UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité **: Automatique et Productique**

Arrêté ministériel : 7 août 2006

Présentée par

Mohamed Fathi KAROUI

Thèse dirigée par **Hassane ALLA** et **Abderrazak CHATTI**

Préparée au sein du **Laboratoire GIPSA-Lab**, département automatique dans **l'École Doctorale Électronique, Électrotechnique, Automatique, Traitement du Signal.**
En cotutelle avec **l'Institut National des Sciences Appliquées et de Technologie, « INSAT », Tunisie, Ecole Doctorale Informatique industrielle « II ».**

Surveillance des processus dynamiques événementiels

Date de soutenance de thèse **31 Octobre 2011**, devant le jury composé de :

M. Jean-Marc Thiriet Professeur, UJF (Grenoble)	Président
Mme Lilia El Amraoui Professeur, ESTI (Tunis)	Rapporteur
M. Eric Niel Professeur, INSA (Lyon)	Rapporteur
Mme Isabel Demongodin Professeur, U3 (Marseille)	Examinateur
M. Faouzi Ben Ammar Professeur, INSAT (Tunis)	Examinateur
M. Abderrazak Chatti Professeur, INSAT (Tunis)	Directeur de thèse
M. Hassane Alla Professeur, UJF (Grenoble)	Directeur de thèse

A mes Parents
A ma femme
A ma petite Sarra
A mon frère et
ma sœur

Remerciement

C’est avec un immense plaisir que je rédige cette rubrique de remerciement. Dans cette partie où sont évoqués les personnes qui m’ont aidé à faire et à achever ce projet que je considère comme le plus important de ma vie professionnelle.

Je tiens à exprimer ma reconnaissance profonde à mes directeurs de recherche : Professeur **Hassane ALLA** et Professeur **Abderrazak CHATTI** qui m’ont accordé leur confiance et m'ont soutenu sur de nombreux plans durant cette thèse. Je les remercie vraiment du fond du cœur, que ce soit pour leurs qualités scientifiques ou humaines. Je les remercie pour l’aide scientifique qu’ils m’ont toujours apportée, et surtout pour leur disponibilité, leur soutien et leurs encouragements.

Je tiens à remercier :

Madame **Lilia EL AMRAOUI**, Professeur et directrice de l’Ecole Supérieure de Technologie et d’Informatique ainsi que Monsieur **Eric NIEL** Professeur à l’Institut National des Sciences Appliquées de Lyon d’avoir accepté d’étudier mes travaux et d’en être les rapporteurs ainsi que pour l’intérêt et l’attention qu’ils ont accordés à cette étude.

Madame **Isabel DEMONGODIN**, Professeur à U3 de Marseille pour avoir accepté avec Monsieur **Faouzi BENAMMAR**, Professeur à l’Institut Nationale des Sciences Appliquées et de Technologie à Tunis d’examiner mes travaux.

Monsieur **Jean-Marc THIRIET**, Professeur à l’Université Joseph Fourrier de Grenoble de m’avoir fait l’honneur de présider ce jury.

Mes remerciements vont à tout le personnel du laboratoire Gipsa-lab à Grenoble et l’équipe de recherche RAMIS à l’INSAT qui m’ont accueilli durant ces années de thèse.

Je tiens tout particulièrement à remercier tous les chercheurs de l’equipe RAMIS (Yasser, Imen, Asma, Amira, Houda, et les autres . . .), pour l’ambiance sympathique qu’ils ont réussi à instaurer.

Je tiens aussi à remercier particulièrement mes deux amis du laboratoire Gipsa-lab Amine et Haithem qui m'ont aidé à m'intégrer au sein du laboratoire et surtout de leur aide lors de mes séjours à grenoble.

Je tiens à exprimer mon éternelle gratitude à mes parents qui m'ont toujours soutenu tout au long de mon cursus. Qu'ils trouvent dans l'achèvement de ce travail l'aboutissement de leurs efforts et de leurs sacrifices. J'exprime ma gratitude à ma femme Ons qui a toujours été ma motivation la plus importante et qui n'a cessé de m'encourager à achever ce travail. Je remercie mon frère Zied et ma sœur Imen pour tous leurs encouragements.

Je ne pourrai pas terminer sans exprimer un remerciement venant du plus profond du coeur à tous mes collègues de l'Ecole Supérieure de Technologie et d'Informatique qui m'ont toujours soutenu pendant mes années de thèses (Tarek, Nahla, Elyes, Younes, Mongi, Khaled et les autres . . .).

Table des matières

INTRODUCTION GENERALE 16

CHAPITRE 1 : LES SYSTEMES DYNAMIQUES HYBRIDES, PRESENTATION ET MODELISATION 21

1.1. Introduction 22

1.2. Présentation des systèmes dynamiques hybrides 22

1.2.1. Exemples de systèmes dynamiques hybrides 22

1.3. Structure globale des SDH 28

1.3.1. Le système physique 28

1.3.2. Propriétés de fonctionnement 29

1.4. Modélisation des SDH 32

1.4.1. Approche de modélisation des SDH 32

1.4.2. Outils de modélisation 33

1.5. Conclusion 36

CHAPITRE 2 : ETAT DE L'ART SUR LA SURVEILLANCE DES SYSTEMES DYNAMIQUES HYBRIDES 39

2.1. Introduction 40

2.2. Définitions et terminologie 40

2.3. Les systèmes commandés 42

2.3.1. Présentation des systèmes commandés 42

2.3.2. Mode de fonctionnement des systèmes commandés 44

2.3.3. Défaillances des systèmes commandés 46

2.4. Surveillance des systèmes dynamiques hybrides 47

2.4.1. Les différentes approches de la surveillance 47

2.4.2. Les méthodes de surveillance 50

2.4.3. Méthodes de surveillance à base de modèles.....52

2.5. Surveillance à base de modèle des SED.....61

2.5.1. Surveillance à base de modèle de comportement global 61

2.5.2. Surveillance à base de modèle de comportement de bon fonctionnement66

2.5.3. Surveillance des défauts interruptibles dans les SED 67

2.6. Conclusion.....72

Chapitre 3 : INTRODUCTION A LA SURVEILLANCE DES SYSTEMES DYNAMIQUES : APPROCHE DE LA SURVEILLANCE PAR AUTOMATE HYBRIDE LINEAIRE..... 75

3.1. Introduction 76

3.2. Modélisation par automates.....76

3.2.1. Les automates temporisés 76

3.2.2. Principe de calcul des successeurs d'une région.....78

3.2.3. Principe de calcul des prédécesseurs d'une région 82

3.2.4. Automate hybride linéaire.....84

3.3. Présentation de l'approche de surveillance 86

3.4. Construction du système de surveillance par automate hybride linéaire 88

3.4.1. Présentation intuitive de l'approche.....88

3.4.2. Surveillance du système par le temps enveloppe.....93

3.4.3. Démarche formelle de la construction du système de surveillance. 94

3.5. Conclusion.....103

CHAPITRE 4 : METHODOLOGIE DE SURVEILLANCE DES SYSTEMES DYNAMIQUES HYBRIDES..... 106

4.1. Introduction 107

4.2. Automate hybride rectangulaire 107

4.2.1. Syntaxe.....107

4.2.2. Sémantique 109
4.2.3. Analyse d'atteignabilité 110
4.3. Surveillance par automate hybride rectangulaire 111
4.3.1. Spécification de l'exemple 112
4.3.2. Analyse d'atteignabilité de l'exemple 115
4.4. Synthèse de la méthodologie de surveillance 119
4.4.1. Modélisation du processus 119
4.5. Exemple illustratif 1 122
4.5.1. Spécification de l'exemple 122
4.5.2. Construction du modèle de surveillance 123
4.5.3. Implémentation du système de surveillance 126
4.6. Exemple illustratif 2 128
4.7. Conclusion 130

