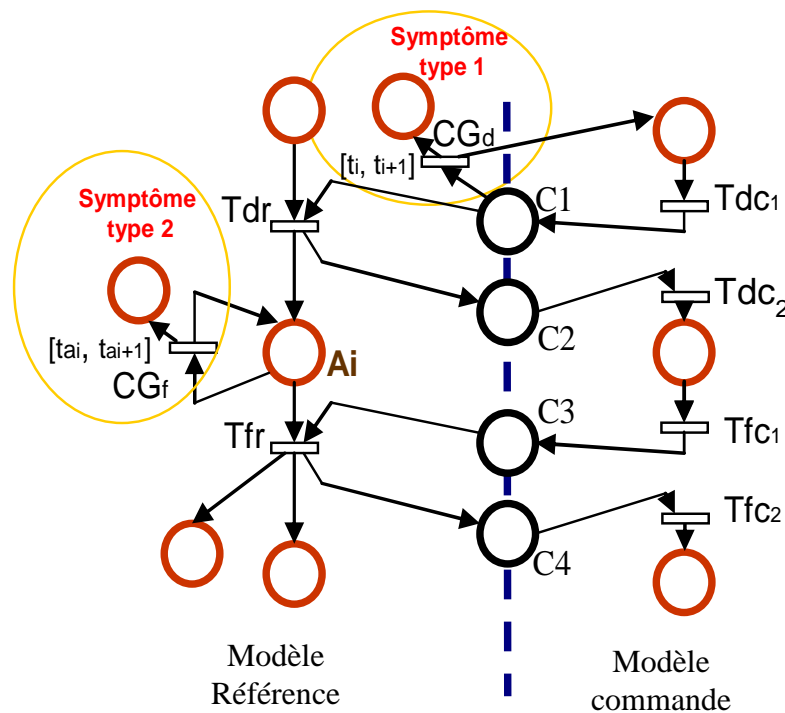


Fig. 4.17- Modèle Suivi-Détection des symptômes temporels [COM 91]

Où :

DEB_{tard} : date de début au plus tard,

FIN_{tard} : date de fin au plus tard,

$[t_i, t_{i+1}]$: intervalle alloué à la transition de CGd ,

$[ta_i, ta_{i+1}]$: intervalle alloué à la transition de CGf .

Les chiens de garde doivent remplir les conditions suivantes :

- La transition CGd doit être franchie à la date absolue $DEB_{tard} = FIN_{tard} - Di = ti$; ($i \in \mathbb{N}$),
- La transition CGf doit être franchie à la date absolue $FIN_{tard} = tai$ ($i \in \mathbb{N}$).

Ce modèle a été exploité ultérieurement comme une base pour plusieurs chercheurs dans le cadre de développement de l'ingénierie de conduite Systèmes Automatisés de Production tels que les travaux [ZAM 97], [BOU 03], [Minca et al 03], [DAS 03a], [DAS 03b], [MHA 06] , etc.

Par rapport au modèle de M. Combacau [COM 91] et aux travaux développés postérieurement, nous cherchons à proposer, pour notre architecture de Surveillance-Commande, un modèle Suivi-Détection qui permet la génération des déviations de types temporels et non-temporels afin de détecter les symptômes des défaillances. L'outil RdPTàO est capable de résoudre cette problématique vu leurs propriétés intrinsèques :

- prise en compte des contraintes logiques,
- prise en compte des contraintes temporelles,
- création d'objets par franchissement,
- modélisation des politiques décisionnelles locales,
- absence de blocage.

4.3.2- Modèle de Suivi-Détection multicritères : Qualité-flux [TEL 08c]

La fonction Suivi-Détection de notre architecture de Commande-Surveillance comprend deux modèles de Suivi-Détection multicritères. Le premier permet de suivre les états provenant du bloc pronostic afin de détecter les probables symptômes. Il est appelé *modèle Suivi-Détection estimé*. Le second consiste à suivre les états réels du système afin de détecter les dépassements des seuils de détection. Il est appelé *modèle Suivi-Détection réel*. Pour notre cas, comme il est indiqué dans le premier chapitre, les deux seuils de détection sont la borne inférieure et la borne supérieure de la marge de robustesse passive. Les deux modèles sont mis en redondance active. La redondance permet de masquer les erreurs de prédiction. Nous allons s'intéresser, dans la suite de cette section, à la conception d'un modèle type qui va être intégré dans le bloc suivi-détection en double : modèle Suivi-Détection réel et modèle Suivi-Détection estimé.

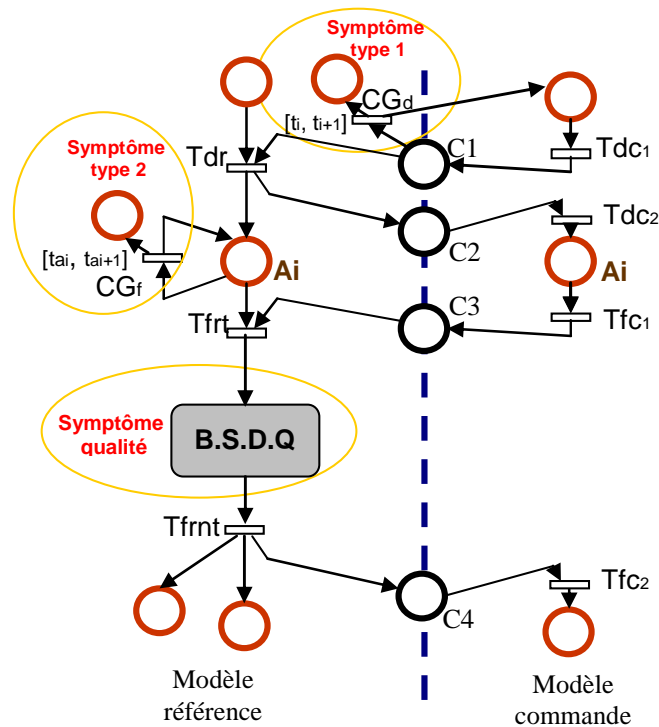


Fig. 4.18- Modèle de Suivi-Détection multicritères : Qualité-flux

La démarche qu'on va suivre dans les lignes qui suivent pour élaborer notre modèle de Suivi-Détection multicritères consiste à exploiter le modèle de [COM 91] comme une base. Ceci par l'augmentation des aptitudes du modèle de [COM 91] en lui faisant muter pour qu'il soit capable d'intégrer aussi des contraintes logiques non temporelles. Pour cette raison, nous intégrons un bloc de Suivi et de Détection Qualité (B.S.D.Q) à ce modèle (Fig. 4.18). Ce bloc est localisé sur le modèle référence à la sortie de l'activité A_i . Il ne se sensibilise qu'après le franchissement de la transition T_{frt} qui traduit la fin de A_i , vu que dans la plupart des industries, le processus de contrôle qualité (par capteur ou par échantillonnage) ne s'effectue qu'après la fin des activités de transformation de matière. Dans ce qui suit, nous expliquons, à travers un exemple, le principe de modélisation d'un tel bloc.

Supposons un poste de travail d'un système de production désigné a subi une transformation (activité A_i) à la pièce P . Cette pièce est caractérisée par deux indicateurs de qualité a et b qui devront être respectés. A la fin de l'activité A_i , cette pièce P est considérée comme un produit semi-fini. Pour qu'elle passe à l'étape suivante de processus de production (c.-à-d. à l'activité A_{i+1}), il faut que les contraintes mises sur a et b soient respectées. Sinon, la pièce serait retirée pour être rejetée ou pour être recyclée. Au même temps, un symptôme de défaillance future sera réclamé. Les intervalles de conformité des a et b sont respectivement $Int_a = [a_{min}, a_{max}]$ et $Int_b = [b_{min}, b_{max}]$.

Le bloc Q.S.D.Q de cet exemple est présenté par la figure 4.19.

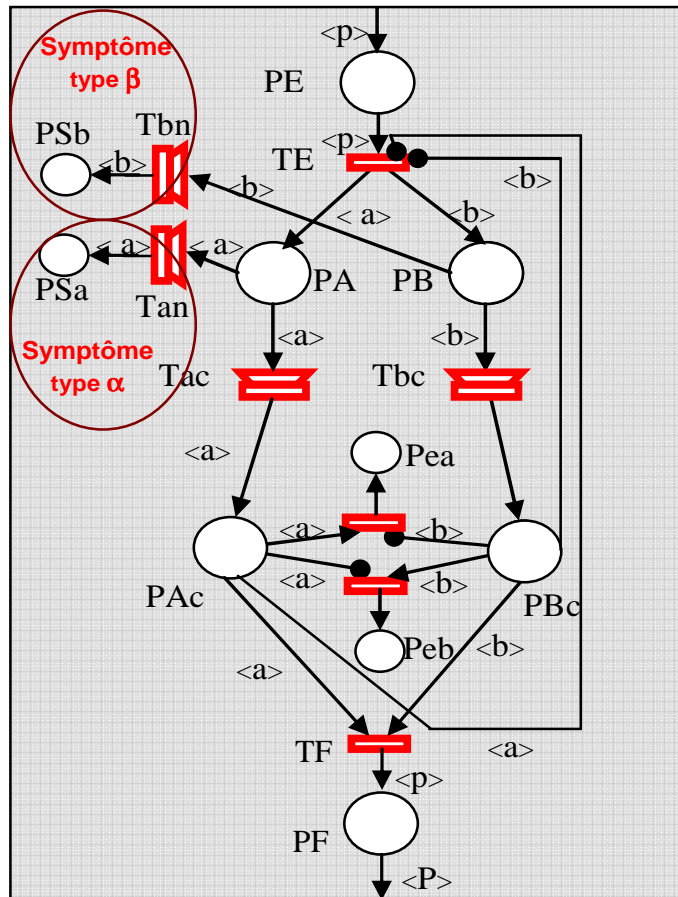


Fig. 4.19- B.S.D.Q est modélisé par un RdPTàO

Le principe de fonctionnement du bloc B.S.D.Q est décrit dans les lignes qui suivent :

Nous considérons :

- Les classes des objets :

P : - numéro : entier typée par p comme variable

A : - a_{max} : réel typée par a comme variable

 - a_{min} : réel

B : - b_{max} : réel typée par b comme variable

 - b_{min} : réel

- Les conditions :

$$a \in [a_{min}, a_{max}] \quad (C1)$$

$$a \notin [a_{min}, a_{max}] \quad (C2)$$

$$b \in [b_{min}, b_{max}] \quad (C3)$$

$$b \notin [b_{min}, b_{max}] \quad (C4)$$

Dès que la pièce p arrive dans la place PE , la transition TE , qui permet la création d'objet par franchissement, se sensibilise, d'où la naissance de deux nouveaux objets A et B sur le réseau. A et B se localisent immédiatement et respectivement dans les places PA et PB .

Si les conditions $(C1)$ et $(C3)$ sont satisfaites, par les franchissements de Tac et Tbc , les objets A et B se déplacent au même temps et respectivement vers PAc et PBc , puis immédiatement vers PF par franchissement de la transition TF . Ceci signifie que les paramètres qualité a et b sont respectés, d'où on peut passer à l'activité $Ai+I$.

Si les conditions $(C2)$ et $(C4)$ sont satisfaites, les transitions Tan et Tbn se sensibilisent. Les objets A et B se déplacent au même temps et respectivement vers PAc et PBc . cet état signifie la naissance de deux symptômes qualité de type α et de type β .

Si les conditions $(C1)$ et $(C4)$ sont satisfaites, nous signalons seulement le symptôme de type β .

Si les conditions $(C2)$ et $(C3)$ sont satisfaites, nous signalons seulement le symptôme de type α .

Les places Pea et Peb permettent le réglage du processus de suivi et détection, au cas où une des deux contraintes de conformités est non respectée. Pea retire l'objet $\langle a \rangle$ si la contrainte mise sur b est non respectée. Similairement, Peb retire l'objet $\langle b \rangle$ si la contrainte mise sur a est non respectée. Ceci est assuré par l'exploitation du principe des arcs inhibiteurs.

4.3.3- Application

Pour fabriquer un nombre n des boîtes métalliques selon les normes exigées, une industrie de fabrication des tôles pour le secteur agroalimentaire a la gamme de production suivante :

Une tôle imprimée de grand format, prise du stock, doit passer par les étapes des transformations suivantes : cisailage, enroulage, soudage, rechampissage intérieur, rechampissage extérieur, bordage, moulurage et finalement sertissage. Après le cisailage des tôles imprimées, les petites tôles seront stockées dans un stock S . les petites tôles seront entraînées par un ouvrier sur l'enrouleuse. A partir de cette étape, le transfert de produit semi-fini d'un poste de travail vers un autre est assuré par les tapis roulants.

Les paramètres significatifs de qualité que nous considérons, pour une boîte, sont :

- Pour le cisailage : hauteur ($h0$), largeur (l), équerrage (e),
- Pour bordage : bordure (bi , bs), hauteur ($h1$)
- Pour moulurage : hauteur ($h2$), profondeur (m)
- Pour sertissage (figure 5) : hauteur de la boîte finie (hf), hauteur de serti (hs), crochet de fond (cf), crochet de corps (cc), croisure a (c) et espace intérieur b (ei).

Pour un fonctionnement normal du procédé la valeur de chaque paramètre doit être comprise dans un intervalle donné.

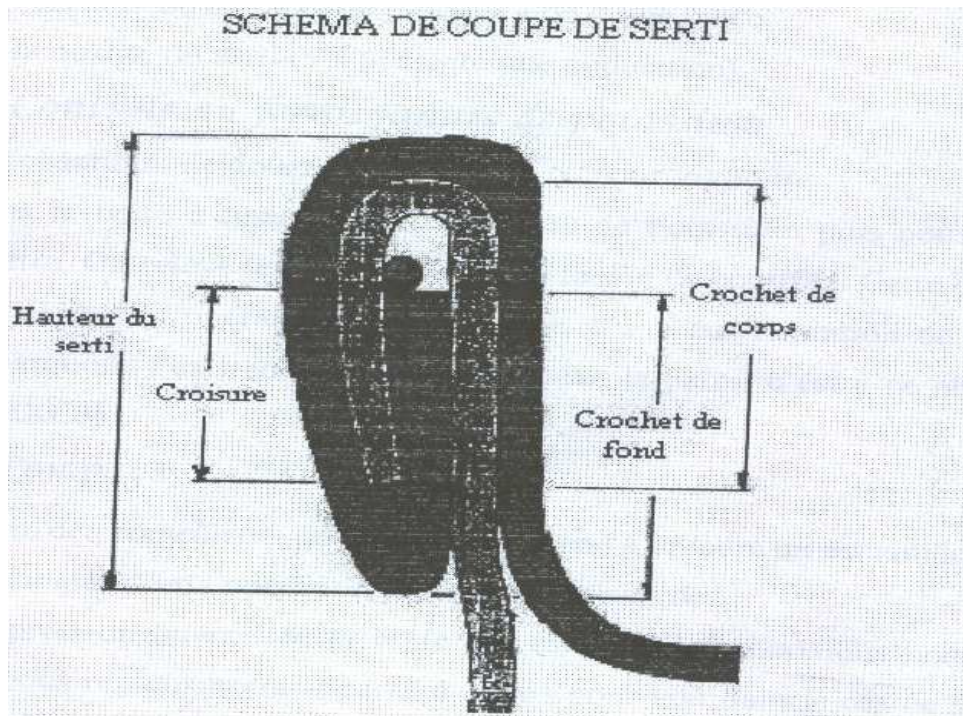


Fig. 4.19- Schéma de coupe de serti

En prenant compte du processus de contrôle qualité, nous avons décomposé le système de production en quatre phases :

- phase 1 : cisailage,
- phase 2 : enroulage, soudage, rechampissage intérieur et rechampissage extérieur,
- phase 3 : bordage et moulurage,
- phase 4 : sertissage.

Le contrôle qualité s'achève à la fin de chaque phase sauf la deuxième. Chacune de toutes les autres phases est suivie par un contrôle qualité assurée par un technicien. Les paramètres contrôlés se répartissent donc comme suit :

- phase 1 : hauteur ($h0$), largeur (l), équerrage (e),
- phase 3 : bordure (bi , bs), hauteur ($h1$), Pour hauteur ($h2$) et profondeur (m)
- phase 4 : hauteur de la boîte finie (hf), hauteur de serti (hs), crochet de fond (cf), crochet de corps (cc), croisure a (c) et espace intérieur b (ei).

Le modèle de Suivi-Détection multicritères (Qualité-flux) de ce système de production est présenté par la figure 4.20. En effet, comme il est indiqué, au début de cette section, ce modèle est présenté dans le bloc Suivi-Détection en double.

Les chiens de garde sont implantés au niveau des tapis roulants au niveau du modèle référence. D'autre part, trois B.S.D.Q sont localisés : B.S.D.Q.1 à la fin de la première phase, B.S.D.Q.2 à la fin de la troisième phase et finalement B.S.D.Q.3 à la fin de la quatrième phase.

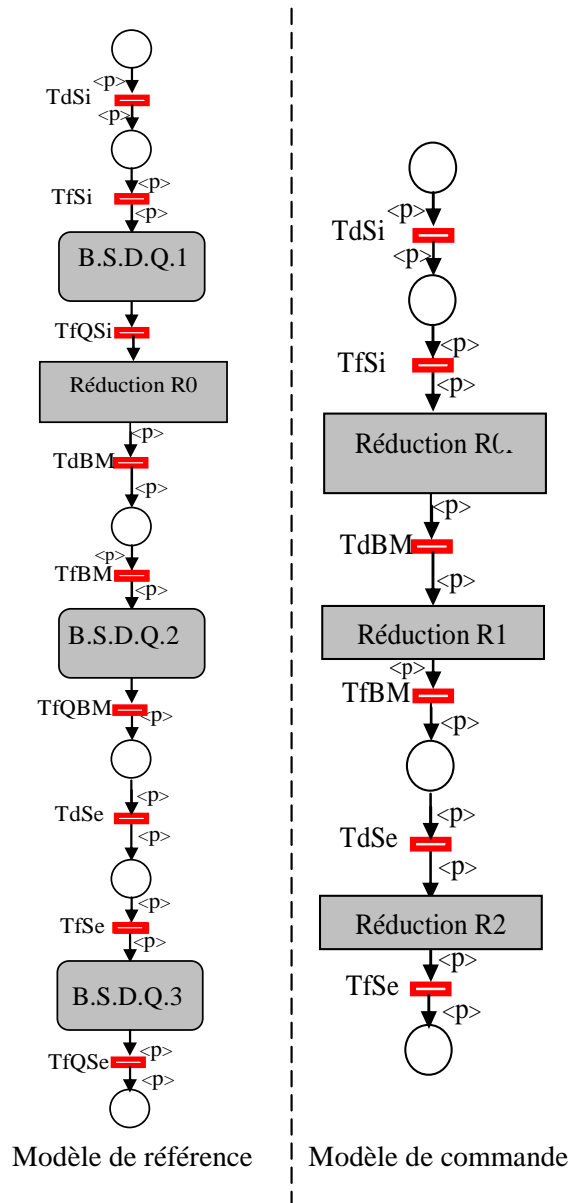


Fig. 4.20- Modèle de Suivi-Détection multicritères (Qualité-flux) d’une gamme fabrication des tôles: modèle de référence et modèle de commande

Avec :

Réduction R0r : réduction de modélisation de la phase deux,

Réduction R1 : réduction de modélisation de la phase trois,

Réduction R2 : réduction de modélisation de la phase quatre,

TdSi : transition d’entrée de l’activité cisailage,

TfSi : transition de sortie de l’activité cisailage avant la phase de contrôle qualité,

TfQSi : transition de sortie de l’activité cisailage après la phase de contrôle qualité,

TdBM : transition d’entrée de la phase 3,

TfBM : transition de sortie de l’activité de la phase 3 avant le contrôle qualité,

TfQBM: transition de sortie de l’activité de la phase 3 après de contrôle qualité,

TdSe : transition d'entrée de l'activité sertissage,

TfSe : transition de sortie de l'activité sertissage avant la phase de contrôle qualité,

TfQSe : transition de sortie de l'activité sertissage après la phase de contrôle qualité.

A titre d'exemple nous illustrons la figure 4.21 qui présente le bloc B.S.D.Q.3 modélisant six paramètres qualité : hf , hs , cf , cc , c et ei .