CONCLUSION GENERALE 133

BIBLIOGRAPHIE 137

Table des figures

FIG. 1. 1 – EVOLUTION CONTINUE ET DISCRETE DU SDH 24
FIG. 1. 2 – A - BALLE EN REBONDISSEMENT, B- SYSTEME HYDRAULIQUE 24
FIG. 1. 3 – ARCHITECTURE GENERIQUE D'UN SYSTEME DE PRODUCTION 26
FIG. 1. 4 – PROCEDE BATCH 27
FIG. 1. 5 – PARTIE D'UN AUTOMATE HYBRIDE 30
FIG. 1. 6 – MODELE DU THERMOSTAT 31
FIG. 1. 7 – AUTOMATE HYBRIDE 34
FIG. 1. 8 – AUTOMATE HYBRIDE RECTANGULAIRE 36
FIG. 2. 1 – STRUCTURE GLOBALE D'UN SYSTEME COMMANDE 40
FIG. 2. 2 – CHRONOGRAMME D'UNE EVOLUTION DU SED 43
FIG. 2. 3 – CLASSIFICATION DES MODES DE FONCTIONNEMENT 44
FIG. 2. 4 – DUREE D'EXECUTION ET MODES DE FONCTIONNEMENT 45
FIG. 2. 5 – EXEMPLE DE SYSTEME INDUSTRIEL 46
FIG. 2. 6 – APPROCHE FILTRE 49
FIG. 2. 7 – APPROCHE COMPARATEUR 49
FIG. 2. 8 – APPROCHE PAR MODELE DE REFERENCE 50
FIG. 2. 9 – NIVEAUX DE CONNAISSANCE POUR LA SURVEILLANCE 53
FIG. 2. 10 – EXEMPLE D'UN AUTOMATE A ETATS FINIS 54
FIG. 2. 11 – AUTOMATE DE DYSFONCTIONNEMENT 56
FIG. 2. 12 – PRINCIPE DE SURVEILLANCE A BASE DE MODELE 57
FIG. 2. 13 – PRINCIPE DE L'APPROCHE DE SAMPATH 62
FIG. 2. 14 – EXEMPLE D'UN MODELE D'AUTOMATE A ETAT FINIS G ET SON DIAGNOSTIQUEUR G_D 64
FIG. 2. 15 – EVOLUTION DU SYSTEME 65
FIG. 2. 16 – SOLUTION POUR LA SURVEILLANCE DU SYSTEME 65
FIG. 2. 17 – MODELE TEMPOREL D'UN PROCEDE 66
FIG. 2. 18 – A- LE MODELE DU SYSTEME COMMANDE BASE SUR L'AUTOMATE A CHRONOMETRE. B- LA STRUCTURE PROPOSEE AFIN D'IMPLEMENTER LE SYSTEME DE SURVEILLANCE. 68

FIG. 2. 19 – A- LE SYSTEME MANUFACTURIER. B- LE MODELE GRAFCET REPRESENTANT LA COMMANDE DU SYSTEME CONSIDERE. 69
FIG. 2. 20 – A- L'AUTOMATE A_1 MODELISANT LA TACHE DU CONVOYEUR. B- L'AUTOMATE A_2 MODELISANT LA TACHE DU ROBOT. C- L'AUTOMATE A DU SYSTEME CONSIDERE. 70
FIG. 2. 21 – EXEMPLE DE SCENARIO D'EVOLUTION ET CHRONOGRAMMES CORRESPONDANTS 72
FIG. 3. 1 – AUTOMATE TEMPORISE 80
FIG. 3. 2 – SUCCESSEUR CONTINU D'UNE REGION : A – L'ESPACE Q_1, B – SUCCESSEUR (S_1, Q_1) 80
FIG. 3. 3 – AUTOMATE TEMPORISE 81
FIG. 3. 4 – SUCCESSEUR DISCRET D'UNE REGION : A- REGION Q_2, B- ESPACE SUCCESSEUR DISCRET DE LA REGION Q_2 82
FIG. 3. 5 – PREDECESSEUR CONTINU D'UNE REGION : A – ESPACE Q3, B – ESPACE PRE_T (Q_3) 84
FIG. 3. 6 – MODES DE FONCTIONNEMENT D'UN SYSTEME HYBRIDE 87
FIG. 3. 7 – ATELIER DE COLLAGE 89
FIG. 3. 8 – MODELE DE LA SURVEILLANCE DU CONVOYEUR 90
FIG 3. 9 – ESPACE DES ETATS ATTEIGNABLES 91
FIG. 3. 10 – (A) TRAJECTOIRE 1, (B) TRAJECTOIRE 2 92
FIG. 3. 11 – ESPACE ATTEIGNABLE LIMITANT LE COMPORTEMENT NORMAL D'UN SYSTEME CARACTERISE PAR UN ENSEMBLE DE CONTRAINTE TEMPORELLE 93
FIG. 3. 12 – SURVEILLANCE DE L'ATELIER DE COLLAGE PAR LE TEMPS ENVELOPPE : A- RDP TEMPOREL, B- PAR AUTOMATE TEMPORISE, C- ESPACE D'ETAT ATTEIGNABLE ET INSTANT DE DETECTION DE LA DEFAILLANCE. 94
FIG. 3. 13 – COMPORTEMENT DU PROCESSUS 95
FIG. 3. 14 – MODELE DE L'AUTOMATE REPRESENTANT LE SYSTEME DE SURVEILLANCE .. 97
FIG. 3. 15 – ESPACE D'ETAT CALCULE PAR L'ANALYSE EN AVANT (E'_2) 100
FIG. 3. 16 – ESPACE D'ETAT CALCULE PAR L'ANALYSE EN ARRIERE (E'_2) 102
FIG. 3. 17 – ESPACE DES ETATS DU PROCESSUS COMMANDE (E) 103
FIG.4. 1 – MODELE DE SURVEILLANCE D'UN SYSTEME REEL COMMANDE 112
FIG.4. 2 – MODELE DE SURVEILLANCE DU CONVOYEUR 113

FIG.4. 3 – MODELE DE SURVEILLANCE AHR ENRICHI 115
FIG.4. 4 – A – SUCCESSEUR CONTINU DE LA REGION *Z*, B – SUCCESSEUR DE LA REGION *Z* 116
FIG.4. 5 – A - ESPACE DE L'ANALYSE AVANT, B - EXEMPLE DE TRAJECTOIRES 117
FIG.4. 6 – MODELE DE SURVEILLANCE DU CONVOYEUR 117
FIG.4. 7 – A - ESPACE DE L'ANALYSE ARRIERE, B - ESPACE DES ETATS INITIAUX NON ACCESSIBLES. 118
FIG.4. 8 – A - ETATS NON ACCESSIBLES PAR L'ETAT INITIAL, B - ETATS NE CONDUISANT PAS VERS L'ETAT FINAL, C - ESPACE DU FONCTIONNEMENT NORMAL. 119
FIG.4. 9 – STRUCTURE GENERALE D'UN MODELE DE SURVEILLANCE 119
FIG.4. 10 – ESPACE D'ETAT PAR L'ANALYSE AVANT ET ESPACE D'ETAT DU FONCTIONNEMENT NORMAL 120
FIG.4. 11 – INTERBLOCAGE DE L'AHR DANS UN SOUS-ESPACE 121
FIG.4. 12 – ESPACE DES ETATS A PARTIR DE L'ANALYSE ARRIERE 121
FIG.4. 13 – MODELE *A*HR DU SYSTEME DE TRANSMISSION 125
FIG.4. 14 – EXEMPLE DE FONCTIONNEMENT : A - NORMALE, B - DYSFONCTIONNEMENT 127
FIG.4. 15 – ESPACE DES ETATS ATTEIGNABLES : A- PARAMETRE *X* EN FONCTION DE *H*, B- PARAMETRE *Y* EN FONCTION DE *H*, C- SUPERPOSITION DES DEUX ESPACES DU PARAMETRE *X* 127
FIG.4. 16 – ESPVACE DES ETATS ATTEIGNABLES : A- MODE 0, B- MODE 1, C- MODE 2, D- MODE 3 129
FIG.4. 17 – MODELE IMPLEMENTE DE L'AUTOMATE DE L'ATELIER DE COLLAGE 129
FIG.4. 18 – A - MODELE DE SURVEILLANCE DU MODE DE FONCTIONNEMENT 2, B - MODELE DE SURVEILLANCE IMPLEMENTE DU MODE 2. 130

INTRODUCTION GENERALE

L'objectif de toute industrie est d'accroître ses profits. L'automatisation des processus industriels est un moyen parmi d'autres pour atteindre cet objectif. En effet, en plus d'augmenter les taux de productivité, l'automatisation permet de minimiser les coûts de fabrications en réduisant la main d'œuvre. Cette tendance à l'automatisation des installations industrielles a donnée naissance à des systèmes réels de plus en plus complexes. De ce fait la mise en place d'un système de surveillance est devenue une nécessité, car il permet de signaler le plus tôt possible à l'opérateur les écarts détectés par rapport au comportement nominal prévu. C'est une action indispensable pour la mise en œuvre des actions préventives et correctives sur le système afin de d'éviter la propagation de pannes et de limiter leurs conséquences.

La surveillance est définie comme une action permettant la détection d'un défaut, la localisation de sa source et la détermination de ses causes. Nos travaux sont basés sur une méthodologie de surveillance comportant les deux aspects : l'acquisition des données et la détections des défauts. Son principe général consiste à confronter les données relevées au cours du fonctionnement réel du système avec la connaissance dont on dispose sur son fonctionnement normal et celui anormal.

Dans la littérature plusieurs méthodes de surveillances sont rencontrées : (Chow and willsky, 1984), (Zwingelstein, 1995), (Sampath et al, 1995), (Frank, 1996), (Sampath et al, 1996), (Lunz, 2000), (Tripakis, 2002), (Zad et al, 2005), (Ghazel et al, 2005), (Mezyani, 2005), (Allahham, 2008) … Ces méthodes s'appuient sur des modèles de fonctionnement normal ou défaillant du système. Plusieurs critères peuvent influer la méthode à choisir : l'évolution du système (discrète, continue ou hybride), l'implémentation du système de surveillance, la nature de l'information disponible et sa distribution. Les méthodes de surveillances des systèmes dynamiques hybrides sont généralement divisées en deux catégories : les méthodes sans modèle et les méthodes à base d'un modèle. Les méthodes sans modèles nécessitent l'historique du procédé et s'appuient sur des règles heuristiques ou sur des exemples d'exécution. Tandisque la deuxième catégorie se base sur une modélisation du comportement normal du système, une connaissance approfondie du système est nécessaire pour cette méthode de surveillance. Dans certains travaux trois types de fonctionnement sont définis pour un système : le fonctionnement normal, le fonctionnement défaillant et un fonctionnement dégradé qui est un état intermédiaire entre les deux premiers. On associe à ce troisième type un intervalle de temps supplémentaire qui se situe juste après l'intervalle de bon fonctionnement. L'intervalle de bon fonctionnement ainsi que celui du fonctionnement dégradé constituent ce qu'on appelle une enveloppe temporelle dans laquelle le système évolue. Les travaux existant permettent de détecter un retard ou l'absence d'un évènement à l'instant d'expiration de la date au plus tard. La détection du défaut dans cette enveloppe temporelle et non pas à ces extrémités est l'un des objectifs de nos travaux de recherches.

Contribution de la thèse

Notre contribution dans ce mémoire concerne la surveillance des systèmes dynamiques hybrides commandés ayant un caractère temporel. L'utilisation des données temporelles relatives à l'occurrence des évènements qui se produisent au cours du fonctionnement du processus est très utile pour un système de surveillance. Nous considérons ici des modèles de dynamique hybride où la dynamique de la partie continue est donnée par des équations différentielles où les dérivées des variables sont

constantes. Cela permet, en effet, d'augmenter la classe des systèmes étudiés. La surveillance consiste alors à garantir à tout instant un fonctionnement du système tout en respectant à la fois les contraintes physiques et les contraintes de spécification. Le fait de disposer des dynamiques qui vont au delà de l'horloge permet de modéliser de manière plus fine les contraintes physiques du système. Le comportement du système est alors présenté sur un modèle de l'automate hybride. Les défauts ne sont pas représentés dans le modèle de surveillance du système considéré. Le mécanisme développé dans nos travaux de recherche exploite d'une part les contraintes temporelles et physiques qui définissent le comportement normal du système. Et d'autre part, il utilise les techniques d'analyse d'atteignabilité de l'automate hybride. Dans une première étape nous modélisons le comportement du système par un automate hybride, tout en spécifiant ses différents modes de fonctionnement. Dans une seconde étape nous appliquons une procédure de synthèse à cet automate afin d'extraire uniquement les trajectoires qui vérifient les contraintes imposées au système. L'espace temporel synthétisé dans chaque sommet délimite le comportement normal du système. Tout écart de cet espace temporel dans un sommet déclenche une alarme signifiant que le système ne pourra plus réaliser ses objectifs dans les délais qui lui sont attribués tout en respectant les contraintes. La détection au plus tôt implique une détection d'une défaillance dans le système sans attendre l'expiration des délais temporels qui lui sont imposés.

Organisation du mémoire de thèse

Ce mémoire est organisé en quatre chapitres.

Le premier chapitre comporte une présentation des Systèmes Dynamiques Hybrides (SDH) incluant plusieurs exemples. Les systèmes hybrides ont la particularité de présenter deux aspects de fonctionnement à savoir, l'aspect continu et discret. Des approches de modélisation continue, discrète et mixte sont présentés dans ce chapitre.

Dans le deuxième chapitre nous présentons un état de l'art sur la problématique de la surveillance des systèmes dynamiques hybrides. En premier lieu, nous exposons les définitions et la terminologie de la surveillance, ensuite une présentation des systèmes commandés, puis les différentes approches et méthodes de surveillances présentes dans

la littérature sera faite. Deux méthodes de surveillance sont généralement utilisées ; la méthode sans modèle et la méthode à base de modèle. Nous nous intéresserons en particulier à la surveillance à base de modèle des systèmes à évènements discrets.

Dans le troisième chapitre, on définit les études d'atteignabilité sur les automates temporisés, permettant de calculer les espaces atteignables par les systèmes étudiés. Une présentation de l'approche de la surveillance est également faite ainsi que la construction du système de surveillance par automate hybride linéaire.

Le dernier chapitre est consacré à la généralisation de la méthodologie de surveillance élaborée par l'utilisation de l'automate hybride rectangulaire. Cette sous-classe d'automates nous permet de modéliser des systèmes plus complexes donc une modélisation hybride riche et permet aussi une analyse formelle. Cette partie sera ponctuée par l'implémentation du système de surveillance qui consiste à déterminer les relations caractérisant chaque sommet de l'automate qui modélise le système. Deux exemples illustratifs de notre démarche seront présentés.

Chapitre 1

LES SYSTEMES DYNAMIQUES HYBRIDES, PRESENTATION ET MODELISATION

1.1. Introduction

Les systèmes dynamiques hybrides sont des systèmes pour lesquels les dynamiques discrètes et continues interagissent. Cette interaction détermine le comportement du système. On peut trouver plusieurs types de systèmes hybrides : systèmes intrinsèquement hybrides, systèmes continus avec commandes discrètes, systèmes à événements discrets évoluant d'une manière continue, ou systèmes continus évoluant avec des commutations discrètes. La plupart des systèmes hybrides présents dans l'industrie sont intégrés dans des systèmes industriels commandés.

L'automatisation des systèmes industriels est la tendance actuelle de l'industrie moderne. Ce changement important est opéré en vue d'une amélioration de la qualité du produit ainsi que pour la diminution des coûts de fabrication. Pour réaliser ces objectifs, des systèmes de surveillances des équipements industriels sont utilisés, afin de garantir un fonctionnement optimal des systèmes de productions.

Ce chapitre comporte une présentation des systèmes dynamiques hybrides (SDH), ainsi que les différentes techniques utilisées pour les modéliser.

1.2. Présentation des systèmes dynamiques hybrides

Les systèmes hybrides sont des systèmes dynamiques faisant intervenir explicitement et en même temps des phénomènes ou des modèles de type dynamique continue et événementielle. Ces systèmes sont classiquement constitués de processus continus interagissant avec/ou supervisés par des processus discrets. [Zaytoon, 2001].

1.2.1. Exemples de systèmes dynamiques hybrides

Plusieurs systèmes technologiques ont un comportement hybride du fait qu'ils possèdent à la fois une dynamique discrète et une dynamique continue. Nous allons présenter plusieurs types de ces systèmes [Mezyani, 2005].

A. Systèmes continus pilotés par un contrôleur à événements discrets

Un système continu, commandé ou supervisé par un système à événements discrets, est appelé système hybride par la commande. Cette classe de systèmes hybrides est largement étudiée dans la littérature [Antsaklis et al, 1993] ; [Branicky et al, 1994] ; [Stiver et al, 1996].

Soit l'exemple d'un thermostat utilisé pour maintenir la température dans une chambre [Alur et al, 1995].

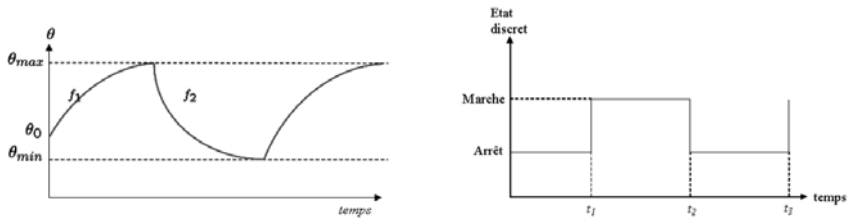
Le système est composé d'un élément chauffant et d'un capteur de température. Les seuils inférieurs et supérieurs du thermostat sont fixés à des valeurs θ_{min} et θ_{max} avec ($\theta_{min} < \theta_{max}$). la nature hybride du système est imposée par la commande discrète et l'évolution continue de la température.