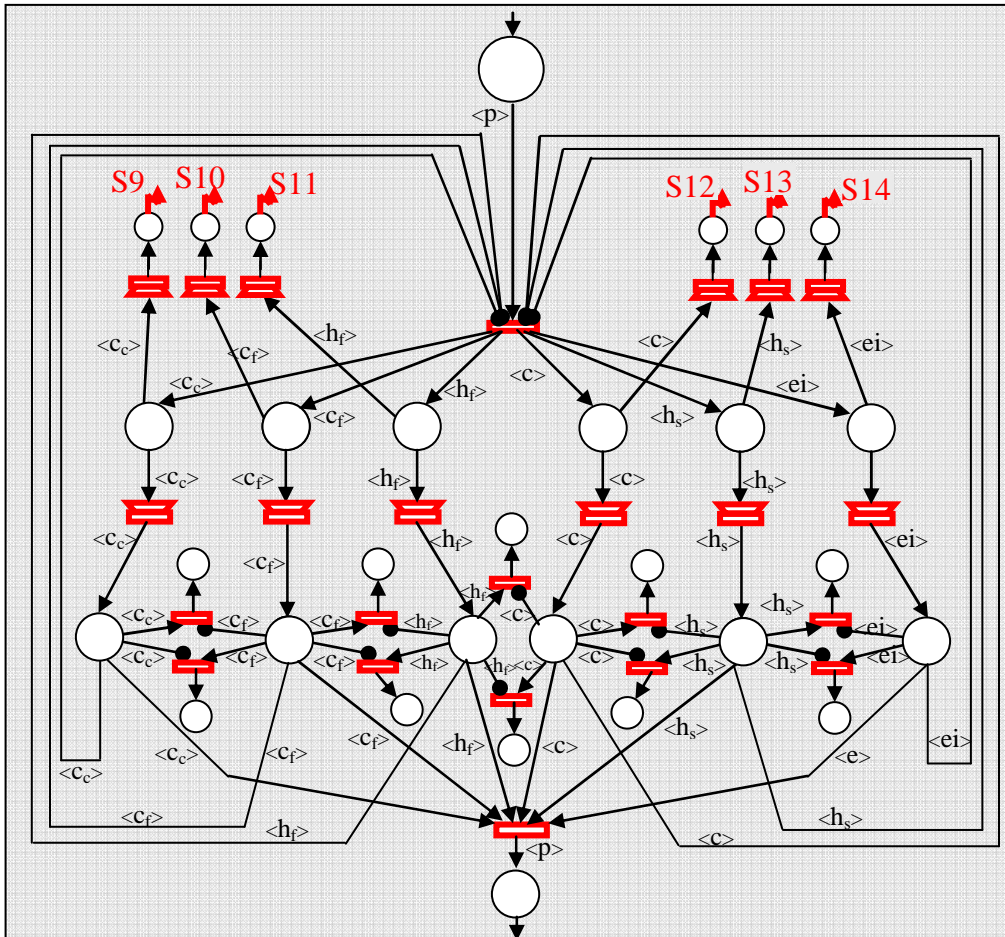


Fig. 4.21- B.S.D.Q.3

Avec :

$S9$: symptôme de type 9 dû à un dépassement de l'intervalle de conformité de cc ,

$S10$: symptôme de type 10 dû à un dépassement de l'intervalle de conformité de cf ,

$S11$: symptôme de type 11 dû à un dépassement de l'intervalle de conformité de hf ,

$S12$: symptôme de type 12 dû à un dépassement de l'intervalle de conformité de c ,

$S13$: symptôme de type 13 dû à un dépassement de l'intervalle de conformité de hs ,

$S14$: symptôme de type 14 dû à un dépassement de l'intervalle de conformité de ei .

Les étiquettes des transitions ainsi que des places ne sont pas présentées vu la complexité de la figure. Néanmoins, on conserve le même principe de modélisation et de nominations présenté par la figure 4.19.

4.3.4- Interprétation générale et apport de Modèle de Suivi-Détection proposé

Afin de résoudre notre problématique de suivi et détection des paramètres à caractères temporels et non temporels, nous avons proposé, par l'exploitation de l'outil RdPTàO, un modèle Suivi-Détection bi-critère. Tout en partant du modèle temporel proposé par [COM 91], le modèle de Suivi-Détection proposé permet la détection des symptômes des défaillances par suivi des paramètres temporels et non-temporels. Les paramètres concernés sont ceux de la loi de commande de notre architecture Surveillance-Commande « Surveillance indirecte suite à une commande robuste ». L'intégration des blocs B.S.D.Q permet la détection des symptômes suite à la non-conformité des paramètres qualité.

4.4- Synchronisation entre la fonction Pronostic et la fonction Suivi-Détection

Les deux blocs pronostic et Suivi-Détection sont deux fonctions clés du modèle surveillance notre architecture de Surveillance-Commande. Les outils de surveillances à exploiter par ces deux blocs sont choisis, développés et améliorés dans les sections précédentes. Nous achevons ce chapitre par la mise en interaction de ces deux blocs tout en essayant d'expliquer le principe de synchronisation et d'échanges d'informations entre les RR²FR et le modèle de Suivi-Détection multicritères (Qualité-flux).

4.4.1- Principe de communication

Les informations provenant du procédé et qui portent sur l'état du produit en terme de performance qualitative et quantitative passe par le bloc pronostic pour prédire l'état futur du système. Ce dernier est traduit par les paramètres qualité et flux spécifiant la loi de commande robuste du bloc commande.

Pour le modèle RdPTàO de Suivi-Détection estimé, la fonction suivi-détection ne peut être déclenchée que si les informations produites par le RR²FR sont prêtes sur les entrées du modèle (Fig. 4.22). Par contre, le modèle RdPTàO de Suivi-Détection réel fonctionne en temps réel.

Les entrées des deux modèles RdPTàO de Suivi-Détection (estimé et réel) se présentent sous la forme d'une requête Σ_e (respectivement Σ_r) des signaux de synchronisations $\{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_n\}$ (respectivement $\{s_1, s_2, \dots, s_n\}$) provenant de la sortie de RR²FR (respectivement des comptes rendus traduisant l'état réel du système). Ces signaux sont les valeurs prédites (respectivement les valeurs réelles) des paramètres de la loi de commande robuste. Ces deux requêtes des signaux vont être réparties sur l'entrée de chaque modèle RdPTàO.

A cet instant, le modèle suivi-détection estimé commence la simulation pour juger l'état futur du système. Vu la structure de notre modèle RdPTàO de Suivi-Détection définie dans la section précédente, si on détecte une déviation temporelle, le processus de simulation s'arrête en annonçant un symptôme probable de type 1 (date de début au plus tard est dépassée) ou de type 2 (date de fin au plus tard est dépassée). Dans ce cas, il n'est plus utile de continuer la simulation des paramètres de produit « *qualité* » vu les objectifs de notre architecture Surveillance-Commande qui cherche à éviter la maintenance corrective en faisant maintenir la robustesse passive de tous les paramètres temporels et non temporels de système.

Par contre, le modèle suivi-détection réel traite l'état réel du système en parallèle avec l'avancement de processus de production. Les comptes rendus provenant du procédé assurent

la synchronisation en temps réel entre l'état du procédé et le modèle suivi-détection réel. Des symptômes temporels et non-temporels peuvent être constatés au fur et à mesure.

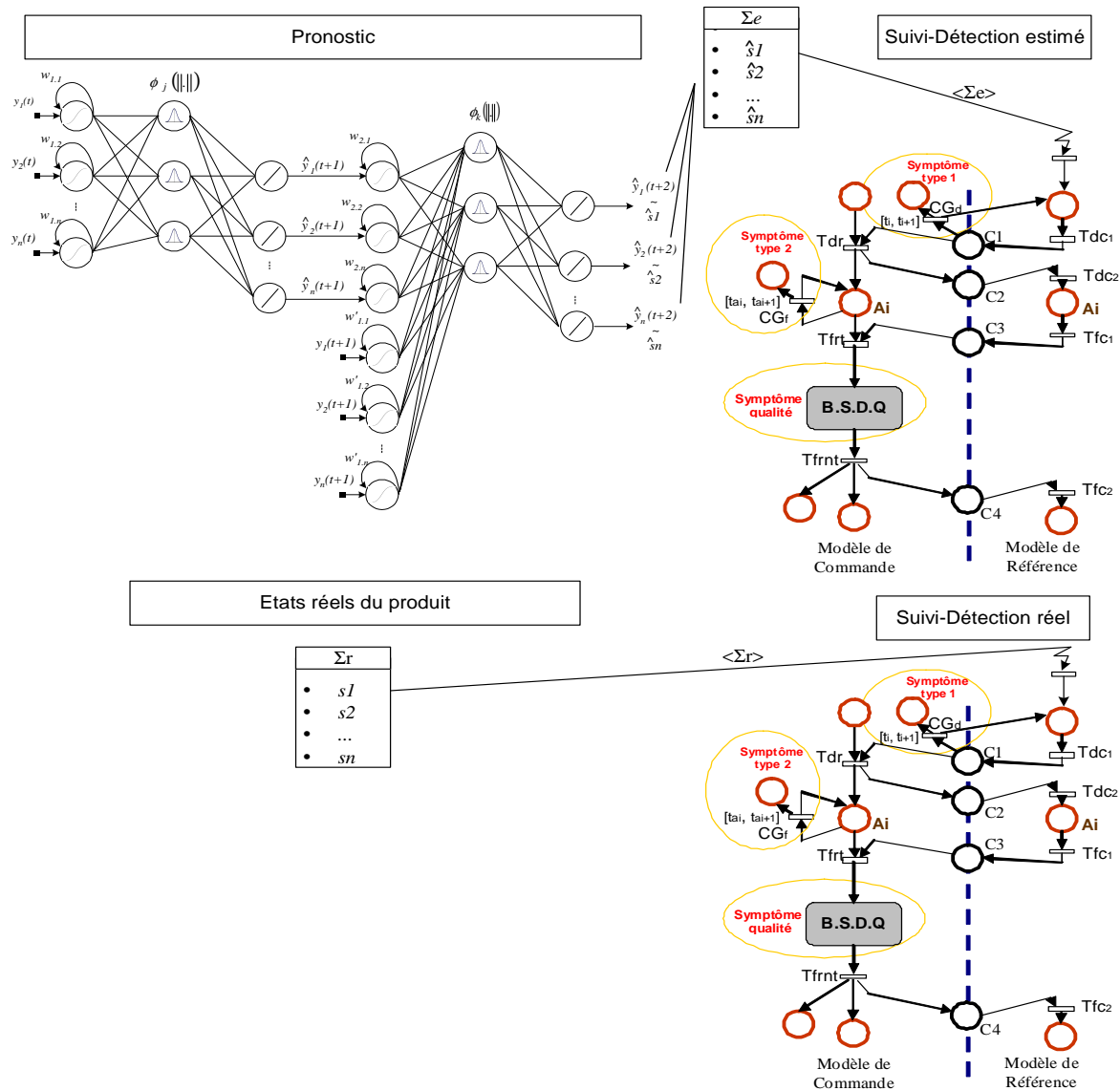


Fig. 4.22- Synchronisation entre le bloc Pronostic et le bloc Suivi-Détection

4.4.2- Application

Nous prenons ici l'exemple de benchmark d'un four à gaz présenté dans la section précédente pour la confirmation de l'apport de l'architecture RR²FR.

Ce four à gaz est exploité par une industrie de poterie. Afin de respecter les normes de qualité environnementale (NQE). Les responsables ont mis des préférences traduites par des marges de conformité pour que la concentration de sortie en CO₂ ne pollue pas l'environnement.

Considérons que la marge de robustesse passive de la concentration de sortie en CO₂, présentant le seul paramètre qualité de ce système, est comprise entre 46,5 et 59,5. Nous supposons que la base de données de l'application (4.2.2.4.2-) a été extraite lors de processus de cuisson d'une quantité de 290 produits « Π ». Nous supposons aussi que la prédiction du paramètre temporel de cuisson pour les 290 produits est toujours dans l'intervalle de la

robustesse passive. La figure 4.23 présente le bloc Suivi-Détection synchronisé avec le bloc Pronostic et les états réels du produit.

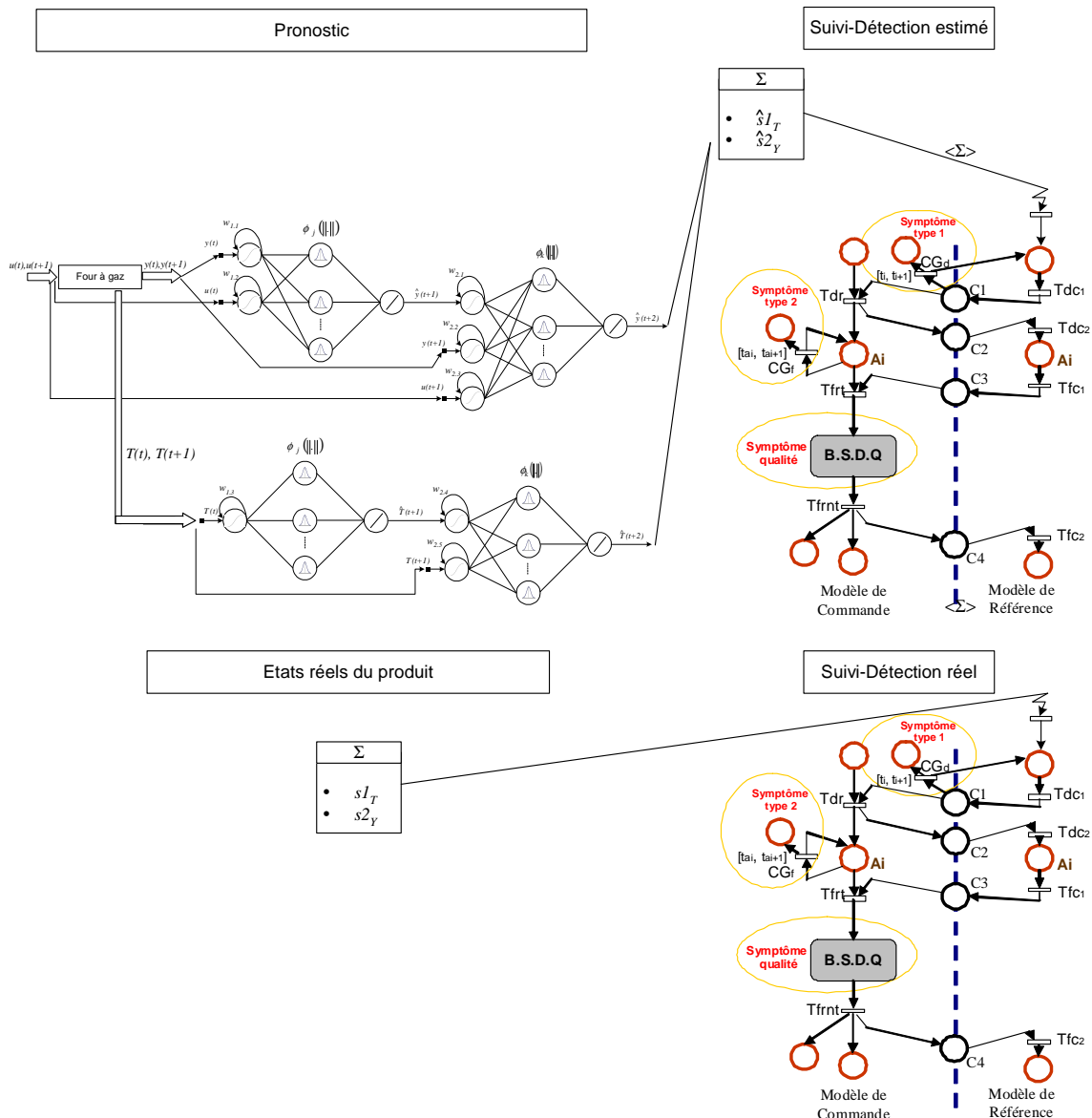


Fig. 4.23- Synchronisation entre le bloc Pronostic et le bloc Suivi-Détection pour l'application de benchmark d'un four à gaz

Nous distinguons ici deux modèles RR^2FR . Le premier est destiné pour la prédiction de la concentration de sortie en CO_2 ($\hat{y}(t+2)$). Le deuxième a le rôle de prédire le temps de cuisson $\hat{T}(t+2)$.

La requête Σ_e comprend deux informations :

$\hat{s}I_T$: signal de synchronisation relatif au temps de cuisson prédit,

$\hat{s}I_Q$: signal de synchronisation relatif à la concentration de sortie en CO_2 prédite.

La requête Σ_s comprend deux informations :

sI_T : signal de synchronisation relatif au temps de cuisson réel,

SI_Q : signal de synchronisation relatif à la concentration de sortie en CO₂ réelle.

La figure 4.24 présente la prédiction de la concentration de sortie de CO₂ par le RR²FR.

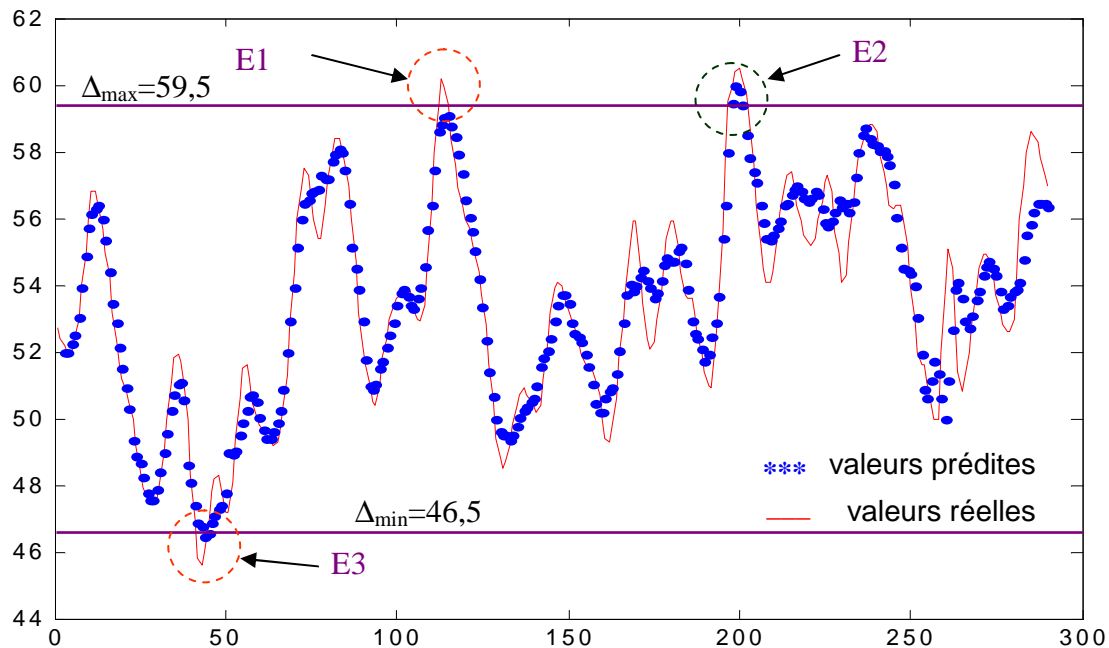


Fig 4.24- Analyse de comportement de prédiction de la concentration de sortie de CO₂ par le RR²FR

Les deux seuils de détection Δ_{min} et Δ_{max} sont les marges de robustesse. Si une valeur prédite soit à l'extérieur de l'intervalle $[\Delta_{min}, \Delta_{max}]$ nous devons recourir au processus de diagnostic. En effet, nous observons sur cette figure deux types des symptômes. Le premier présente les points anormaux détectés par le modèle Suivi-Détection estimé. L'ensemble des points encadrés par *E2* présente ce type des symptômes. Le deuxième type concerne les symptômes qui sont détectés par le modèle Suivi-Détection réel. L'ensemble des points encadrés par *E1* et *E3* présente ce type des symptômes.

4.5- Conclusion

Après avoir déterminé les outils qu'on va utiliser pour les blocs pronostic et suivi-détection dans le chapitre deux, nous avons développé, dans les sections précédentes de ce chapitre, ces outils pour qu'ils soient adaptables à notre architecture de Surveillance-Commande.

En effet, pour le pronostic, nous avons proposé une architecture de réseau de neurones appelée RR²FR. Elle présente une amélioration de l'architecture RFR proposée par [ZEM 03]. Ceci est affirmé en termes de minimisations de l'erreur absolue de prédiction pour les deux applications : prédiction des valeurs de la série temporelle *MackeyGlass* et prédiction d'une concentration de sortie en CO₂ d'un four à gaz.

En ce qui concerne la fonction Suivi-Détection, nous avons élaboré un modèle qui permet le suivi des évolutions des paramètres qualitatifs et quantitatifs et la détection des symptômes de défaillances. Nous avons intégré des Blocs de Suivi et de Détection Qualité au modèle de [COM 91].

Nous avons terminé ce chapitre par la mise en interaction de ces deux blocs en faisant expliquer le principe de synchronisation et d'échanges informationnels entre le RR²FR et le modèle de Suivi-Détection multicritères (Qualité-flux).

Le chapitre suivant sera l'objet d'une application de notre architecture Surveillance-Commande à une industrie alimentaire.

Chapitre 5
Validation de notre approche sur un
SFPM

Chapitre 5

Validation de notre approche sur un SFPM

Table des matières du chapitre 5

5.1-	Introduction	148
5.2-	Présentation de la chaîne de production	148
5.2.1-	Présentation de la chaîne de production de double concentré de tomate.....	148
5.2.2-	Contrôle qualité de double concentré de tomate	150
5.2.3-	Paramètres temporels de la chaîne de production de double concentré de tomate	153
5.3-	Application de l'architecture « Surveillance Indirecte suite à une Commande Robuste » à la chaîne automatisée de production de DCT	156
5.3.1-	Modèle commande : commande robuste.....	156
5.3.1.1-	Loi de commande robuste activement bi-critère : qualité-flux	156
5.3.1.2-	Loi de commande robuste hybride bi-critère : qualité-flux	157
5.3.2-	Modèle Surveillance.....	162
5.3.2.1-	Fonction pronostic : prédiction de la qualité de produit DCT par RR ² FR	163
5.3.2.1.1-	Modèle RR ² FR pour la prédiction de Brix de produit DCT.....	164
5.3.2.1.2-	Base de données	165
5.3.2.1.3-	Simulation.....	166
5.3.2.1.4-	Interprétations	169
5.3.2.2-	Fonction suivi-détection	169
5.3.3-	Synchronisation entre les différents éléments de l'architecture	174
5.4-	Conclusion.....	174

5.1- Introduction

Ce chapitre présente un exemple d'application de notre architecture « Surveillance Indirecte suite à une Commande Robuste » à un système industriel. L'exemple industriel choisi est une chaîne de production des produits alimentaires de la société SIAK-MAHBOUBA (Société des Industries Alimentaires KORBA). L'activité principale de cette société consiste à la fabrication de conserves alimentaires de Double Concentré de Tomate (DCT) à base de tomates fraîches et de Concentré d'Harissa (CH) à base de piments. Quatre raisons ont justifié ce choix :

- la nature implicite des industries alimentaires qui subissent à des contraintes temporelles (de type flux) et non-temporelles (de types qualité) ; la qualité d'un produit alimentaire est un facteur très important qui ne peut être jamais négligé,
- le comportement d'une industrie alimentaire saisonnière qui est sensible aux interruptions de fonctionnement du procédé,
- le processus de production est de type flow-shop,
- la chaîne automatisée de production installée dans l'usine de cette société.