- **la commande discrète.** Il s'agit d'une commande Tout Ou Rien (TOR). L'élément chauffant est en marche tant que la température ne dépasse pas θ_{max}. Le chauffage est arrêté quand la température descend en dessous du seuil θ_{min}. Les états discrets du système correspondent aux états « marche » et « arrêt ». L'évolution discrète correspond aux transitions de l'état « marche » à l'état « arrêt » et inversement.
- **L'évolution continue de la température.** La température de la chambre est une variable dont l'évolution est continue. L'évolution de la température peut être modélisée par les équations différentielles suivantes :

$$\dot{\theta} = \begin{cases} -\theta + \alpha & \text{Si le chauffage est en marche} \\ -\theta & \text{Si le chauffage est en arrêt} \end{cases}$$

Où $\alpha \in \mathbb{R}^+$ est une constante réelle positive.

Un exemple d'évolution est donné par la figure 1.1.a.

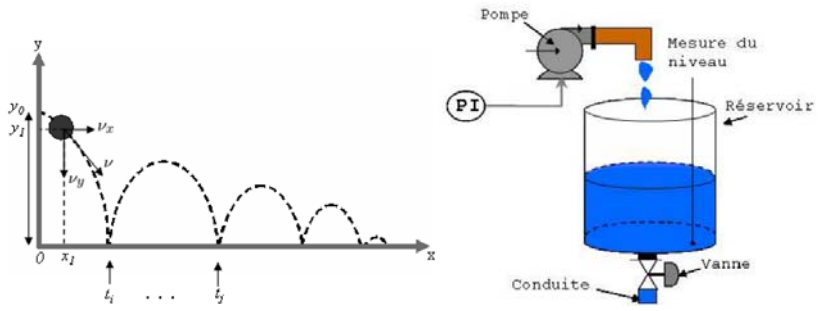


- a - Evolution de la température - b - Evolution de l'état discret

FIG. 1. 1 – Evolution continue et discrète du SDH

B. Systèmes continus comportant des discontinuités

Les phénomènes de discontinuités se produisent lorsque l'état passe instantanément de sa valeur courante à une autre valeur. Ce phénomène de commutation est illustré à travers l'exemple classique d'une balle en rebondissement [Branicky, 1995]. Considérons l'exemple d'une balle en chute libre (figure 1.2.a). Lors du contact avec le sol, la composante vitesse change de signe de façon instantanée, ce changement entraîne une évolution différente de la balle. Chaque état de la vitesse caractérise une dynamique différente.



- a - - b -

FIG. 1. 2 – a - Balle en rebondissement, b- Système hydraulique

C. Systèmes comportant des éléments discrets et continus

Certain systèmes sont constitués essentiellement d'éléments de type 'continu' et d'éléments de type 'discret'. On peut citer l'exemple des circuits électroniques qui contiennent à la fois des éléments à évolution continue (résistance, condensateur, …) et des éléments à états discrets (interrupteur, diode, thyristors, …). [Mezyani, 2005].

D. Systèmes continus pour lesquels des dynamiques discrètes sont introduites par abstraction

La modélisation de certains phénomènes physiques complexes requiert l'utilisation de fonctions non linéaires difficiles à manipuler. Plusieurs recherches proposent d'inclure des phénomènes discrets au sein de l'évolution continue afin de simplifier la modélisation [Mezyani, 2005].

Considérons le système hydraulique illustré par la figure 1.2.b Il est constitué d'un réservoir avec une conduite d'évacuation. Une vanne pneumatique placée sur la conduite d'évacuation et commandée en Tout ou Rien, permet de régler le débit de l'évacuation du liquide du réservoir. Une pompe permet d'alimenter le réservoir et une commande automatique permet de maintenir un niveau de liquide constant dans le réservoir.

Le niveau du liquide est une variable continue. Il dépend des débits entrant et sortant. Ces deux débits ont eux aussi des évolutions continues : le débit d'entrée est réglé par la commande et la valeur du débit de sortie est fonction de l'ouverture de la vanne pneumatique. L'ouverture de la vanne est une fonction continue dans le temps. Le débit sortant varie de 0% à 100% selon l'ouverture de la vanne. Un état discret est associé à la vanne. Cet état prend deux valeurs 0% (vanne fermée) et 100% (vanne ouverte). L'introduction de cet état discret met en évidence deux modes de fonctionnement du système et permet de simplifier la modélisation complète du système.

E. Systèmes discrets pour lesquels des dynamiques continues sont introduites par abstraction

Ces systèmes sont généralement des systèmes dont l'évolution de l'état discret est très rapide par rapport à la dynamique globale du système. [Gershwin and schitck,] ; [Alla and Coll] ;[Kurovszky, 2002].

Le système de production représenté par la figure 1.3 est composé de ressources correspondant à des stocks (S_1, S_2, S_2', …) et des groupes de machines (M_1, M_2, M_3, …). Chaque groupe est composé d'un nombre de machines identiques. Les stockes (S_1, S_2, S_2', …) sont utilisés pour emmagasiner les pièces jusqu'au moment ou une machine en aval est disponible pour commencer un nouveau traitement. Ce système de production peut être vu comme un système hybride ayant une évolution continue, représentant les flux de pièce dans le système et ayant une évolution discrète liée à l'état des ressources.

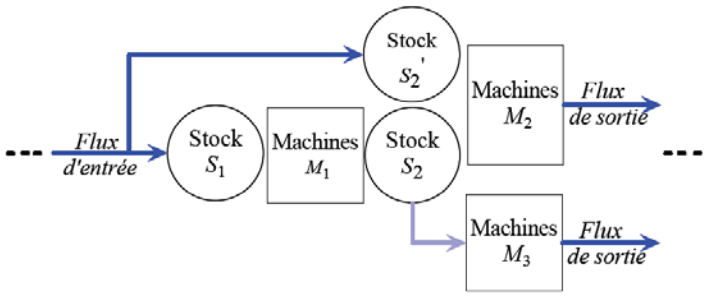


FIG. 1. 3 – Architecture générique d'un système de production

Le niveau des pièces dans les stocks peut être modélisé par une équation différentielle linéaire :

$$\dot{x}(t) = A.x(t) + B.u(t)$$

Où $x = [x_1, ..., x_i, ...x_n]^T$ représente le niveau de pièce dans les stocks. u est le débit de pièce en entrée. L'évolution discrète du système de production est décrite par l'occurrence des évènements associés à l'intervention de l'opérateur pour démarrer ou

arrêter le flux d'entrée et aux états des ressources. L'occurrence de l'un de ces événements entraîne le changement de l'état discret.

F. Procédé batch

Les systèmes Batch s'adressent aux industriels utilisant principalement des procédés discontinus. Les industries les plus sujettes à ces systèmes sont l'agro-alimentaire, la chimie, la pharmacie ou toute autre nécessitant des contrôles de mélanges et de dosages (ex : production de peintures, bains de trempage...). La production peut se faire en continu ou par traitements successifs (traitement par lots). Les procédés Batch sont constitués essentiellement de plusieurs sous-systèmes hybrides comme le montre la figure 1.4 [Manon, 2001].

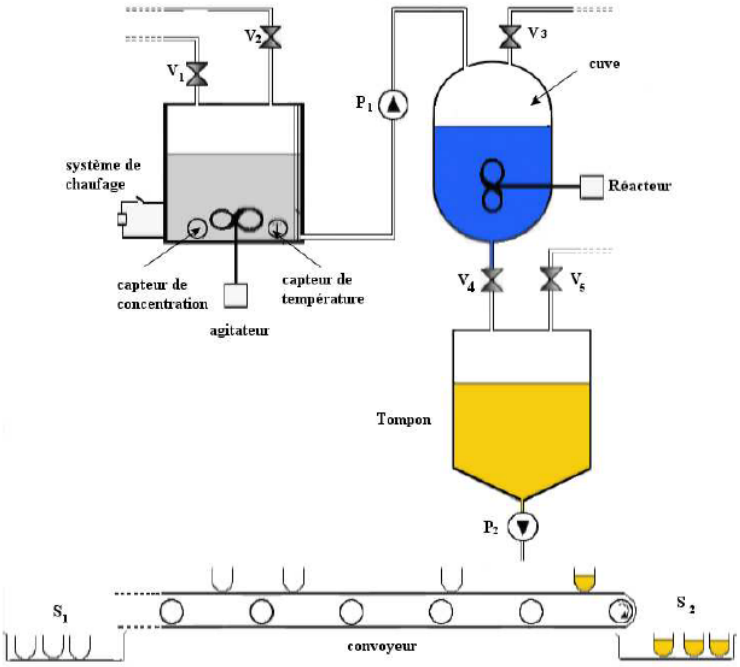


FIG. 1. 4 – Procédé Batch

L'objectif de ce procédé est de produire, à partir de deux produit '*A*' et '*B*' un produit chimique '*D*'. les produits *A* et *B* fournis à travers les vannes V_1 et V_2 au réservoir T_1 sont d'abord mélangé puis chauffés. Le mélange de *A* et *B* est alors transporté au

réservoir T_2 où il réagit avec un produit *C*. le produit résultant est ainsi transporté et conservé dans le réservoir T_3, attendant pour remplir les bouteilles vides qui sont ensuite transportées par le convoyeur servi par le stock S_1.