En effet, nous voyons que notre architecture à caractère prévisionnel est très performante pour ce type d'industrie. Le processus de production de produit « Double Concentré de Tomate » sera le sujet de notre application.

Pour situer le lecteur, nous allons commencer ce chapitre, par une analyse structurelle et fonctionnelle de l'atelier considéré en vu de présenter la société dans son environnement productif. Les caractéristiques physiques du système et les détails de son fonctionnement seront bien expliqués. En suite, nous allons appliquer notre architecture à la chaîne automatisée de production installée dans l'usine de la société SIAK-MAHBOUBA.

5.2- Présentation de la chaîne de production

Créé en 1986, SIAK-MAHBOUBA est une société de type S.A.R.L. 60% de la production de double concentré de tomate est destinée pour l'exportation vers la Lybie, la France, l'Italie,...

En ce qui concerne le côté technique, toutes les machines du processus de production sont d'origine Italienne. Ceci est dû au fait que l'Italie est l'un des pays les plus spécialisés en matière de fabrication et de montage des machines et des chaînes de production de conserverie. Manzini, Mingazzini et Ghizzone sont les fournisseurs du matériel et des pièces de rechange.

5.2.1- Présentation de la chaîne de production de double concentré de tomate

Afin d'obtenir un produit de type double concentré de tomate, dès la réception de la matière première, tout un processus de transformation et de production doit être respecté. En effet, nous distinguons les étapes élémentaires (*Ee*) de production suivantes :

- *réception* : les tomates sont soumises à un contrôle qualité. Seuls les produits appropriés aux normes en vigueur sont admis ;
- *prélavage et triage* : c'est une chaîne comportant un élévateur à godets et une cuve. Les tomates acceptées sont déchargées et lavées avec de l'eau propre à haut débit afin

d'enlever les restes de terre, les résidus antiparasitaires, les boues et les petites feuilles, la charge microbienne et l'entassement des tomates. Le triage est manuel. Les ouvriers enlèvent les tomates vertes ainsi que celles pourries et les tiges de tomates. La partie initiale du transporteur de la table de tirage plonge dans la cuve de lavage et se termine dans le broyeur ;

- *broyage* : le broyage est un traitement préliminaire à l'extraction de jus de tomates. la machine utilisée pour assurer cette fonction est un ensemble broyeur avec cuve et pompe volumétrique. Le processus de production de cette activité consiste à décharger les tomates de la table de triage dans le broyeur pour être broyées. Les tomates ainsi broyées passent à travers une passoire pour être récupérées dans une cuve. A partir de cet instant on parle d'un jus de tomates. Ce dernier s'envoie grâce à une volumétrie vers le préchauffeur;
- *préchauffage* : le jus de tomates sortant du broyeur est aspiré par la pompe volumétrique pour s'écouler dans des tuyaux chauffés extérieurement chauffés par la vapeur circulent à contre courant. Cette activité permet : la réduction de la charge microbienne superficielle, l'inactivation des pectinases afin d'obtenir un concentré de consistance meilleure, l'empêchement de la carbonisation et de l'oxydation du produit, etc. La température à la quelle s'effectue cette opération est entre 70 et 85 °C ;
- *extraction et raffinage de jus* : Le jus de tomates passe à travers le passoir où une destruction de l'épiderme du fruit aura lieu sous l'action d'une force centrifuge, les déchets sont extraits vers l'extérieur. Le jus obtenu est retenu dans une citerne à une température voisine de 80°C avec un le pourcentage en matière sèche ou Brix de 5 à 7% de concentration ;
- *concentration de jus* : Le jus passe dans un ensemble d'évaporateur à triple effets pour l'extraction d'eau et l'obtention d'un concentré de tomate de l'ordre de 28% minimum. l'ensemble d'évaporateur consiste en trois évaporateurs à faisceaux tubulaires verticaux chacun ayant sa propre cyclone. Cette concentration ou évaporation se fait sous vide à une température de 90 à 100 °C ;
- *pasteurisation* : la pasteurisation est un traitement thermique qui consiste à faire chauffer le jus obtenu à la sortie de l'évaporateur à une température de ~90 °C pendant un certain temps bien précis. le but de cette opération est de détruire la plus part des micro-organismes et essentiellement les bactéries pathogènes non sporulées ;
- *dosage* : après la pasteurisation le produit est dosé à l'aide d'une doseuse volumétrique à piston dans des boites métallique par la suite ces boites passent dans une sertisseuse ;
- *sertissage* : la boite pleine arrive à sensibiliser un capteur. A cet instant, le système de commande permet la pose du couvercle tout en lui faisant sertir à la boite. La date de fabrication s'écrit, au même temps, sur le fond de la boite grâce à un système d'impression sur le fond de la boite ;
- *stérilisation* : les boîtes serties arrivent à un bassin de pasteurisation à une température de ~95°C, ce traitement thermique permet de préserver le produit de toutes bactéries sans altérer sa valeur nutritive. A la sortie du bassin de pasteurisation, les boites passent brusquement dans un autre grand bassin d'eau froid. Le choc thermique permet la destruction des bactéries ;

- *conditionnement* : l’emballage des boîtes se fait par une cartonneuse. Un marquage du numéro de lot qui précise la date et l’heure de fabrication (pour assurer la traçabilité) est effectué.

Ces étapes élémentaires peuvent être classées selon leurs fonctions en quatre grandes étapes : traitement de tomates fraîches, pré-cuisson, cuisson et emballage (Fig. 5.1).

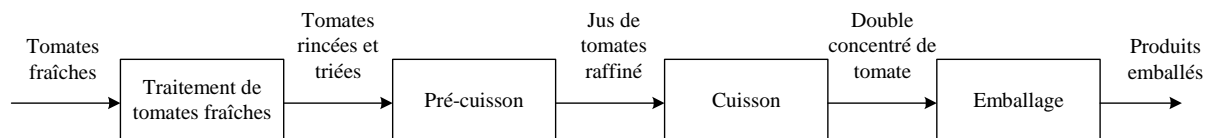


Fig. 5.1- Etapes de production

L’étape de traitement de tomates fraîches est composée de deux étapes élémentaires : réception ainsi que pré-lavage et triage. L’étape de pré-cuisson comprend le broyage le préchauffage et enfin l’étape élémentaire extraction et raffinage de jus. L’étape de cuisson est constituée de deux étapes élémentaires : concentration de jus et pasteurisation. L’étape d’emballage consiste à passer par les quatre étapes élémentaires suivantes : dosage, sertissage, stérilisation et conditionnement.

5.2.2- Contrôle qualité de double concentré de tomate

Etapes	Etapes élémentaires	Paramètres à contrôler	En ligne	laboratoire	Fréquence	
Traitement de tomates fraîches	Réception	Brix (brix)		X	A chaque réception	
	Pré-lavage et triage	Contrôle visuelle de tomates fraîches	X		En continue	
		Chloration de l'eau (ppm)	X		En continue	
Pré-cuisson	Broyage	Brix (brix)		X	30 mn	
	Préchauffage	Température (°C)	X		En continue	
	Extraction et raffinage	Brix (brix)	X		2 heures	
Cuisson	Concentration de jus	Température (°C)	X		En continue	
		Brix		X	30 mn	
	Pasteurisation	Température (°C)	X		2 heures	
Emballage	Dosage	Température (°C)	X		En continue	
	Sertissage					
	Stérilisation	Température (chauffage) (°C)	X			2 heures
		Température (refroidissement) (°C)	X			1 heure
		Chloration de l'eau de refroidissement (ppm)	X			En continue
		Poids net (grammes)			X	1 heure
		Brix (brix)			X	30 mn
		PH (unités)			X	30 mn
		Acidité (%)			X	30 mn
Conditionnement						

Tab. 5.1- Paramètres qualité contrôlés durant la production de DCT

Le contrôle qualité de double concentré de tomate est effectué, selon la nature du paramètre contrôlé, soit en ligne soit par échantillonnage, suite aux étapes élémentaires au laboratoire.

En effet, SIAK-MAHBOUBA dispose d'un laboratoire de contrôle qui a pour objectif d'aboutir à un produit qui répond aux exigences du marché en termes de qualité et de flux. D'autre part, les lignes de production sont bien équipées par les instruments de mesure en temps réel. Le tableau 5.1 illustre bien les différents paramètres contrôlés en ligne et au laboratoire. Le Réseau de Petri à Intervalles suivant illustre les marges de robustesse active de qualité de DCT définies par les responsables de SIAK-MAHBOUBA.

Note : On considère que les grandeurs physique influant la qualité du produit des paramètres qualité produit.

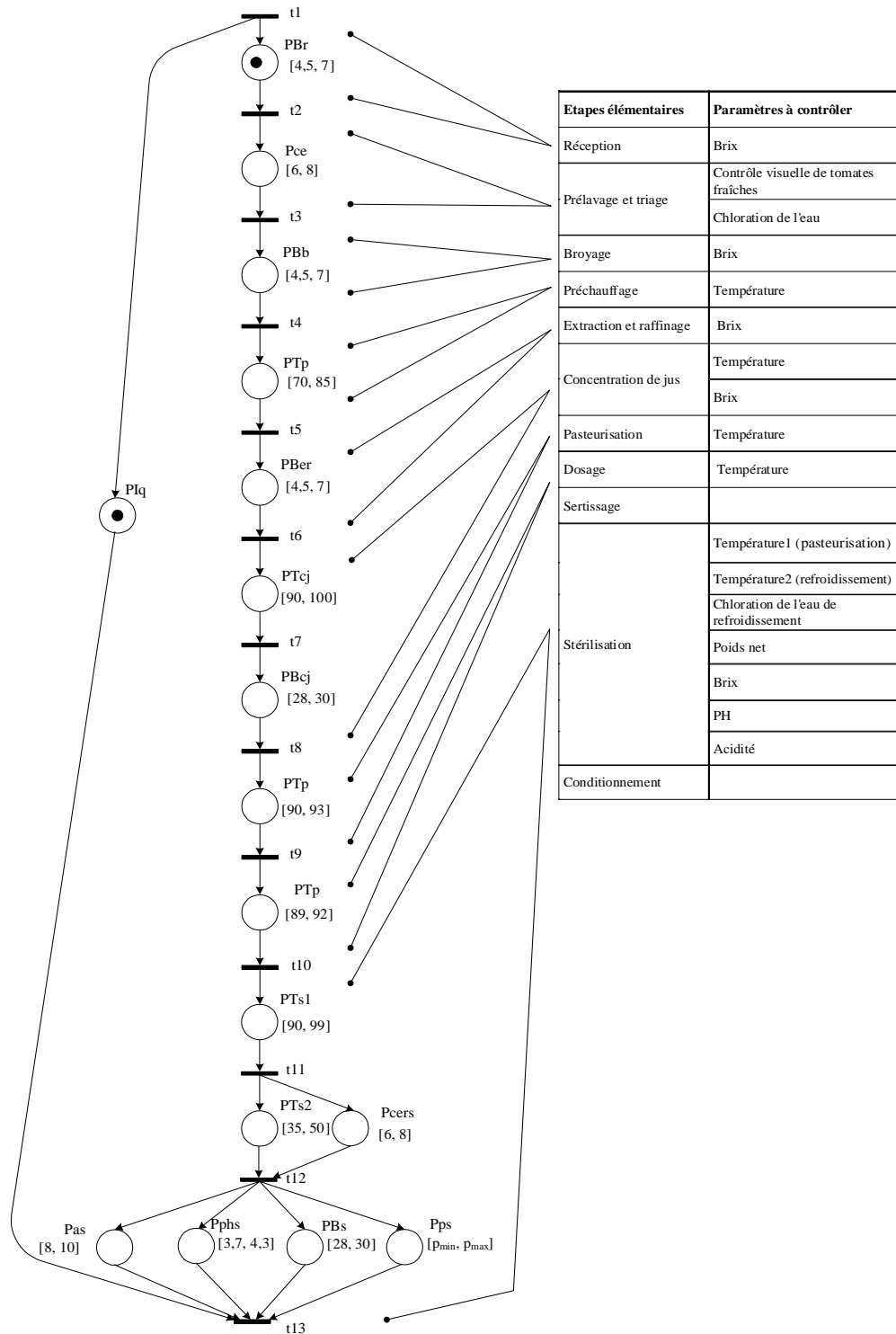


Fig. 5.2- RdP à Intervalles modélisant les marges de robustesse active de la qualité de DCT

Avec :

PBr : Place modélisant la robustesse active du paramètre qualité *Brix* suite à l'étape élémentaire *réception*,

Pce : Place modélisant la robustesse active du paramètre qualité *chloration de l'eau* durant l'étape élémentaire *prélavage et triage*,

PBb : Place modélisant la robustesse active du paramètre qualité *Brix* suite à l'étape élémentaire *Broyage*,

PTpc : Place modélisant la robustesse active du paramètre de contrôle *température* durant l'étape élémentaire *préchauffage*,

PBer : Place modélisant la robustesse active du paramètre qualité *Brix* suite à l'étape élémentaire *extraction et raffinage*,

PTcj : Place modélisant la robustesse active du paramètre de contrôle *température* durant l'étape élémentaire *concentration de jus*,

PBcj : Place modélisant la robustesse active du paramètre qualité *Brix* suite à l'étape élémentaire *concentration de jus*,

PTp : Place modélisant la robustesse active du paramètre de contrôle *température* durant l'étape élémentaire *pasteurisation*,

PTs1 : Place modélisant la robustesse active du paramètre de contrôle *température de pasteurisation* durant la sous-étape élémentaire *pasteurisation* de l'étape élémentaire *stérilisation*,

PTs2 : Place modélisant la robustesse active du paramètre de contrôle *température de refroidissement* durant la sous-étape élémentaire *refroidissement* de l'étape élémentaire *stérilisation*,

Pcers : Place modélisant la robustesse active du paramètre qualité *chloration de l'eau de refroidissement* durant la sous-étape élémentaire *refroidissement* de l'étape élémentaire *stérilisation*,

Pas : Place modélisant la robustesse active du paramètre qualité *acidité* suite à l'étape élémentaire *stérilisation*,

Pphs : Place modélisant la robustesse active du paramètre qualité *PH* suite à l'étape élémentaire *stérilisation*,

PBs : Place modélisant la robustesse active du paramètre qualité *Brix* suite à l'étape élémentaire *stérilisation*,

PPs : Place modélisant la robustesse active du paramètre qualité *poids* suite à l'étape élémentaire *stérilisation*,

PIq : place modélisant la robustesse active de l'indicateur globale de qualité de DCT.

Les intervalles de robustesse active (marge de conformité qualité) attribués aux paramètres qualité par les responsables sont indiqués sur la figure. La figure 5.2 présente un RdP à Intervalles modélisant les différentes étapes et sous étapes de contrôle qualité de produit (ou matière première ou semi-produit) DCT, ceci tout en intégrant les marges de robustesse active

des paramètres qualité. Nous remarquons, sur la figure, des étapes élémentaires qui se décomposent en deux ou plus paramètres qualité. Dans ce cas, on fait avancer les places modélisant les paramètres contrôlés en ligne dans la structure de modélisation de l'étape élémentaire. Les paramètres qualité contrôlés au laboratoire se modélisent en avale. Les paramètres qualité d'une même étape élémentaire contrôlés au laboratoire se modélisent en parallèle ; exemple : acidité, poids net, PH et Brix de l'étape stérilisation. Les paramètres qualité d'une même étape élémentaire contrôlés en ligne se modélisent en série; exemple : température de refroidissement et chloration de l'eau de refroidissement de la sous étape refroidissement de l'étape stérilisation.

5.2.3- Paramètres temporels de la chaîne de production de double concentré de tomate

Le contrôle de flux de matière première, produit semi-fini et produit fini est une opération nécessaire pour maîtriser le cycle de production qui cherche toujours à satisfaire les besoins de marché. En effet, la société SIAK-MAHBOUBA a mis des marges de conformité afin de créer une certaine robustesse vis-à-vis la qualité de produit et les besoins de marché. Le tableau suivant résume les paramètres flux ou temporels relatifs à la chaîne de production de produit DCT.

Etapes	Etapes élémentaires	Processeur ou équipement	Paramètres	Fréquences	
Traitement de tomates fraîches	Réception	bascule	Temps de pesage	À l'arrivé de chaque camion	
	Prélavage et triage	Prélavage	2 Tapis roulant	Vitesse (mètre/minute)	En continue
		Triage	3 Tapis roulant	Vitesse (mètre/minute)	En continue
Pré-cuisson	Broyage	Ensemble broyeur avec cuve et pompe volumétrique.	Vitesse (tr/min)	En continue	
			Débit à la sortie (m ³ /h)	En continue	
	Préchauffage	Tuyaux chauffés extérieurement chauffés par la vapeur circulent à contre courant	Vitesse (mètre/minute)	En continue	
	Extraction et raffinage	Passoir, raffineuse	Débit à la sortie (m ³ /h)	En continue	
Cuisson	Concentration de jus	Evaporateur	Temps de séjour à l'évaporateur (seconde)	En continue	
	Pasteurisation	Pasteurisateur	Temps de séjour au pasteurisateur1 (seconde)	En continue	
Emballage	Dosage	Doseuse	Débit (m ³ /h)	En continue	
	Sertissage	Sertisseuse	vitesse de sertissage (boites /mn)	En continue	
	Stérilisation	Bassin de pasteurisation		Temps de séjour au bassin de pasteurisation (seconde)	- 15 mn pour boîte (1/2)
		Bassin de (refroidissement)		Temps de séjour au bassin de refroidissement (seconde)	- 20 mn pour boîte (4/4) - 35mn pour boîte (5/1)
	Conditionnement	Cartonneuse		Temps de cycle (seconde)	

Tab. 5.2- Paramètres flux de la chaîne de production de DCT

Le Réseau de Petri à Intervalles suivant illustre les marges de robustesse active de flux (ou temporel) de DCT définies par les responsables de SIAK-MAHBOUBA.