La nature hybride de ce système est bien illustrée dans cet exemple :

- Il contient des processus discrets (stocks S_1, S_2) et des processus continus, correspondant à l'évolution des niveaux, de la température et de la concentration des produits.
- Il est piloté par une commande discrète de type tout ou rien.
- Les capteurs sont de natures différentes : discrète et continue.

1.3. Structure globale des SDH

L'objectif de ce mémoire est la synthèse de la surveillance des systèmes dynamiques hybrides il est alors primordial de caractériser ces systèmes. Un système dynamique peut être décrit par le système physique et par ses propriétés de fonctionnement. Ces deux principes seront exposés par la suite.

1.3.1. Le système physique

A l'instar des systèmes dynamiques hybrides, le comportement dynamique du système évolue lui aussi avec deux aspects qui interagissent ; l'aspect continu et l'aspect discret [Kurovszky, 2002].

A. Aspects continus

L'évolution dynamique d'un système est déterminée généralement par les processus physiques qui ont lieu. Le modèle mathématique est acquis à partir des propriétés physiques du système permettant de trouver une représentation sous la forme d'une équation d'état.

Le modèle mathématique est du type :

$$\dot{x}(t) = f(x(t), u(t), t)$$

Avec *u(t)* le vecteur d'entrées et *x(t)* le vecteur des variables d'état. La relation précédente présente un modèle général dont l'analyse de ses propriétés doit se faire d'une façon continue dans le temps. Mais on peut trouver des modèles plus simples et plus pertinents, tout dépend de la manière de gestion de l'entrée de commande. Prenons l'exemple d'une entrée de commande linéaire par intervalle. La dynamique continue sera traduite par un ensemble d'équations différentielles simples.

Par exemple, dans le cas des procédés batch il faut définir des procédures globales contenant un nombre de phases du procédé. Chacune de ces phases détermine une évolution simple des variables observées. Dans ce contexte, les modèles décrivant le comportement du système seront de type :

$$\dot{x}(t) = v_i \qquad t_i \le t < t_{i+1}$$

Où v_i représente la variation de la variable x dans une partie de l'évolution du système et t_i indique les instants de passage entre les différentes évolutions.

B. Aspects discrets

L'occurrence des événements externes ou interne détermine l'évolution discrète d'un système dynamique hybride. Ces évènements peuvent être contrôlables ou non. L'occurrence des évènements implique le changement de la dynamique continue du système. Ces changements peuvent être déclanchés soit par la composition du système, soit par des entrées/sorties discrètes générées par différentes composantes du système.

1.3.2. Propriétés de fonctionnement

La description du système commandé est nécessaire dans le but de formuler la problématique de la synthèse de la commande et particulièrement du procédé de surveillance. Aussi il en est de même pour la description des propriétés de

fonctionnement. Ces propriétés forment une sorte de restriction dans l'évolution générale du système commandé.

Dans le cas de systèmes hybrides, les restrictions imposées par les propriétés de fonctionnement du système peuvent être décrites en les divisant en deux groupes : les propriétés relatives à la partie continue et ceux relatives à la partie discrète [Kurovszky, 2002].

A. Propriétés continues

Les propriétés continues correspondent aux limites imposées sur les variables d'états continus *x(.)*. Ces propriétés sont exprimées par des conditions logiques qui restreignent l'évolution des variables d'états dans des régions différentes selon leurs valeurs. Ces régions seront divisées en des régions de fonctionnement désirables et des régions de fonctionnement non désirables. Dans le cas de l'utilisation des automates hybrides les régions non désirables sont appelées sommets interdits.

Considérons la partie de l'automate hybride présentée dans la figure 1.5

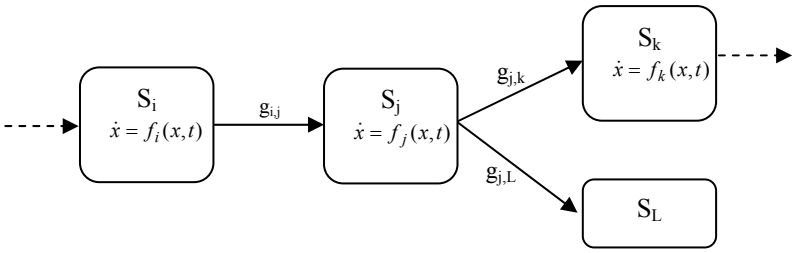


FIG. 1. 5 – Partie d'un automate hybride

L'évolution dynamique de l'automate hybride a lieu par une alternance de pas discrets et continus. Ainsi l'évolution continue a lieu dans les sommets de l'automate tandis qu'une évolution discrète est réalisée par le passage d'un sommet à un autre. Le sommet interdit S_L est atteint depuis le sommet S_j par la validation de la transition étiquetée par $g_{l,j}$. Ainsi la garde de la transition $T_{j,l}$ représentera la région interdite.

Exemple 1.1. Pour illustrer la notion de sommet interdit, reprenant l'exemple du thermostat dont le fonctionnement a été détaillé dans la section 1.2.1.A. le modèle de l'automate modélisant son fonctionnement est illustré par la figure 1.6

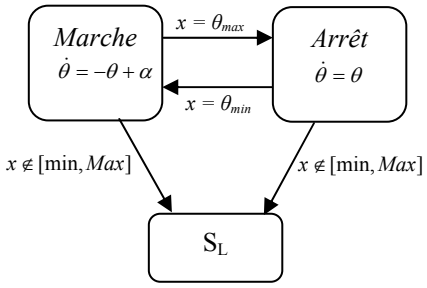


FIG. 1. 6 – Modèle du thermostat

La température représente ici la variable d'état continue du système. Les propriétés d'évolution du système imposent des contraintes sur la température et ceci quelque soit l'état du système en marche ou en arrêt.

Les conditions d'évolutions imposées aux variables d'état continues sont des contraintes globales du système. Notre approche de la surveillance des systèmes dynamiques hybrides repose essentiellement sur ces contraintes appliquées au système.

B. Propriétés discrètes

La partie discrète d'un processus peut être vue comme une machine à états finis. En général, les propriétés discrètes sont données sous la forme de conditions logiques décrivant l'ordre d'occurrence des événements dans le système pendant son fonctionnement.

L'outil de modélisation permettant de prendre en compte tous les aspects présentés dans une même structure est l'automate hybride. Une présentation détaillée de cet outil est faite dans la section suivante.

1.4. Modélisation des SDH

On trouve plusieurs outils de modélisation des systèmes dynamiques hybrides dans la littérature. Parmi ces outils on peut citer les automates hybrides [Alur et al., 1993], les automates hybrides rectangulaires [Henzinger et al., 1998], les automates hybrides linéaire [Müller et Stauner, 2000], les réseaux de Petri hybrides [Alla et David, 1998 ; David et Alla, 2004], les Statecharts hybrides [Harel et Pnueli, 1987], les bond graphes hybrides [Mosterman, 1997], … . Nous allons présenter quelques outils parmi ceux qui ont été cités, et que nous avons trouvé aptes à résoudre notre problématique.

1.4.1. Approche de modélisation des SDH

L'existence de plusieurs outils de modélisation des SDH traduit la difficulté du choix adéquat de la méthode de modélisation. Le principal critère de sélection est lié à la problématique considérée. Un système peut être décrit par un modèle continu, discret ou hybride selon les objectifs qu'on veut atteindre. Généralement un SDH est modélisé par un ensemble de systèmes regroupant à la fois des dynamiques continues et des systèmes à événements discrets [Mezyani, 2005]. Nous pouvons classer les approches de modélisation des SDH en trois classes : l'approche continue ; l'approche discrète et l'approche mixte.

A. Approche continue

Le principe de cette approche consiste soit à éliminer la composante discrète du système, soit à la transformer en des équations différentielles [Kurovszky, 2002]. De ce fait le système hybride est présenté comme un système ne comportant que des équations algébriques différentielles linéaires ou non linéaires. L'avantage de cette approche est que l'on revient à es méthodes classiques d'analyses des systèmes continus linéaires ou non linéaires. L'inconvénient majeur de cette modélisation réside dans le fait de ne pas représenter explicitement l'évolution discrète pour l'utilisateur, autre inconvénient non négligeable est la complexité des équations obtenues.

B. Approche discrète

L'approche discrète consiste à remplacer la dynamique continue du système hybride par une évolution discrète. Les travaux de Puri présentent une méthode directe afin d'obtenir un modèle événementiel du système hybride qui consiste à découper l'espace d'état continu en plusieurs régions, associées chacune à un état discret [Puri et al., 1996]. Toutefois, ce concept de modélisation reste confronté au compromis entre la précision et le nombre d'états discrets rapidement explosif.