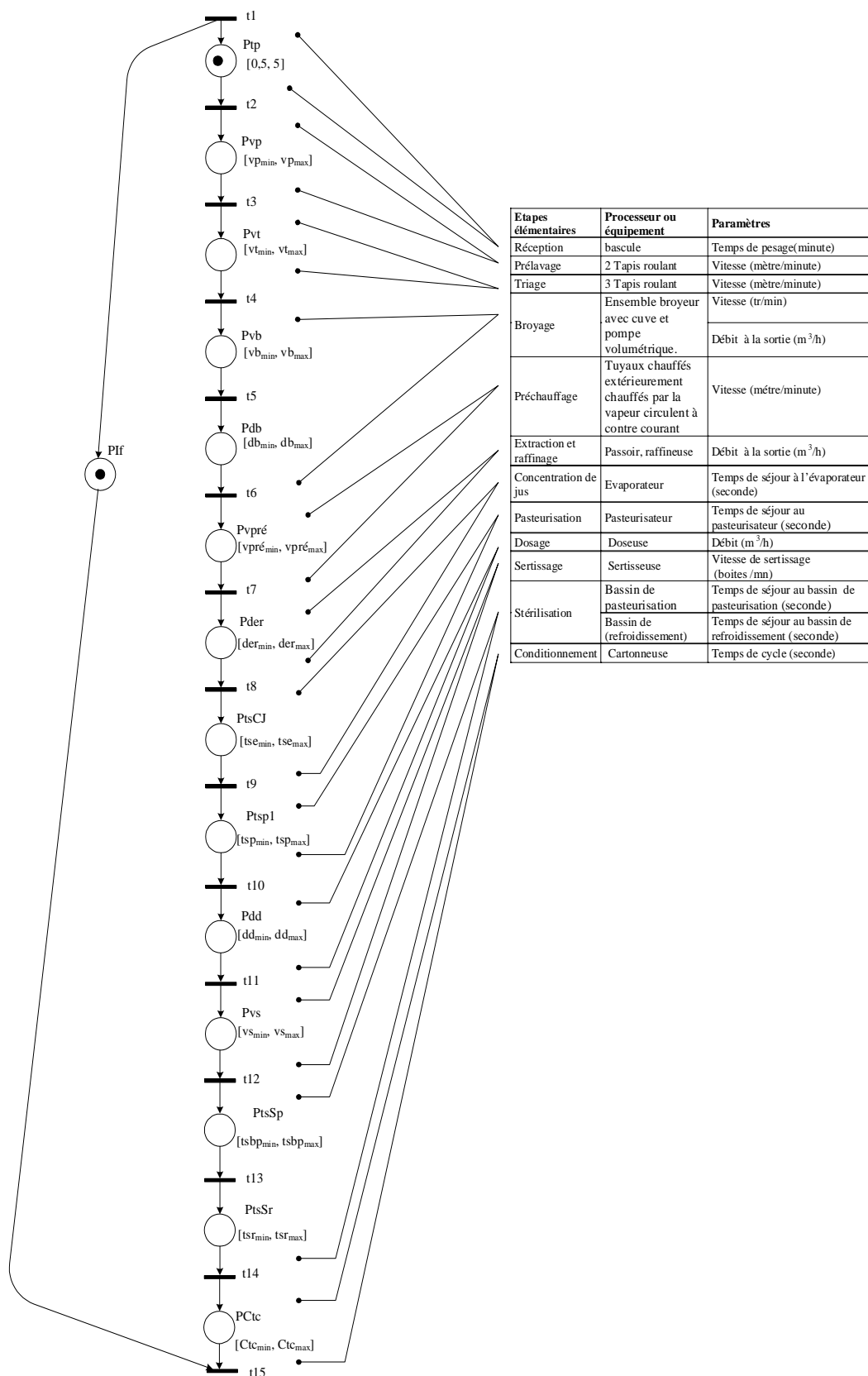


Fig. 5.3- RdP à Intervalles modélisant les marges de robustesse active des paramètres flux de la chaîne de production de DCT

Avec :

Ptp : Place modélisant la robustesse active de *temps de pesage* lors de l'étape élémentaire *réception*,

Pvp : Place modélisant la robustesse active de la *vitesse des tapis* roulants lors de l'étape élémentaire *prélavage*,

Pvt : Place modélisant la robustesse active de la *vitesse des tapis* roulants lors de l'étape élémentaire *triage*,

Pvb : Place modélisant la robustesse active de la *vitesse du broyeur* lors de l'étape élémentaire *broyage*,

Pdb : Place modélisant la robustesse active du *débit à la sortie* du broyeur lors de l'étape élémentaire *broyage*,

Pvpré : Place modélisant la robustesse active de la *vitesse de circulation* de tomates broyées dans les tuyaux chauffés à la sortie du broyeur lors de l'étape élémentaire *préchauffage*,

Pder : Place modélisant la robustesse active du *débit à la sortie* de la raffineuse lors de l'étape élémentaire *extraction et raffinage*,

PtsCJ : Place modélisant la robustesse active du *temps de séjour à l'évaporateur* lors de l'étape élémentaire *concentration de jus*,

Ptsp1 : Place modélisant la robustesse active du *temps de séjour au pasteurisateur (seconde)* lors de l'étape élémentaire *pasteurisation*,

Pdd : Place modélisant la robustesse active du *débit de doseuse* lors de l'étape élémentaire *dosage*,

Pvs : Place modélisant la robustesse active de la *vitesse de sertissage* lors de l'étape élémentaire *sertissage*,

PtsSp : Place modélisant la robustesse active du *temps de séjour au bassin de pasteurisation* lors de l'étape élémentaire *stérilisation*,

PtsSr : Place modélisant la robustesse active du *temps de séjour au bassin de refroidissement* lors de l'étape élémentaire *stérilisation*,

Ptcs : Place modélisant la robustesse active du *temps de cycle* déduit à l'étape élémentaire *conditionnement*,

PIq : place modélisant la robustesse active de l'indicateur globale de flux (ou temporel) de DCT.

Les intervalles de robustesse active attribués aux paramètres flux par les responsables sont indiqués sur la figure.

La figure 5.3 illustre un RdP à Intervalles modélisant les différentes contraintes flux (marges de robustesse active lors des différentes étapes élémentaire de production du produit (ou matière première ou semi-produit) DCT.

Dans la suite de ce chapitre, nous allons exploiter ces marges de robustesse mises sur les paramètres flux et qualité pour implanter notre architecture tout en commençant par la définition d'une loi de commande hybride robuste bi-critère (flux et qualité).

5.3- Application de l'architecture « Surveillance Indirecte suite à une Commande Robuste » à la chaîne automatisée de production de DCT

5.3.1- Modèle commande : commande robuste

5.3.1.1- Loi de commande robuste activement bi-critère : qualité-flux

Nous allons commencer ce paragraphe par créer et présenter la loi de commande robuste qui met en interaction les contraintes de robustesse active mises sur les paramètres qualité et flux de la chaîne automatisée de production de DCT.

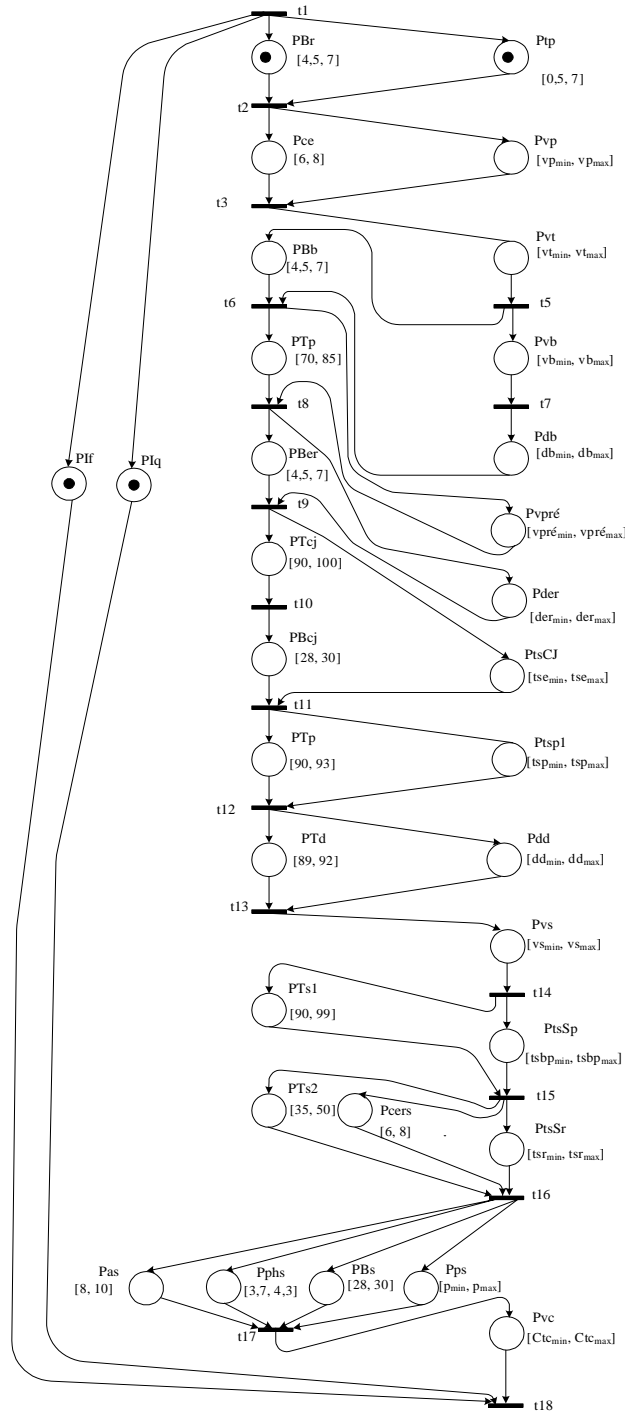


Fig. 5.4- RdP à Intervalles modélisant la robustesse active des paramètres flux et de qualité

Le RdP à Intervalles de la figure 5.4 forme une loi de commande robuste multicritères qui traite seulement les marges de robustesse active des paramètres flux et qualité. En effet, on peut constater que la construction d'une telle loi de commande ne nécessite que les cinq étapes suivantes soient appliquées:

- détermination des paramètres flux de système ainsi que leurs marges de robustesse active,
- modélisation des marges de robustesse active des paramètres flux par un RdP à Intervalles,
- détermination des paramètres flux de système ainsi que leurs marges de robustesse active,
- modélisation des marges de robustesse active des paramètres qualité par un RdP à Intervalles,
- assemblage de deux de deux modèle RdP à Intervalles (flux et qualité) tout en faisant synchroniser entre les étapes élémentaires.

Appliquées à notre système industriel : chaîne automatisée de production de DCT, les quatre premières étapes sont bien présentées dans la section précédente à travers, respectivement, le tableau Tab. 5.1, la figure Fig. 5.2, le tableau Tab. 5.2 et la figure Fig. 5.3. La cinquième étape est importante, puisqu'elle nous permet de passer d'une modélisation de loi de commande robuste monocritère traitant un seul type de robustesse (active) à une modélisation de loi de commande robuste bi-critère. Cette étape consiste à mettre en interaction les places modélisant la robustesse des paramètres flux et la places modélisant la robustesse des paramètres qualité. Donc pour chaque étape élémentaire Ee , nous faisons assembler la tranche de RdP à Intervalles modélisant les paramètres flux de Ee avec la tranche de RdP à Intervalles modélisant les paramètres qualité de Ee . A partir de la transition d'entrée de Ee , deux arcs s'orientent vers, respectivement, la place modélisant le premier paramètre flux et celle modélisant le premier paramètre qualité de Ee . A la sortie de Ee , deux arcs lient respectivement la place modélisant le dernier paramètre flux et celle modélisant le dernier paramètre qualité à la transition de sortie (Fig. 5.4).

5.3.1.2- Loi de commande robuste hybride bi-critère : qualité-flux

Après avoir présenté, dans le paragraphe précédent, la loi de commande robuste activement bi-critère de la chaîne automatisée de production de DCT. Le présent paragraphe est consacré pour la définition d'un modèle de commande robuste hybride bi-critère : qualité-flux.

En effet, la spécification et la construction d'une commande robuste pour un système de production industriel consiste à bien choisir les paramètres de la loi de commande. Ce choix doit répondre aux objectifs technique et économique de l'entreprise. En effet, en collaboration avec les responsables de SIAK-MAHBOUBA, la loi de commande hybride bi-critère qu'on va implanter à la chaîne automatisée de production de DCT, ne doit tenir compte que des paramètres significatifs de flux et de qualité. Ceci afin de choisir les paramètres flux et

qualité permettant d'analyser l'impact de la qualité sur les éléments du procédé de fabrication tout en optimisant la coté économique de l'implantation de notre architecture de Surveillance-Commande « Surveillance indirecte suite à une commande robuste ».

Pour choisir les paramètres significatifs de flux et de qualité, nous définissons les étapes suivantes :

- *analyse fonctionnelle de système de production* : recherche des paramètres du système qui s'interviennent lors de l'évolution de la production et qui influent le flux et la qualité de produit,
- *identification des paramètres significatifs* : recensement des paramètres qui influent les indicateurs globaux du flux et de la qualité de produit,
- *choix des paramètres significatifs pour la loi de commande* : choix et définition des marges de robustesse passive et active des paramètres qu'on va utiliser pour construire la loi de commande.

En collaboration avec les responsables de SIAK-MAHBOUBA, ces étapes de détermination des paramètres significatifs nous amènent à élaborer les tableaux 5.3 et 5.4.

Etapes	Etapes élémentaires	Paramètres significatifs de qualité	Marges de robustesse (qualité)		
			active	passive	
Traitement de tomates fraîches	Réception				
	Prélavage et triage				
Pré-cuisson	Broyage	Brix (brix)	[4,5, 5]	[5, 7]	
	Préchauffage	Température (°C)	[70, 78] et [83, 85]	[78, 83]	
	Extraction et raffinage				
Cuisson	Concentration de jus	Brix (brix)	[28, 28,5]	[28,5, 30]	
	Pasteurisation	Température (°C)	[90, 91] et [92, 93]	[91, 92]	
Emballage	Dosage				
	Sertissage				
	Stérilisation	Température (chauffage) (°C)		[90, 94] et [97, 99]	[94, 97]
		Température (refroidissement) (°C)		[40, 50]	[35, 40]
		Poids net (grammes)		$[pa_{1min}, pp_{min}]$ et $[pp_{max}, pa_{2max}]$	$[pp_{min}, pp_{max}]$
		Brix (brix)		[28, 28,5]	[28,5, 30]
		PH (unités)		[3,5, 3,8] et [4,2, 4,5]	[3,8, 4,2]
		Acidité (%)		[9, 10]	[0, 9]
Conditionnement					

Tab. 5.3- Marges de robustesse des paramètres significatifs de qualité

Les marges de robustesse du poids net ne sont pas présentées en valeurs. En effet, la valeur de ce paramètre varie selon le type d'emballage. Les valeurs limites de marges de robustesse active sont : +/- 3% du poids net indiqué sur la boîte d'emballage. De même, les valeurs limites de marges de robustesse passive sont : +/- 1,5% du poids net indiqué sur la boîte d'emballage. Prenant par exemple le cas d'une boîte de type 4/4, le poids net indiqué sur la boîte d'emballage est 800g, ceci signifie les marges de robustesse suivantes :

- active : $[pa_{1min}, pp_{min}] = [776, 788]$ et $[pp_{max}, pa_{2max}] = [812, 824]$,
- passive : $[788, 812]$.

Etapas	Etapas élémentaires	Paramètres significatifs de flux	Marges de robustesse (flux)	
			active	passive
Traitement de tomates fraîches	Réception			
	Prélavage et Triage			
Pré-cuisson	Broyage			
	Préchauffage			
	Extraction et raffinage			
Cuisson	Concentration de jus	Temps de séjour à l'évaporateur (seconde)	$[tsep_{max}, tsea_{2max}]$	$[tsep_{min}, tsep_{max}]$
	Pasteurisation	Temps de séjour au pasteurisateur 1 (seconde)	$[tspp_{max}, tsppa_{2max}]$	$[tspp_{min}, tspp_{max}]$
Emballage	Dosage			
	Sertissage			
	Stérilisation	Temps de séjour au bassin de pasteurisation (seconde)	$[tsbpp_{1max}, tsbpa_{2max}]$	$[tsbpp_{min}, tsbpp_{max}]$
		Temps de séjour au bassin de refroidissement (seconde)	$[tsrp_{max}, tsra_{2max}]$	$[tsrp_{min}, tsrp_{max}]$
	Conditionnement	Temps de cycle (seconde)	$[Ctcp_{max}, Ctca_{2max}]$	$[Ctcp_{min}, Ctcp_{max}]$

Tab. 5.4- Marges de robustesse des paramètres significatifs de flux

Les quatre premiers paramètres temporels choisis (temps de séjour à l'évaporateur, temps de séjour au pasteurisateur 1, temps de séjour au bassin de pasteurisation et temps de séjour au bassin de refroidissement) sont en effet des paramètres de temps de séjour qui influent implicitement la qualité de produit. Le paramètre vitesse de la cartonneuse traduit implicitement le temps de cycle.

A partir de deux tableaux 5.3 et 5.4 on constate que les paramètres qui vont être intervenus dans la loi de commande robuste hybride bi-critère, appliquée à la chaîne automatisée de production de DCT, se répartissent sur six étapes élémentaires qui sont : broyage, préchauffage, concentration de jus, pasteurisation, stérilisation et conditionnement.

La figure 5.5 présente le processus de contrôle des paramètres flux et qualité choisis par ordre chronologique tout en mentionnant si les paramètres flux et qualité sont contrôlés en parallèle ou en série.

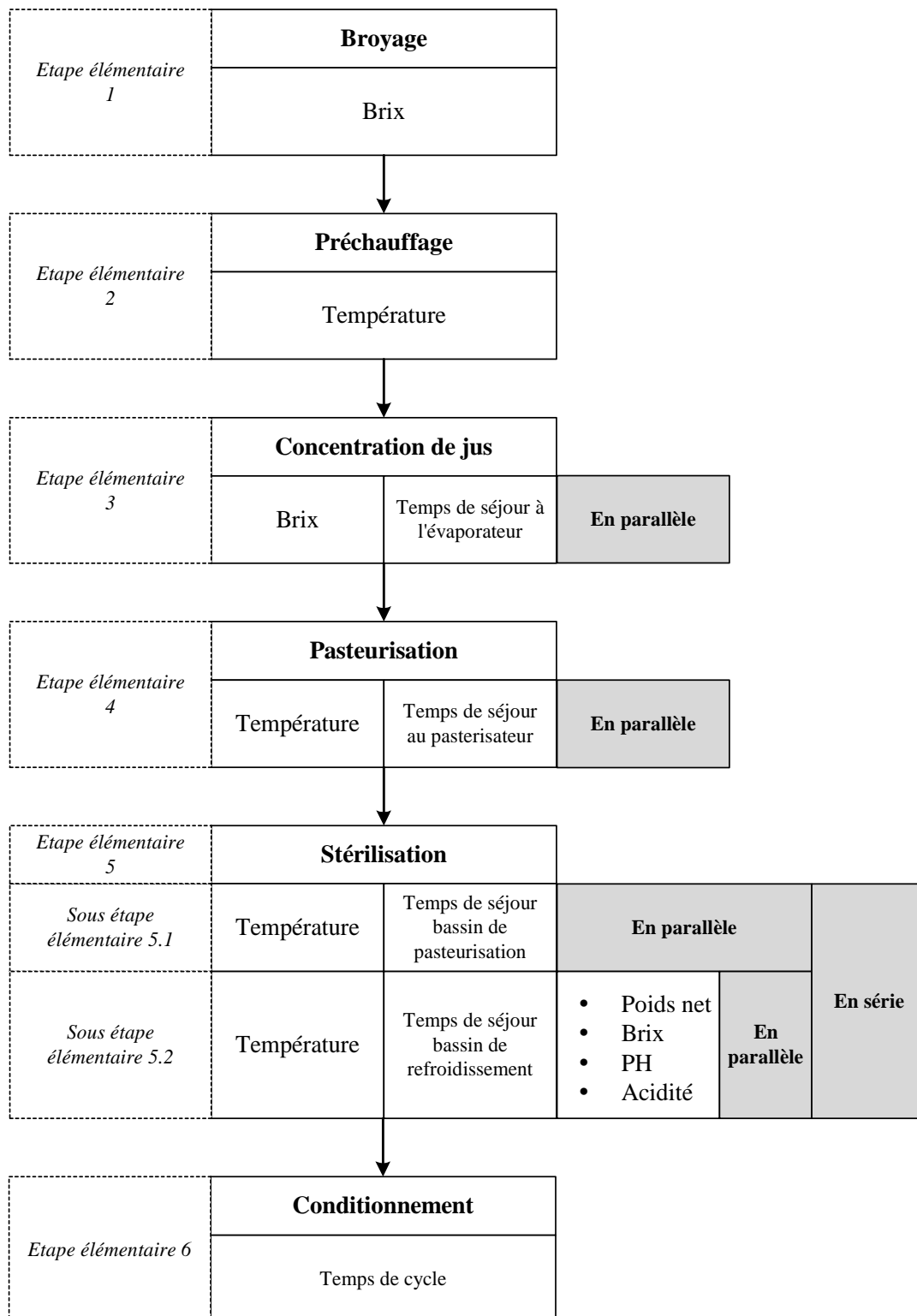


Fig. 5.5- Processus de contrôle des paramètres flux et qualité choisis par ordre chronologique

Finalement, nous présentons par la figure 5.6 la loi de commande robuste hybride multicritères appliquée à la chaîne automatisée de production de DCT.