C. Approche mixte

L'approche mixte consiste à regrouper les aspects continus et discrets dans le même formalisme de modélisation. Cette approche a engendré l'apparition de nouveaux modèles hybrides à partir des models continus et discrets. Nous citons comme exemple les réseaux de Petri hybrides et les automates hybrides obtenues à partir des modèles discrets et les modèles MLD (*Mixed Logical Dynamical*) obtenue par des modèles continus [Bemporad and Moriari, 1999].

L'idée de base de cette dernière méthode est d'introduire des variables auxiliaires qui permettent de modéliser les relations existantes entre les parties continues et discrètes. Ainsi le passage à la partie discrète nécessite l'ajout de variables logiques. Pour la partie correspondant au passage discret/continu, des variables auxiliaires sont ajoutées.

1.4.2. Outils de modélisation

Nous présentons dans la suite quelques approches mixtes rencontrées dans la littérature.

A. Les automates hybrides

Les automates hybrides [Alur et al., 1993] sont une forme étendue d'automate à état fini. Ils modélisent des systèmes regroupant deux composantes : celle ayant un comportement continu modélisé par un système d'équations algébriques ou

différentielles et celle ayant un comportement discret, modélisée par un automate à état fini. L'évolution d'un automate hybride est caractérisée par un comportement continu dans lequel les variables d'état et le temps évoluent de façon continue, elle est caractérisée aussi par un comportement discret dans le quel plusieurs transitions discrètes et instantanées peuvent être franchies. Ainsi l'automate hybride est assimilé à un automate à état finis dirigeant un ensemble d'équations différentielles. L'automate change d'état si un événement discret se produit ou si les variables d'états valident les conditions logiques qui leurs sont associées [Henzinger, 1996].

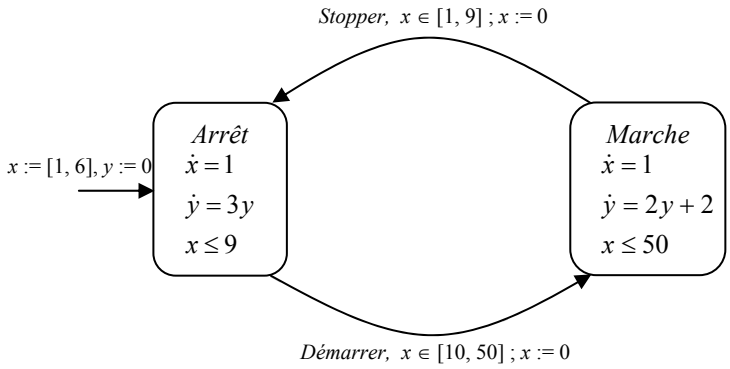


FIG. 1. 7 – Automate hybride

Soit l'automate illustré par la figure 1.7 modélisant un système hybride. Cette modélisation représente l'évolution continue par des équations différentielles qui sont associées aux sommets du graphe, l'évolution discrète quand à elle est représentée par les arcs étiquetés du graphe. Les sommets « Arrêt » et « Marche » représentent les états discrets du système où l'évolution continue a lieu. Les prédicats $x \in [1, 9]$ et $x \in [10, 50]$ sur les arcs traduisent les conditions, dites aussi gardes, pour l'occurrence respective des événements *Stopper* et *Démarrer*. Les prédicats $x \leq 9$ et $x \leq 50$ dans les sommets représentent les invariants de l'automate, c'est-à-dire des conditions imposées aux variables continues du système pour rester dans un sommet discret. L'initialisation du système est représentée par l'arc d'entrée dans le sommet « Arrêt ». L'étiquette de cet arc $x := [1, 6]$ et y := 0 représente l'état initial des variables continues et correspond à une affectation des variables, c'est à partir de cette région que la dynamique du

système hybride démarre. Les variable x et y évoluent dans le sommet « Arrêt » selon les équations différentielles $\dot{x} = 1$ et $\dot{y} = 3y$, et de même dans le sommet « Marche » avec les équations différentielles $\dot{x} = 1$ et $\dot{y} = 2y + 2$.

L'étude du comportement et la vérification ou contrôle des systèmes par l'intermédiaire des automates hybrides occupent une place importante dans l'étude des SDH. Cela consiste à vérifier des propriétés qualitatives et quantitatives sur un système en utilisant des méthodes relevant de l'automatique ou de l'informatique. Le contrôle se fait d'abord par une modélisation du SDH par un automate hybride. Ensuite par une étude du comportement des propriétés qualitatives à travers des techniques de model-checking [Henzinger et al., 1997], ou les propriétés quantitatives à travers l'analyse d'accessibilité [Alur et al., 1993, 1995].

L'analyse d'accessibilité ou d'atteignabilité consiste à trouver l'espace d'état atteignable par l'évolution du système hybride étudié. Ce problème n'est pas décidable pour un automate hybride sans hypothèse particulière [Henzinger et al., 1998]. Il faut alors apporter des restrictions pour avoir des sous-classes pour lesquelles certaines propriétés sont décidables [Derbel, 2009].

B. Les sous-classes d'automates hybrides

Il existe plusieurs sous-classes du modèle automate hybride dans la littérature. Ces sous-classes ont été introduites afin d'alléger la structure du modèle initial et ainsi simplifier son étude comportementale et sa vérification ou son contrôle. Parmi ces modèles, on peut citer :

- Les automates hybrides linéaires [Alur et al., 1993] : un automate hybride est dit linéaire si les conditions de flux, des invariants, des gardes, sont définies par des expressions linéaires sur l'ensemble des variables.
- Les automates hybrides rectangulaires [Kopke, 1996] [henzinger et al., 1998] : c'est une sous-classe des automates hybrides linéaires. La condition de flux dans ce modèle est définie sous la forme de prédicats rectangulaires de la

forme $\dot{x} \in [a,b]$, pour chaque variable x du modèle. De même, les invariants, les gardes, la condition initiale sont décrites par des prédicats rectangulaires. Dans la figure 1.8 nous illustrons l'exemple d'un automate hybride rectangulaire.

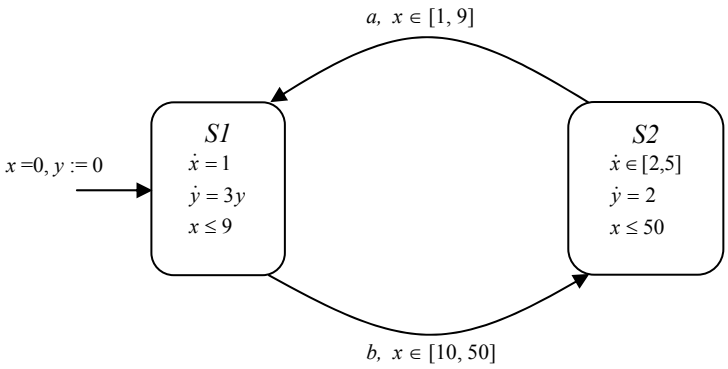


FIG. 1. 8 – Automate hybride rectangulaire

- Les automates hybrides rectangulaire initialisés [Henzinger et al., 1998] sont une sous-classe des automates hybrides rectangulaires. Dans ce modèle, chaque variable qui change de condition de flux, suite au franchissement d'une transition entre deux sommets doit être réinitialisée.

Les sous-classes des automates hybrides linéaires et rectangulaires seront présentées dans les chapitres suivants en détail car ils représentent les modèles de base de notre travail de recherche.

1.5. Conclusion

Dans ce chapitre nous avons présenté plusieurs exemples de systèmes dynamiques hybrides avec les interactions possibles entre les systèmes continus et les systèmes discrets qui les composent.

La structure globale des SDH a été présentée, cette structure comprenant le système physique et les propriétés de fonctionnement à la fois continues et discrètes.

Plusieurs approches ont été introduites en vue de la modélisation, Seule une brève présentation des automates hybrides et de ses sous-classes est faite. En effet, cet outil constitue notre outil principal de modélisation pour la surveillance des systèmes dynamiques hybrides. Il sera décrit en détail dans le chapitre 3.

Chapitre 2

ETAT DE L'ART SUR LA SURVEILLANCE DES SYSTEMES DYNAMIQUES HYBRIDES

2.1. Introduction

Dans tous les systèmes commandés il y a une partie opérative et une partie commande comme il est illustré par la figure 2.1. La partie commande a pour rôle de traduire les consignes venant de l'opérateur en ordres, qui seront transmises à la partie opérative pour exécution. L'état du système est connu grâce à la mise en place de plusieurs capteurs traçant son évolution.

Une exécution correcte de tous les ordres transmis par la partie commande définit un ensemble d'états du système que nous appelons comportement normal. La fonction de surveillance permet d'observer tous les comportements possibles du système et de les comparer en temps réel au comportement normal.

Dans ce chapitre nous allons présenter dans un premier temps les définitions et terminologie qui se rapportent à la surveillance et au diagnostic de défaut. Ensuite nous présenterons les systèmes commandés continus, discrets et hybrides ainsi que les types de défaillances qui les caractérisent. Par la suite on s'intéressera en particulier à la surveillance des systèmes dynamiques hybrides où on explicitera les différentes approches de surveillance ainsi que les méthodes de surveillance utilisées pour ces systèmes. A la fin de ce chapitre on présentera une étude sur la surveillance des défauts interruptibles des systèmes à évènements discrets.

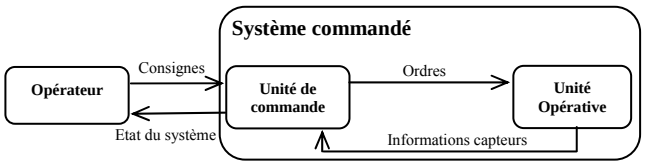


FIG. 2. 1 – Structure globale d'un système commandé

2.2. Définitions et terminologie

Les communautés discrètes et continues ont eu des divergences sur les définitions et la terminologie de la surveillance et du diagnostic des défauts. Dans le paragraphe suivant nous citons les définitions de quelques mots clés de la surveillance des deux

communautés. Pour la terminologie des systèmes à événements discrets on mentionne les travaux de (Rouchon, 1992)) et (Combacau et al., 2000). Concernant les systèmes continus nous mentionnons les travaux de (Iserman et Ballé, 1997).

- Un **défaut** est une déviation par rapport aux conditions acceptables ou normales d'un paramètre caractéristique du système. Cette déviation entraîne au moins le non accomplissement d'une propriété.
- La **défaillance** ou **panne** est une anomalie fonctionnelle dans le système. Dans ces conditions le système est dans l'incapacité à remplir sa fonction selon les conditions de fonctionnement spécifiées au départ.
- La **surveillance** consiste à émettre une alarme en se basant sur les informations délivrées par les capteurs. Elle reconstitue l'état réel du système à partir des signaux provenant du système physique et de l'unité de commande. La fonction surveillance est restreinte à la récolte d'information sans pour autant intervenir sur le procédé ni sur l'unité de commande.
- La **détection** consiste à comparer le comportement actuel du système physique au comportement prévu initialement et ensuite prendre une décision en référence à la comparaison. La fonction de détection alerte les opérateurs de supervision en cas d'écart par rapport au comportement normal. Grâce à cette fonction le comportement du système peut être caractérisé soit en fonctionnement normal soit en fonctionnement anormal. La qualité de la détection est établie en comptabilisant les taux de fausses alarmes et de non détection.
- La **localisation** permet de déterminer le ou les éléments défaillants.
- **L'identification** estime les caractéristiques statiques et dynamiques de ce défaut : l'instant d'apparition de la panne, sa durée et son importance.
- Le **Diagnostic** permet de déterminer le type, la taille, l'endroit et l'instant d'apparition du défaut. L'opération de diagnostic inclue la localisation et l'identification d'un défaut (Iserman et Ballé, 1997).
- La **Supervision** est une macro-fonction regroupant des taches de commande et de surveillance. La supervision doit piloter l'exécution de la séquence d'opération et assurer la gestion et la commande en temps réel des ressources nécessaires à cette

exécution (Dubois et Gentil, 1990), et ceci quel que soit le fonctionnement du système normal ou avec présence de défaillances :

- En fonctionnement normal elle doit surveiller et contrôler le déroulement des opérations.
- En présence d'une défaillance, la supervision doit prendre les décisions nécessaires pour assurer un retour vers le comportement normal (Combacau et al., 2000).

2.3. Les systèmes commandés

2.3.1. Présentation des systèmes commandés

Par définition, un système est un ensemble de composants oeuvrant dans un milieu avec lequel ils interagissent afin d'exécuter la tache qui lui est attribué. Les systèmes commandés peuvent êtres caractérisés par leur dynamique traduisant leurs évolutions. Il existe trois abstractions qui peuvent modéliser cette dynamique d'évolution : les systèmes continus, les systèmes à événements discrets et les systèmes dynamiques hybrides (Dousson, 2007). Ces systèmes ont déjà été présentés dans le chapitre 1, nous les reprenons brièvement ci-dessous.

A. Les systèmes continus

Les systèmes continus comportent des éléments quantifiés par une ou plusieurs mesures qui évoluent avec le temps. On peut citer comme grandeurs ; une accélération, une vitesse, une position une pression, un niveau, une température, un débit, etc. Des outils mathématiques sont utilisés pour gérer ce type de système. Grâce à ces outils on représente la dynamique continue par des équations différentielles. On associe les systèmes dynamiques à une représentation regroupant des variables d'états continues et une variable temporelle, discrète ou continue.

B. Les systèmes à évènement discrets

Un système à évènement discret (SED) est un système comportant plusieurs états discrets. Le passage d'un état à un autre se fait suite à l'occurrence d'un évènement discret asynchrone (Cassandras et al., 1999). Plusieurs cas peuvent être considérés comme des SED allant de la gestion d'un dîner (problème des philosophes), (David et Alla, 1995) au fonctionnement de machines dans une chaîne de fabrication flexible.

Exemple 2.3.1.B

Nous considérons l'exemple d'une machine industrielle. Ce système peut être traité comme un SED, en nous plaçant à un niveau d'abstraction particulier. On suppose que ce système admet trois états possibles : **arrêt**, **marche** et **panne**. L'évolution de ce système d'un état à un autre est réalisée par l'occurrence des évènements suivants : ***début du travail*** (*d*), ***fin du travail*** (*f*), ***panne*** (*p*), ***réparation*** (*r*). La figure 2.2 présente un comportement possible de ce SED.

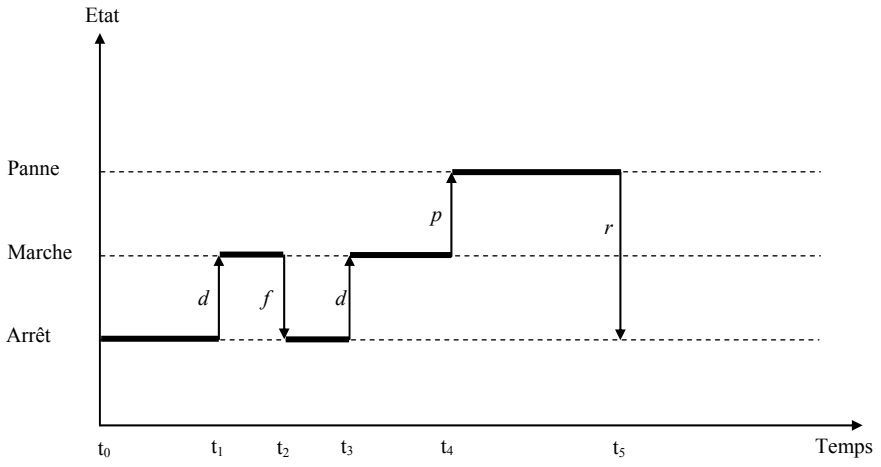


FIG. 2. 2 – Chronogramme d'une évolution du SED

Initialement le système est à l'état **arrêt** (instant t_0), l'occurrence de l'évènement *d* à l'instant t_1 bascule le système vers l'état **marche**. De la même manière, le SED passe aux états **arrêt**, **marche** et **panne** suites aux occurrences respectives des évènements *d, f, p et r*. Ainsi l'évolution d'un SED peut être représentée par un ensemble de couples : (δ, t) où δ représente un évènement et t représente l'instant où cet évènement s'est

produit. Notre exemple est défini par la séquence d'événement suivante : (d, t_1), (f, t_2), (d, t_3), (p, t_4), (r, t_5), …

Cet ensemble ordonné de couples représente une trajectoire (mot ou trace) du système. L'information temporelle est représentée dans cette description d'une manière explicite, ainsi cette trajectoire est dite temporisée. En faisant abstraction du temps l'évolution du système sera décrite par l'ordre d'occurrence des évènements. Cette description du SED est dite logique. Dans l'exemple précédent, la trajectoire *d f d p r*, décrit une évolution logique du SED.

La modélisation de ces systèmes pour la commande et l'analyse est réalisée avec plusieurs outils mathématiques tels que les réseaux de Petri et les automates à état finis.

C. Les systèmes dynamiques hybrides

Les systèmes dynamiques hybrides (SDH) regroupent en même temps les deux aspects discret et continu. Ces systèmes sont caractérisés par une évolution temporelle des variables continus et des variables discrètes (Zaytoon, 2001). Ces systèmes seront explicités en détail dans le chapitre suivant.