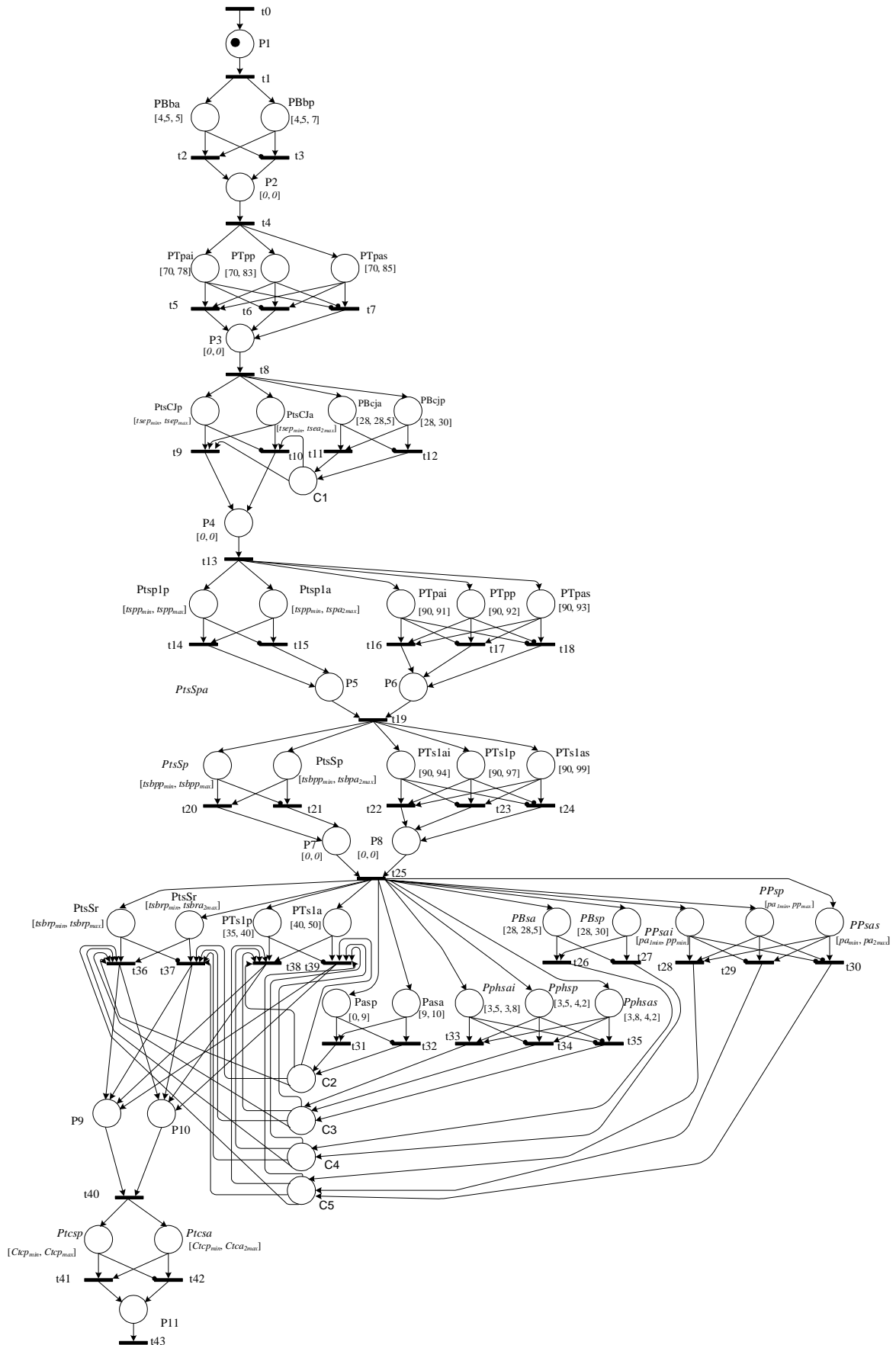


Fig. 5.6- RdP à Intervalles modélisant une commande robuste hybride multicritères appliquée à la chaîne automatisée de production de DCT

La place $P1$ modélise les deux étapes élémentaires : Prélavage et triage Réception. La place $P2$ modélise le stock intermédiaire entre les deux étapes élémentaires : Broyage et Préchauffage. La place $P3$ modélise le stock intermédiaire entre les deux étapes élémentaires : Préchauffage est Concentration de jus. La place $P4$ modélise le stock intermédiaire entre les deux étapes élémentaires : Concentration de jus et pasteurisation. Les places $P5$ et $P6$ modélisent les deux étapes élémentaires Dosage et sertissage. Les places $P7$ et $P8$ modélisent le stock intermédiaire entre les deux sous-étapes élémentaires de l'étape élémentaire Stérilisation : pasteurisation et refroidissement. Les places $P9$ et $P10$ modélisent le stock intermédiaire entre les deux étapes élémentaires : stérilisation et conditionnement. La place $P11$ modélise le stock du produit fini. Les $C1$, $C2$, $C3$, $C4$ et $C5$ permettent l'échange d'informations entre les paramètres qualité de produit d'une part et les paramètres temporels (flux) ou physique (température) d'une autre part. L'autre nomenclature est la même que celles des figures (Fig. 5.1, Fig. 5.2 et Fig. 5.3) sauf que nous ajoutons :

- ai lorsqu'on modélise la place relative à la marge de robustesse active inférieure,
- p lorsqu'on modélise la place relative à la marge de robustesse passive,
- as lorsqu'on modélise la place relative à la marge de robustesse active supérieure,
- a lorsqu'on modélise la place relative à la marge de robustesse active.

Nous attribuons l'intervalle $[0, 0]$ aux places désignées aux stocks intermédiaires vu que le produit doit passer immédiatement d'une étape élémentaire à une autre. Ceci est dû à la nature des systèmes des industries alimentaires qui sont très sensibles face aux contaminations bactériennes et aux spores.

Cette loi de commande modélisée par un RdP à Intervalles permet la synchronisation des paramètres de types différents : temporels et non temporels. En effet toutes les marges de robustesse des différents paramètres choisis sont implantées au sein de cette loi de commande. Ce RdP à Intervalles modélise une loi de commande de la chaîne automatisée de production de DCT selon l'ordre chronologique de la réalisation des activités de contrôle des paramètres qualité ou flux. Autre que son rôle d'ordonnancement, elle permet, durant le déroulement normal de production, de bien contrôler la variation et l'évolution de chaque paramètre de cette loi d'où la constatation de la nature de robustesse des étapes élémentaires et du système entière. La chaîne de production ne peut être robuste passivement qu'au cas où le parcours de commande passive est marqué : le jeton simulant le produit passe par toutes les places modélisant les marges de robustesse passive des différents paramètres significatifs flux et qualité.

Vu que les marges de robustesse active des paramètres non significatifs sont importantes quant à la maintien de la robustesse active globale du système, notre modèle de commande pour cette application comprend les deux lois de commande : robuste activement (Fig. 5.4) et hybride multicritère (Fig. 5.6).

5.3.2- Modèle Surveillance

Après développement d'une loi de commande robuste hybride multicritères désignée à la maintien de la robustesse de la chaîne automatisée de production de DCT face aux perturbations de types flux et qualité de produit, cette section est consacrée à l'implantation du bloc Surveillance. Comme il est mentionné précédemment (chapitre 1 et chapitre 4) nous ne nous intéressons, dans le cadre de notre thèse, qu'aux fonctions Pronostic et Suivi-Détection.

5.3.2.1- Fonction pronostic : prédiction de la qualité de produit DCT par RR²FR

Plusieurs paramètres qualité font partie du squelette de la loi de commande développée pour le maintien de la robustesse de deux critères flux et qualité. Néanmoins le paramètre qualité *Brix du produit fini* (après la phase de stérilisation) est un facteur très important quant à la commercialisation et l'exportation de produit DCT. En effet, vu l'importance de ce paramètre nous avons choisi de faire appliquer l'architecture RR²FR pour pronostiquer sa valeur future tout en intervenant les paramètres qui participent normalement au comportement intrinsèque du Brix de DCT. En effet, nous présentons par la figure 5.7 les paramètres qui influent le paramètre Brix du produit fini (après la phase de stérilisation).

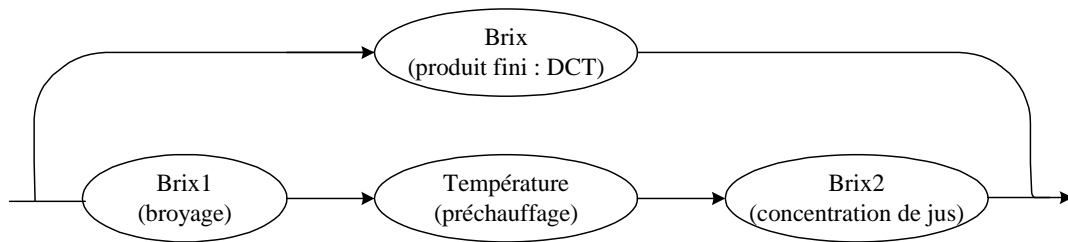


Fig. 5.7- Paramètres qui influent le paramètre Brix du produit fini DCT

Cependant, il existe des autres paramètres qui influent implicitement la valeur du Brix du produit fini comme le temps de séjour à l'évaporateur, le temps de séjour au pasteurisateur, etc. Mais, vu que la société SIAK MAHBOUBA ne possède pas, lors de la période de notre application, des appareils de mesure fiables pour quantifier ce type des paramètres, nous avons choisi de les éliminer du cycle des paramètres influant le Brix final de DCT.

La figure 5.8 présente une modélisation des marges de robustesse des paramètres à contrôler pour maintenir le Brix du produit fini.

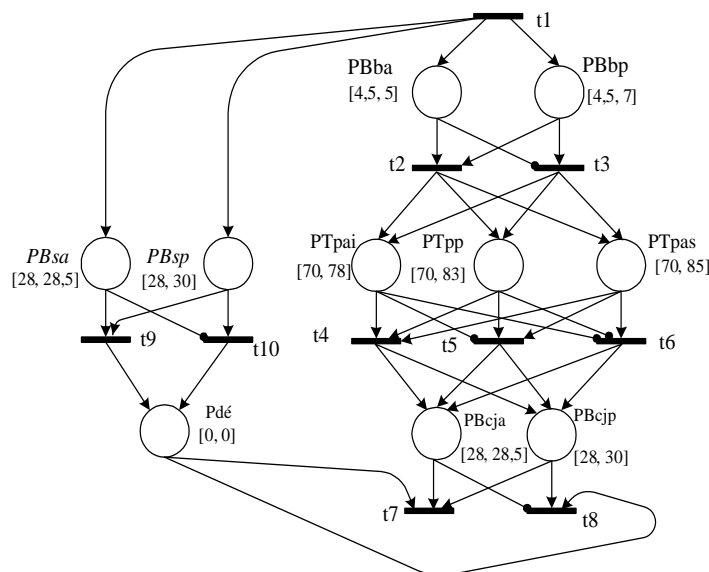


Fig. 5.8- RdP à Intervalles à « Brix valués »

La place *Pdé* est ajouté afin d'éviter les cas des blocages de RdP.

5.3.2.1.1- Modèle RR²FR pour la prédiction de Brix de produit DCT

Le RR²FR implanté pour prédire le Brix du produit final (variable expliquée) à un instant futur à partir d'un historique des mesures de tous les paramètres expliqués et explicatifs est présenté par la figure 5.9.

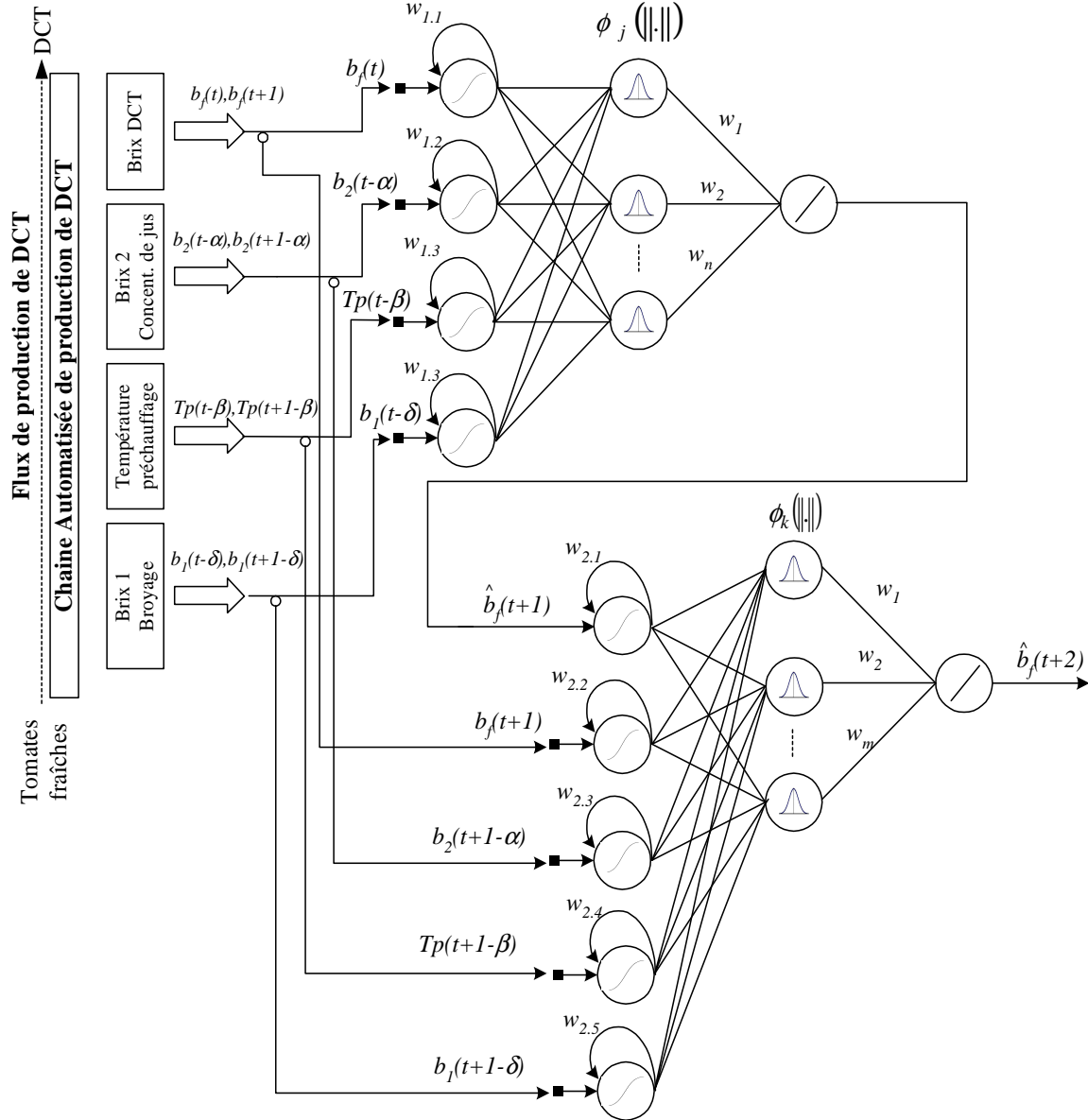


Fig. 5.9- RR²FR appliqué à la prédiction du Brix du produit fini DCT

Avec :

$b_f(t)$: Brix du produit final DCT à l'instant t ,

$b_2(t)$: Brix suite à l'étape élémentaire Concentration de jus à l'instant t ,

$b_1(t)$: Brix suite à l'étape élémentaire Broyage à l'instant t ,

$Tp(t)$: Température lors de l'étape élémentaire Préchauffage à l'instant t ,

α : décalage temporel entre la mesure de b_f et b_2 ,

β : décalage temporel entre la mesure de b_f et Tp ,

δ : décalage temporel entre la mesure de b_f et b_l .

Les décalages temporels consistent à suivre les paramètres du même lot du produit lors des différentes phases de production. Le jus de tomate broyée sera préchauffé après $\delta-\beta$ unité de temps. De même le temps de la cuisson de tomate à l'étape élémentaire concentration de jus nécessite $\beta-\alpha$ unité de temps. Enfin la durée entre la sortie de tomate e l'évaporateur (étape élémentaire concentration de jus) et la mesure finale du Brix b_f de DCT et a unité de temps.

Notre objectif est de prédire $b_f(t+2)$ du produit final DCT à partir des valeurs de $b_l(t)$, $b_l(t+1)$, $b_2(t)$, $b_2(t+1)$, $Tp(t)$, $Tp(t+1)$, $b_f(t)$, $b_f(t+1)$ et la valeur estimée de b_f à l'instant $t+1$ par le premier RRFR ($\hat{b}_f(t+1)$).

5.3.2.1.2- Base de données

En collaboration avec le personnel du service qualité de SIAK-MAHBOUBA, nous avons créé une base de données comprenant 196 valeurs pour chaque paramètre : b_l , Tp , b_2 et b_f . L'intervalle temporel entre deux mesures successives est 30 mn pour chaque paramètre. Les figures 5.10, 5.11, 5.12 et 5.13 présentent l'évolution des valeurs de la base de données des paramètres choisis.

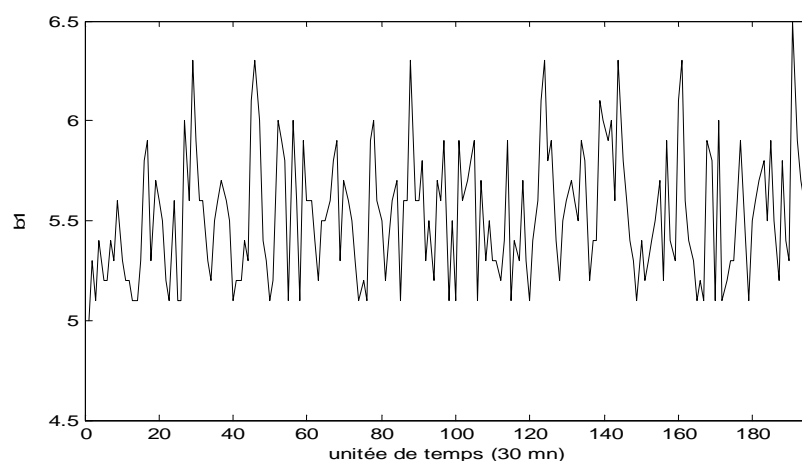


Fig. 5.10- Evolution du Brix (b_l) suite à l'étape élémentaire Broyage

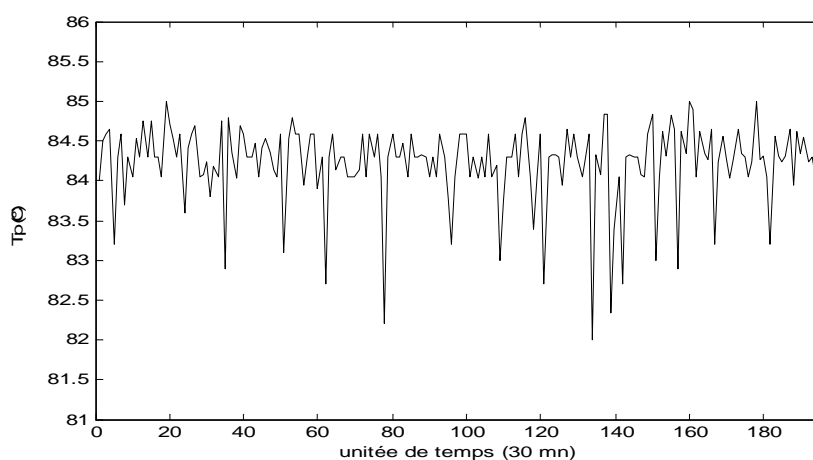


Fig. 5.11- Evolution de la température (Tp) lors de l'étape élémentaire Préchauffage

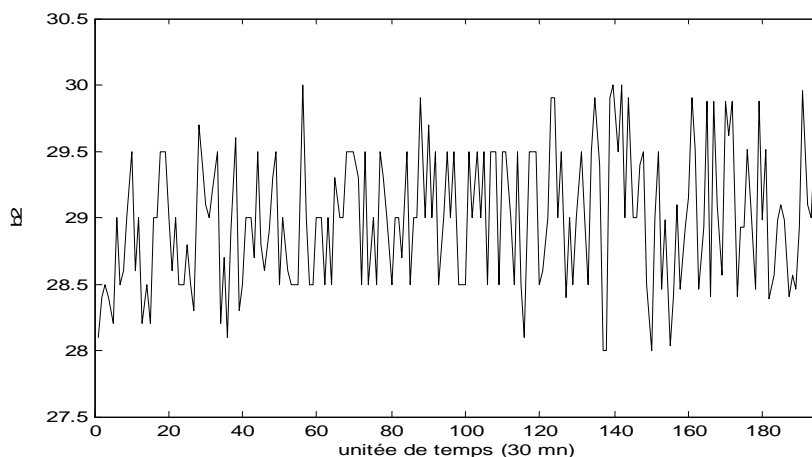


Fig. 5.12- Evolution du Brix (b_2) suite à l'étape élémentaire Concentration de jus

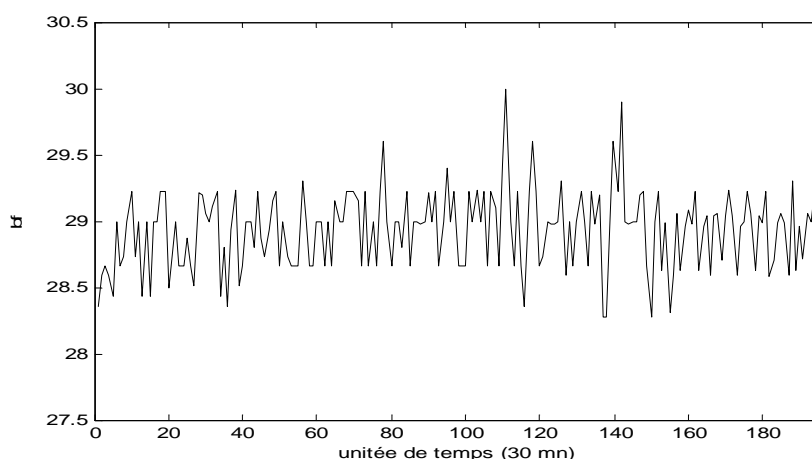


Fig. 5.13- Evolution du Brix (b_f) du produit final DCT

5.3.2.1.3- Simulation

Nous avons pris les 98 premières valeurs des paramètres de la base de données pour la phase d'apprentissage et le reste pour la phase de test.

Pour cette application les valeurs optimales des paramètres d'apprentissage sont :

- $w_{ii}=396$ et $K=0.06$ pour les neurones bouclés de la couche d'entrée de chaque RRFR,
- la variation du paramètre θ entre 0 et 1 avec un pas de 0.01 nous permet de constater la valeur optimale de θ pour initialiser l'algorithme k-moyens amélioré. La figure 5.14 illustre la variation de l'erreur de prédiction en fonction de θ . Le minimum de 1,37 % d'erreur sur les deux populations (Apprentissage + Test) se coïncide avec une valeur optimale de $\theta = 0,76$. Par la figure 5.15, nous présentons le nombre des classes en fonction de θ . La valeur qui convient à $\theta=0,76$ est $K=4$,
- le calcul des poids des connexions de la couche cachée vers la couche de sortie de chaque RRFR est assuré par la méthode MRL.

Comme pour les applications du chapitre précédant, nous avons réalisé trois simulations pour prédire $\hat{b}_f(t+2)$:

- RRFR utilisant l’algorithme K-moyens pour la phase d’apprentissage non supervisé,
- RR²FR avec le même algorithme,
- RR²FR tout en utilisant l’algorithme d’apprentissage non supervisé est K-moyens amélioré.

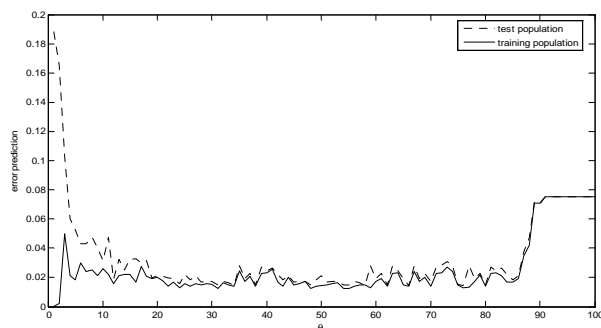


Fig. 5.14- Erreur moyenne relative de prédiction (algorithme RCE) : $f(\theta) = \text{erreur moyenne}$

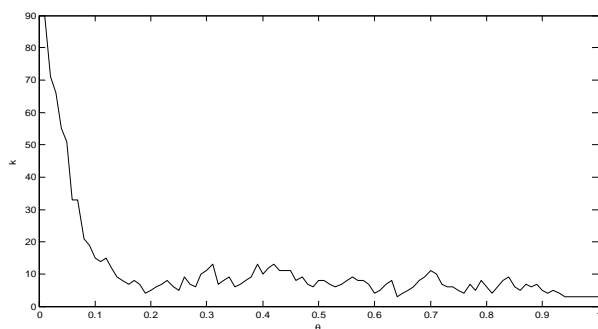


Fig. 5.15- Nombre de neurones cachés (centres) en fonction du paramètre θ

Les figures (Fig. 5.16, Fig. 5.17 et Fig. 5.18) et le tableau 5.5 illustrent les résultats des trois simulations.

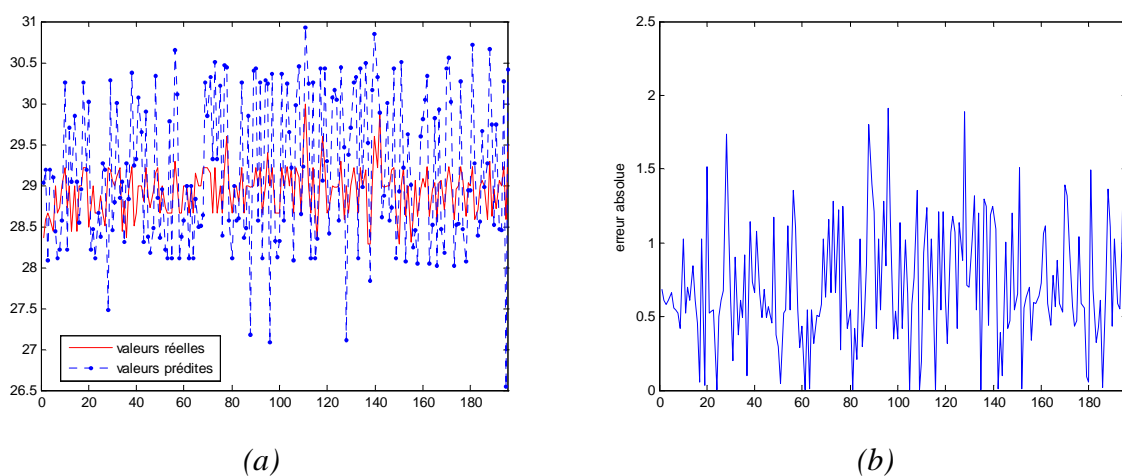
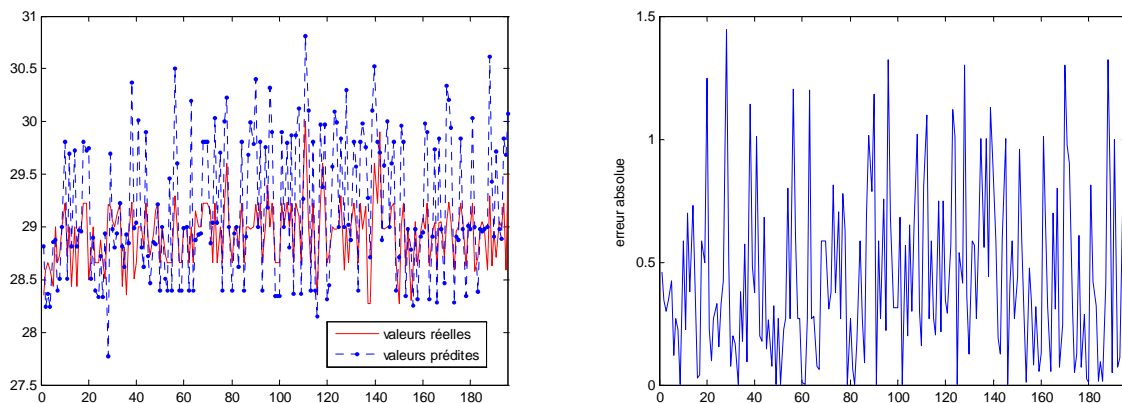


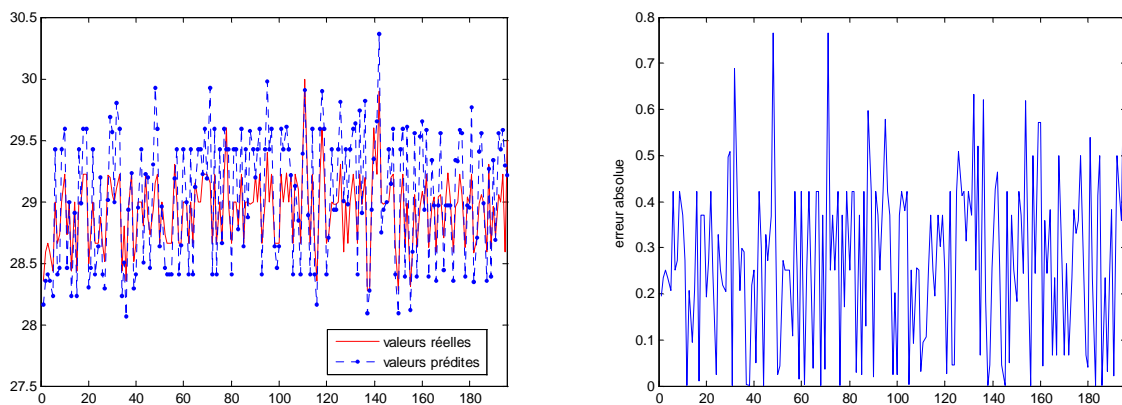
Fig. 5.16- résultat de prédiction obtenu par un RRFR + K-moyens

- (a) Résultat de la prédiction de b_f obtenu par le réseau
 (b) erreur absolue



(a) (b)
Fig. 5.17- résultat de prédiction obtenu par un RR²FR + K-moyens

(a) Résultat de la prédiction de b_f obtenu par le réseau
 (b) erreur absolue



(a) (b)
Fig. 5.18- résultat de prédiction obtenu par un RR²FR + K-moyens amélioré

(a) Résultat de la prédiction de b_f obtenu par le réseau
 (b) erreur absolue

	Moy ₁	% E ₁	Moy ₂	% E ₂	T. app. (secondes)
RRFR+ K-moyens	0.679	2.33	0.714	2.44	0.01891
RR ² FR+ K-moyens	0.407	1.41	0.445	1.51	~0.01943
RR ² FR+ K-moyens amélioré	0.269	0.92	0.273	0.94	~0.02099

Tab. 4.2- Résumé des résultats des simulations pour la prédiction de Brix b_f du produit

Avec :

Moy_1 : la moyenne des erreurs absolues de la population d'apprentissage,

% E_1 : la moyenne de pourcentages des erreurs absolues de la population d'apprentissage,

Moy_2 : la moyenne des erreurs absolues de toute la population (test+apprentissage),

% E_2 : la moyenne de pourcentages des erreurs absolues de toute la population (test+apprentissage),

T_{app} : temps d'apprentissage.

5.3.2.1.4- Interprétations

L'application de RR²RF à la prédiction de b_f a montré une autre fois l'efficacité de cette architecture en termes de minimisation de l'erreur absolue de prédiction. L'erreur absolue de prédiction est concrètement améliorée tout en passant d'une simulation à une autre : RRFR + K-moyens, RR²FR + K-moyens et RR²FR + K-moyens amélioré.

En faisant appliquer l'architecture RRFR utilisant l'algorithme K-moyens, le pourcentage de l'erreur de prédiction par rapport à l'amplitude réelle de b_f est à l'ordre de 2,44%. Ce pourcentage a été diminué tout en passant de la première simulation (RRFR + K-moyens) à la deuxième simulation (RR²FR + K-moyens) pour atteindre une valeur de 1,51%. A ce stage, c.à.d. après la première et la deuxième simulation, on peut constater la performance de cette architecture par rapport aux RRFR. En effet, on remarque une réduction à l'ordre de $((1-(0,445/0,714)) \times 100)\% \sim 37,67\%$ de l'amplitude moyenne de l'erreur absolue de prédiction. Ceci est dû au deuxième RRFR de RR²FR qui permet la prise en compte de la corrélation entre $\hat{b}_f(t+1)$ et $b_f(t+1)$ par le deuxième RRFR.

L'application de l'algorithme K-moyens amélioré lors de la troisième simulation a apporté une solution optimale non seulement en termes de résoudre les inconvénients de la version traditionnelle de cet algorithme situés dans le chapitre précédent (nombre et centres des classes et nuage de points vide), mais aussi en termes de réduction de l'erreur absolue de prédiction. En effet, cette erreur est réduite d'une valeur égale à 38,25% par rapport à l'amplitude moyenne de l'erreur absolue de prédiction de la deuxième simulation et d'une valeur égale à 61,34% par rapport à l'amplitude moyenne de l'erreur absolue de prédiction de la première simulation.

Comme les deux applications du chapitre précédent, les temps d'apprentissage sont à-peu-près semblables pour les trois simulations.

En conclusion, nous pouvons affirmer que l'architecture RR²FR est bien applicable à notre système de production en termes de prédiction de b_f . En effet, malgré les multiples paramètres qui peuvent intervenir quant à la valeur finale du Brix du DCT, l'architecture RR²FR utilisant l'algorithme K-moyens a montré une haute performance d'estimation à partir d'un historique comprenant un nombre de 196 observations, seulement, des variables suivantes : b_1 , Tp , b_2 et b_f .

5.3.2.2- Fonction suivi-détection

L'application de notre modèle de suivi-détection, développé au chapitre précédent, à la chaîne automatisée de production installée à l'usine de la société SIAK-MAHBOUBA est maintenant à notre portée. En effet, suite à l'analyse de comportement de la chaîne de production qui a permis l'identification des différents paramètres influant et non influant la performance de la qualité de produit ainsi que du flux de production et au développement de la loi de commande robuste activement dans une première étape et la loi de commande robuste hybride multicritères en second lieu, l'application de notre modèle est facile.

Le modèle type RdPTàO, présenté par la figure 5.19, modélise la fonction suivi-détection de notre système de production. En effet, deux modèles de ce type doivent être implantés : modèle estimé et modèle réel.

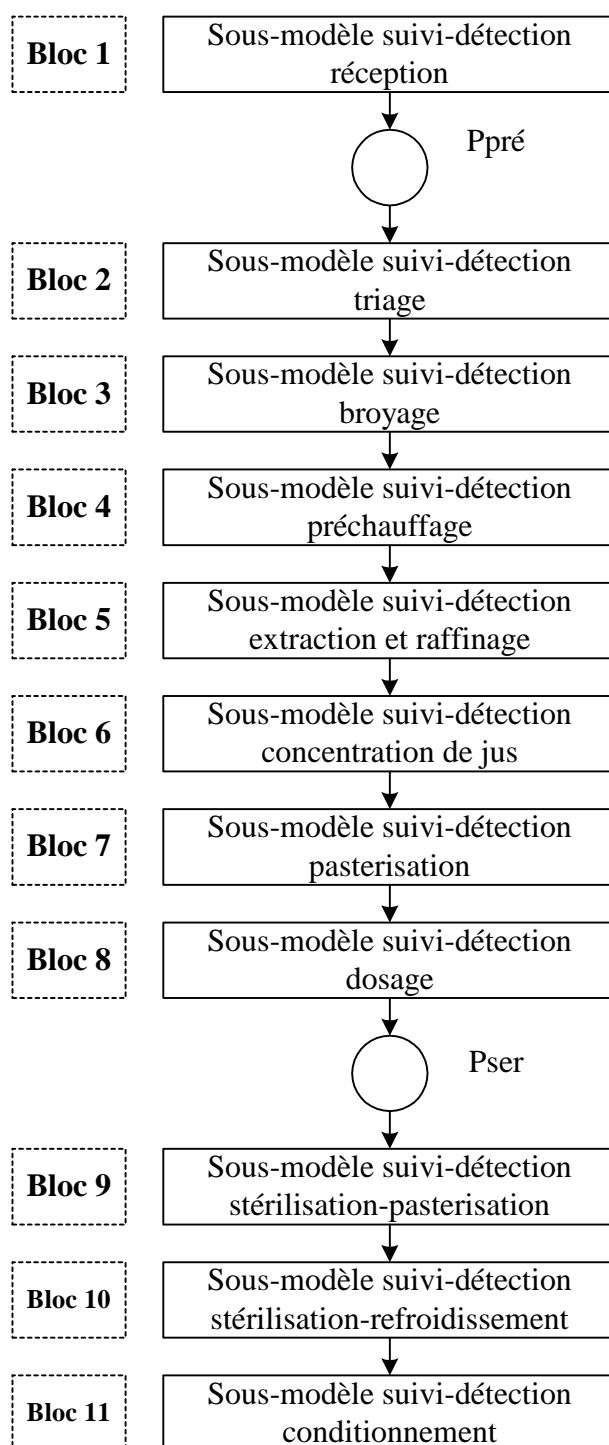


Fig. 5.19- Modèle RdPTàO pour le suivi et la détection

Le modèle de la figure 5.19 est constitué de 11 blocs, comprenant, chacun, le sous modèle de suivi-détection relatif à l'étape élémentaire, et deux places *Ppré* et *Pser* modélisant, respectivement, les deux étapes élémentaires : prélavage et sertissage.

Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire réception modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité Brix b_0 . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire triage modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité coloration de l'eau C_e . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire broyage modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité Brix b_1 . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire préchauffage modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité Température T_p . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire Extraction et raffinage modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité Brix b_i . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire concentration de jus modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité Brix b_2 et le paramètre flux temps de séjour à l'évaporateur $tsep$. Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire pasteurisation modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité température T_{pas} et le paramètre flux temps de séjour au pasteurisateur tsp . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire dosage modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité température T_d . Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire stérilisation (bassin de pasteurisation) modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité température T_{s1} et le paramètre flux temps de séjour au bassin de pasteurisation $tsbp$. Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire stérilisation (bassin de refroidissement) modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre qualité température T_{s2} et le paramètre flux temps de séjour au bassin de pasteurisation $tsrp$. Le sous-modèle modélisant le bloc suivi-détection de l'étape élémentaire conditionnement modélise le mécanisme permettant la détection des symptômes de défaillance suite aux déviations de paramètre flux temps de cycle tc .

En effet, nous pouvons constater aisément que ses blocs peuvent être répartis en trois types :

- blocs suivi-détection temporels : ils permettent la détection des symptômes suites à des déviations temporelles,
- blocs suivi-détection qualité : ils permettent la détection des symptômes suites à des déviations qualité,
- blocs suivi-détection qualité-flux : ils permettent la détection des symptômes suites à des déviations qualité-flux.

En effet, les blocs de notre système se répartissent comme suit :

- blocs suivi-détection temporels : Bloc 1 et Bloc 11 ;
- blocs suivi-détection qualité : Bloc 2, Bloc 3, Bloc 4, Bloc 5 et Bloc 8 ;
- blocs suivi-détection qualité-flux : Bloc 6, Bloc 7, Bloc 9 et Bloc 10.

De ce fait, pour chaque bloc nous proposons pour chaque type un modèle RdPTàO.

Pour un bloc de type suivi-détection temporels, nous proposons le modèle [COM 91] de la figure 5.20.

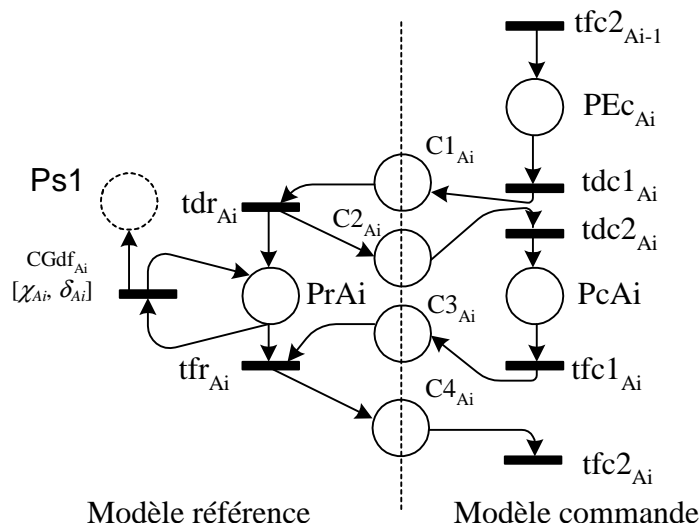


Fig. 5.20- Modèle RdPTàO pour le suivi et la détection des paramètres temporels

Nous pouvons détecter par ce modèle les déviations de type temporel. En effet, nous ne nous sommes pas intéressés que par les symptômes dus aux dates de fin au plus tard de l'activité. Pour ceci, nous avons maintenu des chiens de garde seulement aux niveaux des activités des étapes élémentaires Ai . Les bornes de l'intervalle $[\alpha_{Ai}, \beta_{Ai}]$ sont :

- α_{Ai} = la borne supérieure de la marge de robustesse passive,
- $\beta_{Ai} = \alpha_{Ai} + \varepsilon$; où ε est un réel positif $\gg \alpha_{Ai}$.

Pour les paramètres qui ne font pas partie de la loi de commande robuste hybride multicritères de la chaîne de production de DCT, nous attribuons à l'intervalle $[\alpha_{Ai}, \beta_{Ai}]$ les bornes suivantes :

- α_{Ai} = la borne supérieure de la marge de robustesse active,
- $\beta_{Ai} = \alpha_{Ai} + \varepsilon$; où ε est un réel positif $\gg \alpha_{Ai}$.

Les places modélisant les activités sont $PrAi$ sur le modèle référence et $PcAi$ sur le modèle commande. Les places $C1_{Ai}$, $C2_{Ai}$, $C3_{Ai}$ et $C4_{Ai}$ modélisent les canaux de communication entre le modèle commande et le modèle référence. PEc_{Ai} modélise la place d'attente (stock intermédiaire ou les canaux de transfert pour certaines activités de notre processus) avant que le produit semi-fini passe par l'activité Ai . L'entrée de chaque bloc est la transition $tfc2_{Ai-1}$ et la sortie est la transition $tfc2_{Ai}$. Les significations du reste de la nomenclature et le principe de fonctionnement de ce modèle sont bien expliqués au chapitre précédent.

Pour un bloc de type suivi-détection qualité, le modèle modélisant ce bloc est constitué seulement d'un Bloc de Suivi et de Détection Qualité (B.S.D.Q).

Un bloc de suivi-détection qualité est modélisé par le modèle RdPTàO de la figure 5.21.

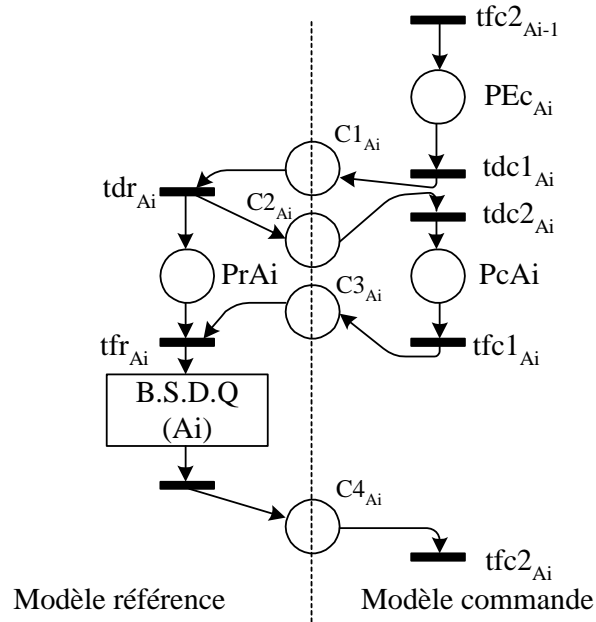


Fig. 5.21- Modèle RdPTàO pour le suivi et la détection des paramètres qualité

Les chiens de gardes ne s'intègrent pas à au modèle de la figure 5.21. Nous modélisons seulement l'activité A_i et le B.S.D.Q.

Le principe de modélisation du bloc B.S.D.Q est bien expliqué au chapitre précédant. Une simple application sur les paramètres qualité de notre système de production, nous permet de construire nos modèles. Sauf que, nous supposons que les transitions de détection des symptômes ne se franchissent que lorsque les valeurs des paramètres sont hors des marges de robustesse passive pour les paramètres de la loi de commande robuste hybride multicritères de la chaîne de production de DCT et des marges de robustesse active pour les autres paramètres.

Nous terminons par le modèle RdPTàO proposé pour la modélisation d'un bloc de type suivi-détection qualité-flux. La figure 5.22 illustre ce modèle.

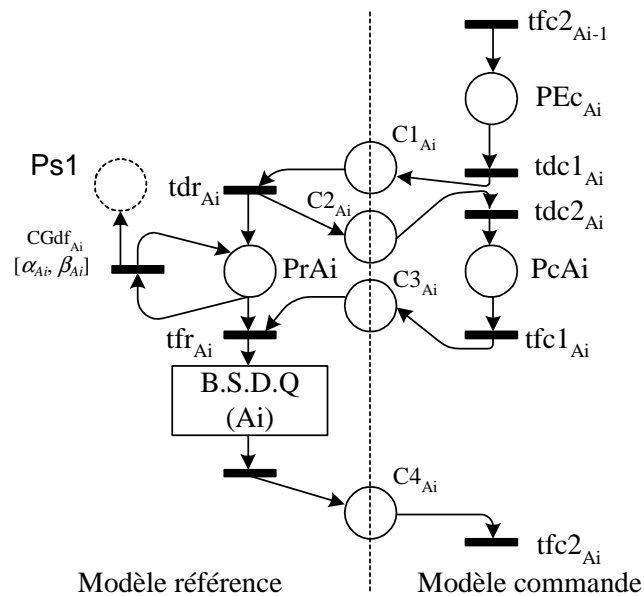


Fig. 5.22- Modèle RdPTàO pour le suivi et la détection des paramètres qualité-flux

Ce modèle est une superposition de deux autres modèles. En effet, nous concevons le même principe de fonctionnement et de modélisation de deux autres modèles.

5.3.3- Synchronisation entre les différents éléments de l'architecture

Les synchronisations entre le modèle commande et le modèle surveillance (au sien de module Surveillance-Commande) d'une part, et entre le procédé et le module Surveillance-Commande d'autre part sont assurées par des voies de communication internes et externes. En effet, les voies de communication externes permettent le transfert des données et des informations entre le procédé et le module Surveillance-Commande. Alors que, les voies de communication internes s'intéressent à l'assurance de circulation des données et des informations au sein du module Surveillance-Commande.

Dés l'installation de deux lois de commande modélisées par RdP à Intervalles, des requêtes d'informations sont produites pour être envoyées au module-composant suivi-détection. Ces requêtes sont, en effet, créées d'une façon que chacune s'intéresse aux informations relatives à un seul paramètre de la loi de commande. Chaque requête, contient les informations relatives à la robustesse (marge de robustesse passive et marge(s) de robustesse active). Lorsqu'on parle d'un paramètre faisant partie de la loi de commande robuste hybride multicritères, la requête d'informations $\Sigma_{Ps,i,j}$ comprend nécessairement les marges de robustesse active et passive, l'étape élémentaire, ainsi que l'ordre de l'étape élémentaire relatif au paramètre (i est l'indice indiquant l'ordre de l'étape élémentaire ; i,j est l'indice relatif au paramètre j de l'étape élémentaire i). Si le paramètre ne fait pas partie de la loi de commande robuste hybride multicritères (elle fait donc partie à loi de commande robuste activement multicritères), la requête d'informations $\Sigma_{PNS,i,j}$ comprend nécessairement les marges de robustesse active, l'étape élémentaire, ainsi que l'ordre de l'étape élémentaire relatif au paramètre.

Les voies de communication externes et internes assurent la circulation des requêtes. Des consignes de commande décrites par l'ensemble des requêtes seront envoyées vers le procédé pour que les réglages des actionnaires et des pré-actionneurs soient effectués. De même, les informations portant sur les marges de robustesse, décrites au sein des l'ensemble des requêtes, seront envoyées vers le module-composant de la fonction suivi-détection pour être installées sur les conditions des transitions de fonctionnement normale et de détection des symptômes du modèle RdPTàO.

5.4- Conclusion

Dans ce chapitre une application concrète de l'architecture « Surveillance indirecte suite à une commande robuste » à une entreprise d'industries alimentaires est présentée. Cette architecture est bien applicable à ce type d'industrie.

Nous avons commencé par la présentation de l'entreprise tout en faisant une analyse structurelle et fonctionnelle de la chaîne automatisée de production de DCT en vu de présenter la société dans son environnement productif. Notre analyse converge à extraire les besoins en données. En effet, nous avons déterminé les paramètres significatifs qui peuvent influencer sévèrement par leurs déviations la qualité totale du produit DCT.

Malgré que l'application de l'aspect théorique, présenté au chapitre trois, à notre système est difficile, nous avons pu construire le modèle de la loi de commande robuste hybride multicritères.

Aussi, dans une étape ultérieure, nous avons implanté les fonctions du modèle surveillance : pronostic et suivi-détection. En effet l'architecture RR^2FR , appliquée à la prédiction du Brix du produit fini, a montré, une autre fois, une bonne aptitude en termes de minimisation de l'erreur de prédiction. Les modèles de suivi-détection proposées pour notre système consistent à répondre aux besoins de l'étape élémentaire en termes de sa composition en paramètres : qualité, flux ou flux et qualité. Nous avons terminé par l'explication de principe de communication et synchronisation entre les différents éléments de l'architecture « Surveillance indirecte suite à une commande robuste ».

Conclusion générale

Conclusion générale

Le travail présenté dans ce manuscrit s'inscrit dans le cadre de conception et de développement des architectures de Surveillance-Commande des Systèmes Flexibles de Production Manufacturière.

L'architecture de Commande-Surveillance, que nous avons proposé, a l'objectif de minimiser l'appel à la maintenance corrective. Ceci est afin d'éviter les cessations de fonctionnement du procédé qui provoquent des conséquences dangereuses influant directement et indirectement la performance globale de l'entreprise au niveau technique et/ou économique.

Notre architecture développée « Surveillance Indirecte suite à une Commande Robuste » est composée de deux modèles : un modèle commande et un modèle surveillance. Le premier s'appuie sur la modélisation de la robustesse du produit de point de vue qualité et flux. Il est constitué de deux fonctions qui sont la fonction commande robuste et la fonction décision. Le second est conçu d'une manière déterminante les événements qui vont cesser probablement le fonctionnement normal du système à partir d'une surveillance des paramètres de la loi de commande. Il est composé de trois fonctions qui sont la fonction pronostic, la fonction suivi-détection et la fonction diagnostic.

Les fonctions objectives de cette architecture sont celles qui influent la conservation des spécificités de système en terme de maintien de la robustesse passive du système. Ces spécificités sont normalement maintenues et assurées par le biais de trois fonctions principales qui sont : la fonction commande robuste du bloc Commande ainsi que les fonctions pronostic et suivi-détection du bloc Surveillance. Les fonctions décision et diagnostic, ne présentent pas en soient un catalyseur pour la réussite de notre architecture.

Notre contribution comporte cinq volets:

- le premier volet concerne l'architecture « Surveillance Indirecte suite à une Commande Robuste » qui est appuyée sur la surveillance des paramètres d'une loi de commande comme un outil de maintien de la robustesse. Elle consiste à surveiller les paramètres de la loi de commande robuste par le suivi momentané de leurs évolutions réelles et estimées. Ceci est en vue de détecter leurs déviations dans un contexte de robustesse active et de préjuger les événements fatales qui peuvent influent la robustesse globale du système ;
- le deuxième volet propose une loi de commande robuste hybride (robustesse passive et active) multicritères (flux et qualité). Par rapport aux modèles classiques cités dans la littérature, notre modèle porte deux apports. Le premier consiste à la modélisation hybride qui intègre les marges de robustesse active et passive au sein d'un même modèle. Le second s'attache à la modélisation de la robustesse de deux critères à la fois : flux (temporel) et qualité (non temporel). Les RdP à Intervalles qui ont été introduits pour augmenter le champ d'application des RdP P-temporels ont été exploités. le développement d'un tel modèle nous a permis l'observation et la constatation de la nature de robustesse des différents composants du système (paramètres, ressources, circuits élémentaires et système de production). Un concept

théorique ainsi que des applications ont bien expliqué notre approche de modélisation des lois de commandes robustes hybrides multicritères ;

- le troisième volet consiste à une nouvelle architecture « Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales (RR²FR) » de prédiction désignée pour la fonction pronostic du modèle surveillance. En plus, une amélioration de l'algorithme K-moyens d'apprentissage non supervisé est proposée. Le réseau RR²FR qui présente une amélioration de l'architecture RRFR proposée par [ZEM 03] est montré performant en terme de minimisation de l'erreur absolue de prédiction. Les résultats de simulations appliquées à la prédiction des valeurs de la série temporelle *MackeyGlass* et à la prédiction d'une concentration de sortie en CO₂ d'un four à gaz ont montrée que cette architecture est meilleure que le RFR classique et les RRFR de [ZEM 03]. Aussi l'apport de l'algorithme k-moyens amélioré est bien constaté ;
- le quatrième volet concerne le modèle RdPTàO de suivi-détection développé pour remplir les objectifs de suivi des paramètres qualitatifs et quantitatifs du produit et de détection des symptômes de défaillances. Nous avons intégré des Blocs de Suivi et de Détection Qualité au modèle de suivi-détection temporel de [COM 91] ;
- le dernier volet étend l'application de l'architecture « Surveillance Indirecte suite à une Commande Robuste » à un système industriel. L'exemple industriel choisi est une chaîne de production de DCT installée à l'usine de la société d'industries alimentaires SIAK-MAHBOUBA. Malgré que l'application de l'aspect théorique reste toujours difficile, nous avons pu appliquer notre architecture à travers ces différents constituants et étapes.

Concernant les perspectives de cette thèse, plusieurs axes de recherche peuvent se dégager pour être envisagés :

- afin de confirmer son apport, il serait intéressant d'appliquer l'architecture à différents types des industries telles que : l'industrie pétrolière, l'industrie pétrochimique, l'industrie chimique, l'industrie avionique, etc.;
- la mise en œuvre de l'architecture « Surveillance Indirecte suite à une Commande Robuste » dans un contexte des ateliers de type job-shop serait très faisable en terme d'augmentation des champs d'applications. Ceci nécessite surtout une reformulation de la commande robuste en termes d'aspect méthodologique et théorique de modélisation ainsi que d'observation et de constatation de la nature de robustesse.
- concernant l'architecture RR²FR préposée, l'étude d'optimisation du nombre n des RRFR successifs sur la performance d'une architecture de type RRⁿRF est une perspective importante.

Références bibliographiques

Références bibliographiques

- [AGH 97] Aghasaryan A., Boubour R., Fabre E., Jard C. et Benveniste A., “A petri net approach to fault detection and diagnosis in distributed systems”, Technical Report 1117. INRIA. Rennes, France, 1997.
- [AGT 74] Agteberg F.P., Geomathematics, Elsevier, Amsterdam, 1974.
- [ALU 94] ALUR R., DILL D., « A Theory of Timed Automata », Theoretical Computer Science, vol. 126, no 2, 1994, p. 183–235, Elsevier Science.
- [AND 96] Andreu D., « Commande et supervision des procédés discontinus : une approche hybride ». Thèse de Doctorat, Université Paul Sabatier, Toulouse III, 1996.
- [ART 97] Artigues C., « Ordonnancement en temps réel d’ateliers avec temps de préparation des ressources ». Thèse de Doctorat, Université Paul Sabatier de Toulouse, décembre 1997.
- [BAS 88] Basseville M., “Detecting Changes in Signals and Systems – A Survey”, Automatica, Vol. 24, N°3, pp. 309-326, 1988.
- [BEN 96] Ben Abdallah I. « Méthodes d’allocation de ressources dans les Systèmes Flexibles de Production Manufacturière fondées sur l’analyse structurelle des réseaux de petri » Thèse de doctorat école centrale de paris. Juillet 1996.
- [BER 00] Berruet P., Toguyeni A. K. A., Elkhatabi S., Craye E., “Toward an implementation of recovery procedures for flexible manufacturing systems supervision”. Computers in Industry, vol. 43, pages 227 – 236, 2000.
- [BER 05] P. Bonhomme, “Control of time critical systems using partial order,” in IMAC’05, Paris, France, July 2005.
- [BER 91] Berthomieu B., Diaz M., « Modeling and verification of time dependent systems using time Petri nets », IEEE transactions on software engineering, vol. 17, no 3, pp. 259-273, 1991.
- [BER 95] Berthold M. R., Diamond J., “Boosting the Performance of RBF Networks with Dynamic Decay Adjustment”, Advances in Neural Information Processing Systems, Gerald Tesauro, David S. Touretzky, and Todd K. Leen editors, vol. 7, pp. 521-528, MIT Press, Cambridge, MA, 1995.
- [BER 96] Bernauer E., « Les réseaux de neurones et l’aide au diagnostic : un modèle de neurones bouclés pour l’apprentissage de séquences temporelles », thèse de doctorat, LAAS/Toulouse 1996.
- [BER 98] Berruet P., « Contribution au recouvrement des systèmes flexibles de production manufacturière : analyse de la tolérance et reconfiguration ». Thèse de Doctorat, Université des Sciences et Techniques de Lille, décembre 1998.
- [BHA 03] Bhatikar S.R., and Mahajan R.L., “Artificial neural-network-based diagnosis of CVD barrel reactor”, IEEE Transactions on Semiconductor Manufacturing, 15(1): pp. 71-78, 2002.

- [BLA 06] Blanke M., Kinnaert M., Lunze J., Staroswiecki M., “Diagnosis and Fault-Tolerant Control”, 2nd edition Springer., 2006.
- [BON 00] Bonhomme, P. Aygalinc, S. Calvez. “Using Robustness Properties for Multi-Products Processing”. IFAC/IFIP/IEEE, 2nd Conférence on Management and Control of Production and Logistics, MCPL'2000, 5-8 July 2000, Grenoble, FRANCE.
- [BON 01] Bonhomme P., « Réseaux de petri P-temporels : contributions à la commande robuste », Thèse de Doctorat, Université de Savoie, décembre 2001.
- [BON 93] de Bonneval A., « Mécanismes de reprise dans les systèmes de commande à événements discrets ». Thèse de Doctorat, Université Paul Sabatier de Toulouse, septembre 1993.
- [BOU 03] Boufaied A., «Contribution à la Surveillance Distribuée Des Systèmes à Événements Discrets Complexes », Thèse de doctorat, Université Paul Sabatier de Toulouse, France, décembre, 2003. production ». Thèse de Doctorat, LAI, Lyon, juillet 1997.
- [BOU 96] Bouchon-Meunier B., « La logique floue et ses applications », Vie Artificielle. 258 p. Addison-Wesley, France, 1996.
- [BRO 88] Broomhead D.S., Lowe D., “Multivariable fonctional interpolation and adaptive networks”, *Complex Systems*, Vol. 2, p. 321-355, 1988.
- [BUY 08] Buyurgan N., Saygin C., “Application of the analytical hierarchy process for real-time scheduling and part routing in advanced manufacturing systems”, *Journal of Manufacturing Systems*, Volume 27, Issue 3, Pages 101-110, July 2008.
- [CAM 97] Camus H., « Conduite des systèmes flexibles de production manufacturière par composition des régimes permanents cycliques : modélisation et évaluation des performances à l'aide des réseaux de Petri ». Thèse de Doctorat, Université de Lille 1, mai 1997.
- [CAS 93] Cassandras C. G., “Discrete event systems : modeling and performance analysis”. Aksen Associates incorporated publishers. ISBN 0-256-11212-6, 1993.
- [CHA 95] Chaillet-Subias A. “Approche multi-modèles pour la commande et la surveillance en temps réel des systèmes à événements discrets”, Thèse de Doctorat, Université Paul Sabatier, Décembre 1995.
- [CHA 01] Chappelier J.-C., Gori M., Grumbach A., "Time in Connectionist Models", in *Sequence Learning: Paradigms, Algorithms, and Applications*, R. Sun and C.L. Giles editors, Lecture Notes in Artificial Intelligence Serie, Vol. 1828, pp 105-134, Springer, 2001.
- [CHN 96] Chng E. S., Chen S., and Mulgrew B., “Gradient Radial Basis Function Networks for Nonlinear and Nonstationary Time Series Prediction” *IEEE Transactions on Neural Networks*, Vol. 7, Number. 1, pp. 190-194, 1996.
- [COL 03] Collart Dutilleul S., H. Dhouibi, E. Craye, “Internal Robustness of Discret Event System with internal constraints in repetitive functioning mode”. ACS'2003 conference, Miedzyzdroje, Poland, 2003.

- [COL 94] Collart Dutilleul S., J. P. Denat, W. Khansa, « Commande robuste d'un atelier flot sans stocks et sans attentes ». Automatique, Informatique Industrielle (APII), vol. 28, n°6/1994, 1994.
- [COL 95] Collart Dutilleul S., J. P. Denat, « Use of periodic controlled Petri nets for Discrete Event Dynamical Control Synthesis ». Ecc'95, 1995.
- [COL 97] Collart Dutilleul S., « Commande Robuste d'Ateliers à Contraintes de Temps de Séjour : Application à la Galvanoplastie ». Thèse de Doctorat, Université de Savoie, décembre 1997.
- [COM 00] Cambacau M., Berruet P., Zamai E. « Supervision and Monitoring of Production Systems ». Seconde Conférence Management et Conduite en Production et Logistique, MCPL'2000, Grenoble, juillet 2000.
- [COM 91] Combacau M., "Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles", Thèse de doctorat, Université Paul Sabatier, Toulouse, Décembre 1991.
- [COR, 96] ARTEMIS-IMAG, ERIHST, LAG et LAMLII-CESALP <<CORINE>> : "COduite Robuste et INtelligente dans les Entreprises manufacturières", Rapport Technique, Mai 1996.
- [COU 07] Coulibaly P., Evora N.D., "Comparison of neural network methods for infilling missing daily weather records" in Journal of Hydrology, Volume 341, Issues 1-2, 20, pp. 27-41, July 2007.
- [DAS 02] Da Silveira M. R., Combacau M., Subias A. "From centralized to distributed models: A systematic procedure based on Petri nets". SMC – IEEE International Conference on Systems, Man and Cybernetics, Hammamet, Tunisia.2002.
- [DAS 03a] Da Silveira M. R., Combacau M., Boufaied A. "Prognosis and recovery evaluation on flexible manufacturing supervision". Journal of Decision Systems, vol. 12, No. 1, 2003.
- [DAS 03b] DA Silveira M., « Sur la distribution avec redondance partielle de modèles à événements discrets pour la supervision de procédés industriels », Thèse de doctorat, Université Paul Sabatier de Toulouse, France, juillet, 2003.
- [DHO 05a] Dhoubi H., « Utilisation des réseaux de pétri à intervalles pour la régulation d'une qualité : application à une manufacture de tabac». Thèse de Doctorat, Ecole Centrale de Lille, Décembre 2005.
- [DHO 05b] Dhoubi H., Collart Dutilleul S., E.Craye., L. Nabli "Computing Intervals of Intervals Constrained Petri Net: a tobacco manufacturing application", IMACS 05 conference, Paris, juillet 2005.
- [DIA 03] M. Diaz, « Vérification et mise en oeuvre des réseaux de Petri ». Paris, Hermès Sciences Publication, 2003.
- [DUD 73] Duda R. O., E.H.P., "Pattern classification and scene analysis", John Wiley & Sons, 1973.
- [DUP 99] M. Dupas, Y. Salliez, T. Berger, R. Soenen « Reconfiguration des systèmes automatisés de production », 3ème Congrès International de Génie Industriel,

- Montréal (Canada), 1999, vol. 3, pp. 2007-2012.
- [ELK 93] El Khattabi S., « Intégration de la surveillance de bas niveau dans la conception des systèmes à événements discrets : application aux systèmes de production flexibles ». Thèse de Doctorat, Université des Sciences et Technologies de Lille, septembre 1993.
- [ELK 93] S. El Khattabi, « Intégration de la surveillance de bas niveau dans la conception des systèmes à événements discrets : application aux systèmes de production flexibles ». Thèse de Doctorat, Université des Sciences et Technologies de Lille, septembre 1993.
- [FRA 90] Frank P. M., « Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge Based Redundancy – A survey and New Results, Automatica, Vol. 26, p. 459-474, 1990.
- [FRA 95] Frasconi P., Gori M., Maggini M et Soda G., “Unified Integration of Explicit Knowledge and Learning by Example in Recurrent Networks”, IEEE Transaction on Knowledge and Data Engineering, Vol. 7, N° 2, pp. 340-346, April 1995.
- [FUK 90] Fukunaga K., “Statistical Pattern Recognition”, Academic Press, 2e édition, 1990.
- [GAU 00] Gaury, E., “Designing pull production control systems : customization and robustness”. Thèse de Doctorat, Université Blaise Pascal, France, 2000.
- [HAC 75] HACK M. “Petri Nets Languages” Comput. struct. Gr. Memo 124. Project HAC. , M.I.T. Cambridge Mass, Juin 1975.
- [HAR 71] Hardy R.L., “Multiquadratic equations of topography and other irregular surfaces”, J. Geophys. Res, 76:1905-1915, 1971.
- [HAR 90] Harris, L., “Mind over manufacturing”. AI. Expert, Nov 1990.
- [HAM 09] Hamani N., Dangoumau N., Craye E., “Reactive mode handling of flexible manufacturing systems” Mathematics and Computers in Simulation, Volume 79, Issue 5, Pages 1421-1439, January 2009.
- [HIL 89] M. Hervé Hillion “Modélisation et analyse des systèmes de production discrets par les réseaux de Petri. ». Thèse de Doctorat, L’université Pierre et Marie Curie Paris VI, 23 Janvier 1989.
- [HON 09] Hong X., Chen S., “A New RBF Neural Network With Boundary Value Constraints”, IEEE Transactions on Systems, Man, and Cybernetics, Part B, Volume 39, Issue 1, pp.298 – 303, February 2009.
- [HUB 81] Huber P.J., « Robust Statistics », Jhon Wiley et Sons 1981.
- [HUD 92] Hudak M.J., “RCE Classifiers: Theory and Practice”, in Cybernetics and systems, vol. 23, pp.483-515, 1992.
- [IOR 02] Iordache M. V., Moody J., and Antsaklis P. J., “Synthesis of deadlock prevention supervisors using Petri nets,” IEEE Trans. Robot. Autom., vol. 18, no. 1, pp. 59–68, Feb. 2002.
- [ISE 84] Isermann R., “Process Fault Detection Based on Modeling and Estimation Methods – A Survey”, Automatica, Vol. 20, N°4, p. 387-404, 1984.

- [IWA 86] Iwasaki Y., Simon H.A. "Causality in device behaviour", *Artificial Intelligence*, vol. 29, n.1, pp. 3-33, 1986.
- [JER 06] Jerbi, N., Collart Dutilleul, S., Craye, E., Benrejeb, M. "Time Disturbances and Filtering of Sensors Signals in Tolerant Multi product Job-shops with Time Constraints", *International Journal of Computers, Communications & control*, Vol. I, n° 4, pp. 61–72, 2006.
- [JIA 06] Jianming L., Yonggon Le., Sudhoff, S.D., Zak S.H., "Self-Organizing Radial Basis Function Network for Real-Time Approximation of Continuous-Time Dynamical Systems" *IEEE Transactions on Neural Networks*, Volume 19, Issue 3, Page(s):460 – 474, March 2008.
- [KAF 01] Kaffel H., "La maintenance distribuée : concept, évaluation et mise en œuvre" Thèse, Faculté des études **supérieures de l'Université Laval, Canada, 2001.**
- [KHA 96] Khansa, W. Aygalinc, P., Denat, J. P. "Structural analysis of P-Time Petri Nets", *CESA' Lille*, France, pp. 127–136, Juillet 1996.
- [KHA 97] Khansa, W. « Réseaux de Petri P-temporels: contribution à l'étude des systèmes événements Discrets ». Thèse de Doctorat de l'Université de Savoie, France, mars 1997.
- [KAZ 01] Kazushi O., Kang G., "Model-based Control for Reconfigurable Manufacturing Systems" *ICRA 2001-IEEE, International Conference on Robotics and Automation*, Séoul, Corée, 21-26 Mai, **2001**
- [KLE 86] De Kleer J., "Theories of causal ordering", *Artificial Intelligence*, vol. 29, n.1, pp. 33-62, 1986.
- [KOI 94] Koivo H.N., "artificial neural networks in fault diagnosis and control", *control inengineering practice*, vol.2, n°1, pp. 89-101, 1994.
- [KOR 05] Koren Y., *Reconfigurable Manufacturing and Beyond*, In the summary of keynote Speech of CIRP05 3rd International Conference on Reconfigurable Manufacturing, Ann Arbor, Michigan, Etats-Unis, 10-12 Mai 2005
- [KOR 98] Korbaa O., « Commande cyclique des systèmes flexibles de production manufacturière à l'aide des réseaux de Petri : de la planification à l'ordonnancement des régimes ansitoires ». Thèse de Doctorat, Université de Lille1, juin 1998.
- [KOR99] Koren Y., Heisel U., Jovane F., Moriwoki T., Pritshow G., Ulosy A. G. et Van Bruseel H., "Reconfigurable Manufacturing Systems", *Annals of the CIRP*, Vol.48, pp.527-540, 1999.
- [KUM 96] Kumar R., Holloway L. E., "Supervisory Control of Deterministic Petri Nets with Regular Specification Languages", *IEEE Transactions on Automatic Control*, Vol. 41, Issue 2, pp.245-249, Février 1996.
- [LAP 92] Laprie J., C., "Dependability basic concepts and terminology", ISBN : 3-211-82296-8, 1992.
- [LEE 04] Lee, R.S.T., "iJADE stock advisor: an intelligent agent based stock prediction system using hybrid RBF recurrent network" in *IEEE Transactions on Systems*,

- Man and Cybernetics, Part A, Volume 34, Issue 3, Page(s):421 – 428, May 2004.
- [LEE 06] LEE E.J., « Reconfiguration dynamique de la commande d'un système manufacturier : approche par la synthèse de la commande », Thèse de Doctorat de l'Ecole Centrale de Lille, 2006.
- [LI 02] Li Z., Zhou M. and Jeng M., "A Maximally Permissive Deadlock Prevention Policy for FMS Based on Petri Net Siphon Control and the Theory of Regions," IEEE Transactions on Automation Science and Engineering, Volume 5, Issue 1, pp182 – 188, January 2008.
- [LON 93] J. Long et B. Descotes-Genon, "Flow Optimization Method for Control Synthesis of Flexible Manufacturing Systems Modeled by Controlled Timed Petri Nets", IEEE International Conference on Robotics and Automation, Atlanta, Georgia, Vol. 1, pp. 598–603, Mai 1993.
- [LY 00] Ly F., Toguyeni A. K. A., Craye E. , "Indirect predictive monitoring in flexible manufacturing systems" ,Robotics and Computer-Integrated Manufacturing, Volume 16, Issue 5, Pages 321-338, October 2000.
- [LY 99] Ly F. « Optimisation de la maintenance dans les SFPM : Approche par la structuration et la surveillance prédictive indirecte ». Thèse de Doctorat de l'Ecole Centrale de Lille, 1999.
- [MAB 96] M. Mabrouk « Proposition d'une méthode et d'un outil d'aide à la reconfiguration des systèmes automatisés de production », Thèse de doctorat, Université de Valenciennes et du Hainaut Cambrésis, 28 mai 1996.
- [MER 74] MERLIN P. M., "A study of the recoverability of computing systems", PhD thesis, Department of Information and Computer Science, University of California, Irvine, CA, 1974.
- [MHA 06] M'halla A., « Supervision par réseaux de Pétri flous:application à un brûleur à fuel », Mémoire de master à l' Ecole National d'Ingénieurs de Monastir, juin 2006
- [MIN 03] Minca E., Racoceanu D., and Zerhouni N., « Approche de détection par réseau de Petri flou », Conférence Pentom2003 Performance et Nouvelles Technologies en Maintenance, Valenciennes, France, CD ROM, March, 2003.
- [MIN 04] Minca, E., « Contribution à la surveillance des systèmes de production à l'aide des réseaux de Petri flous : Application à la e-maintenance », Thèse de doctorat, Université de Franche-Comté, 2004.
- [MOA 81] Moalla M ., « Spécification et conception sûre d'automatismes discrets complexes, basées sur l'utilisation du GRAFCET et des réseaux de PETRI », Doctorat d'état Institut National Polytechnique de Grenoble - INPG Université Joseph-Fourier - Grenoble I, 1981.
- [MON 97] Montmain J., "From diapason research program to its industrial application in nuclear fuel reprocessing", IFAC Symposium on Fault Detection, Supervision and Safety for Technical Process -SAFEPROCESS'97, pp. 209-216, 1997.
- [MUR 89] T. Murata, "Petri Nets: Proprieties, Analysis and Applications", Proceedings of the IEEE of Electrical and Electronics Engineers, Inc. Vol. 77, n°4, Avril 1989.

- [NAB 00] Nabli L. « Surveillance prédictive conditionnelle prévisionnelle indirecte d'une unité de filature textile : Approche par la qualité ». Thèse de Doctorat de l'Ecole Centrale de Lille, 2000.
- [NAB 06] Nabli L., Dhouibi H., Collart Dutilleul S, Craye E. « Utilisation de la logique floue pour la surveillance prédictive indirecte d'un système de production : cas d'une manufacture de tabac », JTEA'06 Hammamet, Tunisie, Mai 2006.
- [NAB 08] Nabli L., **Telmoudi A. J.**, M'hiri R. "Modeling and analysis of a robust control of manufacturing systems: flow-quality approach", International Journal of Computer, Information, and Systems Science, and Engineering (IJCISSE), Volume 2 Number 1, pp. 43-51, 2008.
- [NAB 98] Nabli L., Toguyéni A.K.A., Craye E., Annabi M., "A monitoring method based on fuzzy sets", CESA'98 IMACS Multiconférence, Computational engineering in systems application, Nabeul, Tunisie, avril 1998
- [NAB 99] Nabney I.T. , "Efficient training of RBF networks for classification" Ninth International Conference on Artificial Neural Networks, 1999. ICANN 99, Vol.1, pp. 210-215 vol.1, Edinburgh, UK, 1999.
- [NOU 04] Nourelfath M., Niel E. "Modular supervisory control of an experimental automated manufacturing system", Control Engineering Practice 12, 205–216, 2004
- [NOU 97] Nourelfath M. « Extension de la théorie de la supervision à la surveillance et à la commande des systèmes à événements discrets : Application à la sécurité opérationnelle des systèmes de production ». Thèse de Doctorat, LAI, Lyon, juillet 1997.
- [PAL 05] Palluat N., Racoceanu D., Zerhouni N., « Utilisation des réseaux de neurones temporels pour le pronostic et la surveillance dynamique : Etude comparative de trois réseaux de neurones récurrents » Revue d'intelligence artificielle, vol. 19, no6, pp., 19913-950, 2005.
- [PAR 92] Parayre, T. « Vers une méthode d'exploitation des systèmes automatisés de production », Thèse de Doctorat, Université de Valenciennes et du Hainaut Cambrésis, mars 1992.
- [PAT 99] Patton R. J., Chen J., "Robust model-based fault diagnosis for dynamic systems", Boston : Kluwer Academic Publishers, 1999.
- [PRO 94] Proth J-M., Xie X.L., « Les réseaux de Petri pour la conception et la gestion des systèmes de production », édition Masson 1994.
- [RAF 09] Rafajłowicz E., Skubalska-Rafajłowicz E., "RBF nets for approximating an object's boundary by image random sampling » in Nonlinear Analysis: Theory, Methods & Applications, In Press, Corrected Proof, Available online 24 January 2009.
- [REN 89] Renals S., and Rohwer R., "Phoneme classification experiments using radial basis functions" Proceedings of the international Joint Conference on Neural Networks, p. I-461-I-467, Washington D.C., June 1989. IEEE TAB Neural Network

- Committee.
- [RIC 87] Rich S.H., and Venkatasubramanian V., “Model-based reasoning in diagnostic expert systems for chemical process plants”, *Computers & Chemical Engineering*, 11, pp. 111-122, 1987.
- [ROJ 00] Rojas I., Pomares H., Gonzalez, J., Ros, E., Salmeron, M., Ortega, J., Prieto, A., “A new radial basis function networks structure: application to timeseries prediction”, *IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, vol.4, pp.449 - 454, 2000.
- [ROU 88] Roubellat F. et Thomas V., « Une méthode et un logiciel pour l’ordonnement en temps réel d’ateliers ». *APII 1988*, vol 22, p. 419-439.
- [ROY 97] Roy B. « Un chaînon manquant en Recherche Opérationnelle et Aide à la Décision : les conclusions robustes ». *Cahier du Laboratoire d’Analyse et de Modélisation de Systèmes pour l’Aide à la Décision*, N° 144, Université de Paris Dauphine 1997.
- [RUI 01] Ruiz, D., Nougues J.M., and Puigjaner L., “Fault diagnosis support system for complex chemical plants”, *Computers & Chemical Engineering*, 25(1): pp. 151-160, 2001.
- [ROY 97] Roy B. « Un chaînon manquant en Recherche Opérationnelle et Aide à la Décision: les conclusions robustes ». *Cahier du Laboratoire d’Analyse et de Modélisation de Systèmes pour l’Aide à la Décision*, N° 144, Université de Paris Dauphine 1997.
- [SIL 96] Silva M., “Petri net for the design and operation of Manufacturing Systems”. *17th International Conference on application and theory of Petri Nets- Manufacturing and Petri Nets*, 31-61, Osaka, Japan, 1996.
- [TAW 95] Tawegoum R. « Contrôle temps réel du déroulement des opérations dans les systèmes de production flexibles ». *Thèse de Doctorat*, Université de Lille 1, avril 1995.
- [TEL 08a] **Telmoudi A. J.**, Nabli L., Bourjault A. “The use of redundancy to evaluate the manufacturing system total robustness: A quality approach”, *IEEE- ISCCSP 2008*, St. Julians, Malte, pp. 440 - 446, Mars 2008.
- [TEL 08b] **Telmoudi A. J.**, Nabli L., Bourjault A. « Évaluation de la robustesse des SFPM par étude de la redondance de la robustesse modulaire : approche flux », *MOSIM 2008*, Paris, France, Volume1, pp. 290 - 297, Avril 2008.
- [TEL 08c] **Telmoudi A. J.**, Nabli L., Bourjault A. “Symptoms detection in FMS through performance indicators follow-up: A quality-flow approach”, *IEEE-CMD 2008*, Pékin, Chine, pp. 803-807, Avril 2008.
- [TEL 08d] **Telmoudi A. J.**, Nabli L., M’hiri R. « Conception d’une commande robuste pour une ressource de production : approche flux-qualité », *GEI’08*, Sousse, Tunisie, CD-ROM, Mars 2008.
- [TEL 08e] **Telmoudi A. J.**, Nabli L., M’hiri R. “Robust control of a manufacturing system: flow-quality approach”, *13th IEEE-ETFA 2008* –, Hamburg , Allemagne, pp. 137-

142 Septembre, 2008.

- [TEL 09a] **Telmoudi A. J.**, Nabli L., M'hiri R. "Modeling Method of Robust Control Laws for Manufacturing System to Temporal and non Temporal Constraints through Petri Nets", International Review on Computers and Software, Volume 4 Number 2, pp. 266 - 277, Mars, 2009.
- [TEL 09b] **Telmoudi A. J.**, Barkit S., Nabli L., Chetouani Y, M'hiri R. "Réseau de Neurones doublement Récurrent à Base de Fonctions Radiales pour le pronostic", Journal Européen des Systèmes Automatisés (JESA) Vol. 43 N°7-8-9/Août-Novembre, pp.871-887, 2009
- [TEL 09c] **Telmoudi A. J.**, Nabli L., M'hiri R. "Modeling of robustness margins of the control of a Predictive Control-Supervisory architecture", Journal of Universal Computer Science (J.UCS) (facteur d'impact : 0,699), Volume 15 (17) pp. 3231-3245, 2009.
- [TEL 11] **Telmoudi A. J.**, Nabli L., Tlijani H., Ali M., M'hiri R. "A New Radial Basis Function Neural Network for Time-Series Prediction", International Journal of Information Technology & Decision Making (facteur d'impact : 1,312), à apparaître en 2011.
- [TOG 91] Toguyeni A.K.A., « From the treatment of failure to the management of working degraded modes ». Proceedings of IMACS'91, may 91, pp 595-601, Lille-France.
- [TOG 92] Toguyeni A.K.A., « Surveillance et diagnostic en ligne dans les ateliers flexibles de l'industrie manufacturière ». Thèse de Doctorat, Université de Lille, novembre 1992.
- [TOG 96] Toguyeni A.K.A., Elkhatabi S., Craye E. « fonctionnal and/or structural approach for the supervision of flexible manufacturing system ». IMACS Multiconference, Compitational Engeniring in Systems Applications. CESA'96, Lille france ,Juillet, 1992.
- [TOG03] Toguyéni A.K.A., Berruet P. et Craye E., Models and algorithms for failure diagnosis and recovery in FMS, International Journal of Flexible Manufacturing Systems, Vol. 15, N°1, pp. 57-85, Janvier 2003.
- [TOG05] Toguyéni A.K.A., Korbaa O., Indirect monitoring of the failures of a flexible manufacturing system under cyclic scheduling, Robotic and Computer-Integrated Manufacturing, Vol. 21, pp. 1-10, 2005.
- [UZA 02] M. Uzam, "An optimal deadlock prevention policy for flexible manufacturing systems using Petri net models with resources and the theory of regions," Int. J. Adv. Manuf. Tech., vol. 19, pp. 192–208, February 2002.
- [VAL 01] R. Valette, « cours sur les réseaux de Petri ». 2001 www.laas.fr/~robert/enseignement.d
- [WAN 02] Wang T.Y., and Chen L.H., "Mean shifts detection and classification in multivariate process: a neural-fuzzy approach", Journal of Intelligent Manufacturing, 13(3): p. 211-221, 2002.
- [WID 60] Widrow B., Hoff M. E., "Adaptive switching circuits", dans 1960 IRE WESCON

- Convention Record, New York : IRE, p. 96-104, 1960.
- [YAO 06] Yao Y., Lian Z., Hou Z.. “An Innovative Air-conditioning Load Forecasting Model Based on RBF Neural Network and Combined Residual Error Correction”, *International Journal of Refrigeration* 29, pp. 528–538, 2006.
- [ZAM 97] E. Zamai « Architecture de Surveillance-Commande pour les Systèmes à Evénements Discrets Complexes ». Thèse de Doctorat, Université Paul Sabatier de Toulouse, septembre 1997.
- [ZEM 01] Zemouri R., Racoceanu D., Zerhouni N., “The RRBF : Dynamic representation of time in radial basis function network” *Proc. of the 8th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA'2001*, vol. 2, pp.737-740, Antibes, Juan les Pins, 15-18 October 2001.
- [ZEM 03] Zemouri R., Racoceanu D., Zerhouni N. “Recurrent Radial Basis Function network for Time-Series Prediction”, *Engineering Applications of Artificial Intelligence* (Elsevier Science), Volume 16, Issue 5-6, pp. 453-463, November 2003.
- [ZHA 99] Zhang Q., « Identification et Surveillance de systèmes Dynamiques », Habilitation à diriger les recherches, Université de Rennes1, Institut de Formation Supérieure en Informatique et en Communication, 1999.
- [ZHA 09] Zhang H. and Gu M., “Modeling job shop scheduling with batches and setup times by timed Petri nets,” *Mathematical and Computer Modelling*, Volume 49, Issues 1-2, pp. 286-294 January 2009.
- [ZWI 95] Zwingelstein G., « Diagnostic des défaillances - Théorie et pratique pour les systèmes industriels » *Traité des Nouvelles Technologies, série Diagnostic et Maintenance*, Hermès, Paris, 1995.

Résumé :

Le travail développé dans cette thèse s'inscrit dans le cadre d'une problématique générale portant sur les architectures de Surveillance-Commande des Systèmes Flexibles de Production Manufacturière (SFPM). En effet, l'architecture de Commande-Surveillance, que nous avons proposé, a l'objectif de minimiser l'appel à la maintenance corrective. Ceci est afin de minimiser les arrêts opérationnels du procédé. L'architecture développée « Surveillance Indirecte suite à une Commande Robuste » consiste à surveiller les paramètres de la loi de commande afin de maintenir la robustesse passive du système. Le suivi instantané des déviations dans un contexte de robustesse active est un moyen pour prévoir les événements inévitables qui peuvent influencer la robustesse du système. L'architecture développée est constituée de deux blocs : bloc commande et bloc surveillance.

La fonction commande robuste du bloc Commande ainsi que les fonctions pronostic et suivi-détection du bloc Surveillance sont les fonctions clés de cette architecture. En effet, pour la fonction commande robuste nous avons développé une loi de commande robuste hybride (robustesse passive et active) multicritères (flux et qualité) tout en se basant sur l'outil RdP à Intervalles. Pour la fonction pronostic, une nouvelle architecture « Réseau de Neurones Doublement Récurrent à Base de Fonctions Radiales » (RR²FR) dédiée à la prédiction est développée. Finalement, un modèle RdPTàO de suivi-détection est développé pour remplir les objectifs de suivi des paramètres qualitative et quantitative du produit et de détection des symptômes de défaillances.

A la fin de cette thèse, une application à un système industriel est présentée afin de confirmer l'apport de l'architecture « Surveillance Indirecte suite à une Commande Robuste ».

Mot clés: Architectures de Surveillance-Commande, SFPM, Commande robuste, Surveillance Indirecte, Pronostic, Modélisation, Réseaux de Petri, Réseau de Neurones RR²FR.

Abstract:

This PhD thesis deals with the problematic of Control-Supervisory architectures of Flexible Manufacturing Systems (FMS). Indeed, the proposed Control-Supervisory architecture has the objective to minimize the tasks of the corrective maintenance. The developed architecture 'Indirect Supervisory following a Robust Control' has the objective to monitor the parameters of the control law to maintain the passive robustness of the system. The instantaneous follow-up of deviations in the context of active robustness is a means to predict the inevitable events that may influence the robustness of the system. This architecture is approached in two principal blocks: Control block and Supervisory block.

The robust control function of the control block and the follow-up_detection and prognosis functions of the Supervisory block are the keys of this architecture.

Indeed, for the robust control function we developed a construction methodology of multicriteria (flow and quality) robust control law generating the margins of passive and active robustness. For the prognosis function, we introduced a novel neural architecture for time-series prediction: the 'Double Recurrent Radial Basis Function network' (R²RBF). For the follow-up_detection function, we proposed a new model of the detection deficiency symptoms through the follow-up of the temporal and non-temporal performance indicators.

At the end of this thesis, an application to an industrial system is presented to confirm the contribution of the proposed architecture: 'Indirect Supervisory following a Robust Control'.

Keywords : Control-Supervisory architectures, FMS, Robust control, Indirect Supervisory, Follow-up_detection, Prognosis, Modeling, Petri Nets, R²RBF Neural Network.