2.3.2. Mode de fonctionnement des systèmes commandés

Dans la littérature on trouve souvent une classification par les modes de fonctionnement. La nature du mode de fonctionnement du système commandé implique que la tâche qui lui est attribuée peut être exécutée totalement, partiellement ou non exécutée. Il existe plusieurs modes de fonctionnement (Rayhane, 2004) qui sont illustrés par la figure 2.3 :

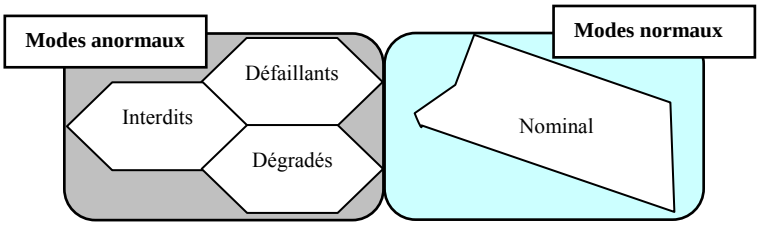


FIG. 2. 3 – Classification des modes de fonctionnement

- les modes de fonctionnement **normaux** : Ils comprennent tous les modes pouvant amener le système à exécuter sa tâche y compris le mode nominal qui permet d'exécuter parfaitement la tâche.
- Les modes de fonctionnement **anormaux** : Dans ces modes le système ne peut exécuter sa tâche complètement ou même ne pas l'exécuter totalement. On peut les décomposer en :
  - Modes critiques : dans ces modes le système fonctionne d'une façon particulière et souvent non souhaitée.
  - Modes dégradés : dans ces modes le système réalise partiellement ses objectifs.
  - Modes défaillants : c'est un mauvais fonctionnement du système.
  - Modes interdits : le système ne doit pas fonctionner dans ces modes pour des raisons de sécurité.

Rayhane a introduit par ses travaux la notion de comportement dégradé. Chaque système est composé de plusieurs tâches qui interagissent. Le système est en fonctionnement normal si le temps d'exécution de la tâche est dans un intervalle noté I_m, il est en mode dégradé si la durée de l'exécution de la tâche dépasse l'intervalle I_m mais reste contenue dans l'intervalle J_m. Trois valeurs de temps sont définis pour chaque tâche; $T^m_{\min}$, $T^m_{\max}$ et T^m_c. Pour une exécution normale de la tâche sa durée d'exécution est comprise dans l'intervalle de fonctionnement normal $I_m = [T^m_{\min}, T^m_{\max}]$.

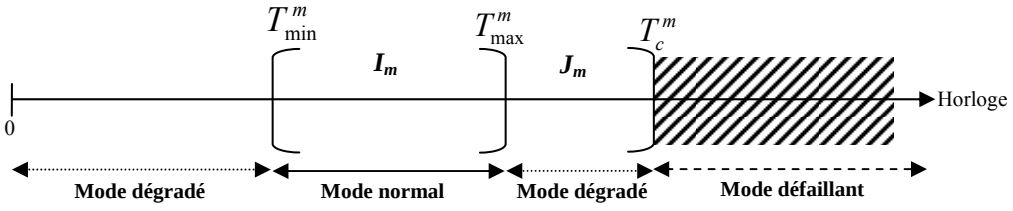


FIG. 2. 4 – Durée d'exécution et modes de fonctionnement

L'intervalle $J_m = [T^m_{\min}, T^m_c]$ correspond à un intervalle de tolérance dans lequel le système est en mode dégradé. Cet intervalle comptabilise les retards qui peuvent être dus au vieillissement des machines, aux surcharges, aux problèmes mécaniques, etc. Si

la durée d'exécution dépasse la limite T_c^m, le système est considéré comme défaillant (FIG. 2.4).

2.3.3. Défaillances des systèmes commandés

Après le lancement d'une application, il n'y a aucune assurance que le comportement du système réponde aux objectifs prédéfinis. Ce dysfonctionnement peut être à l'origine de plusieurs causes : apparition d'un défaut, une mauvaise décision de la partie commande, une fausse information du capteur. Nous allons par la suite présenter les différents types de défauts possibles dans un système commandé pour connaître le comportement du système suite à l'occurrence d'un défaut (Huang et al., 1996).

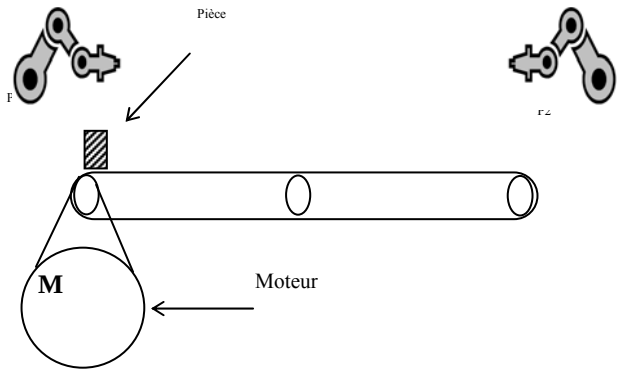


FIG. 2. 5 – Exemple de système industriel

A. Défauts intermittents et permanents

- Les défauts permanents apparaissent lorsqu'il y a un mauvais fonctionnement d'un élément qui doit être réparé ou remplacé.
- Les défauts intermittents apparaissent si un ou plusieurs éléments cessent de fonctionner temporairement, la reprise de fonctionnement de ces composants se fait spontanément.

La différence entre ces deux défauts est que le défaut permanent bloque le système jusqu'à une intervention extérieure, alors que le défaut intermittent permet au système d'osciller entre deux modes de fonctionnement : mode normal et mode anormal.

B. Défauts des actionneurs et des capteurs

Les défauts qui apparaissent sur les actionneurs et les capteurs, touchent en général leur partie mécanique, électromagnétique ou électrique. Ces défauts bloquent la sortie du capteur à niveau haut ou bas, ils figent également les actionneurs sur une position particulière active ou inactive. L'exemple de la figure 2.5 illustre un atelier de collage qui comprend deux postes ainsi qu'un convoyeur amenant les pièces d'un poste à un autre. Le poste 2 récupère la pièce si elle atteint l'extrémité du convoyeur, cette position est détectée par un capteur placé au bout du convoyeur. Ce capteur peut être affecté par un défaut de blocage qui maintient son signal de sortie à un niveau bas. Dans ce cas le système de commande ne donnera pas l'ordre au poste 2 pour la récupération de la pièce. Si le système de convoyage n'est pas muni d'une butée de fin de course, il y a risque d'endommagement ou de destruction de la pièce.

Ce même système peut avoir un défaut affectant un actionneur, le moteur M peut subir une accélération ou un ralentissement. La détection de ce défaut peut être réalisée par une vérification des contraintes temporelle de transport de la pièce.

C. Défauts affectant la structure du système

Tout le système ainsi que ces éléments peuvent subir des défauts structurels ou comportementaux. On peut citer comme exemple une perturbation de l'alimentation électrique ou bien une détérioration physique d'un des éléments du système, si l'atelier illustré par la figure 2.5 est muni d'un convoyeur à bande, il y a risque d'usure de cette bande.

2.4. Surveillance des systèmes dynamiques hybrides

2.4.1. Les différentes approches de la surveillance

On peut distinguer trois approches différentes pour l'implémentation du système de surveillance (Rayhan, 2004).

La première est une surveillance intégrée à la commande. Elle nécessite une connaissance préalable des états anormaux. Tous ces états doivent êtres intégrés dans l'unité de commande.

La deuxième approche consiste à séparer le système de surveillance et l'unité de commande. Cette approche a l'avantage d'alléger les instructions sur l'unité de commande et d'affranchir le système de surveillance dans le choix de la technique de surveillance. Par contre il y a la possibilité d'apparition de conflits entre la surveillance et la commande qui ont accès tout les deux à l'unité opérative.

La troisième approche est une combinaison des deux premières approches ; il y a une séparation entre les fonctions de diagnostic et de décision, par contre il y a une intégration à la commande des fonctions de détection et de reprise. Selon les ordres l'unité de commande définit le comportement normal et tout écart de cette évolution est considéré comme comportement anormal. L'intérêt de cette approche réside dans le fait que le comportement normal est caractérisé dés la spécification du modèle de la commande (Combacau, 1998).

Nous présentons par la suite les différentes approches présentes dans la littérature.

A. L'approche filtre

Le concept de cette approche est d'insérer un ou plusieurs filtres entre l'unité de commande et l'unité opérative comme l'illustre la figure 2.6. Dans cette configuration, l'exécution de la commande n'est autorisée que si l'état réel du système est cohérent avec cette dernière. Les informations instantanées émises par le capteur déterminent l'état réel du système. Le filtre est composé d'un filtre de commande et d'un filtre de valeurs capteur. Le rôle du premier est de tester la cohérence de l'instruction par rapport à l'état du procédé, le rôle du deuxième consiste à comparer les signaux transmis par les capteurs avec ceux correspondant au comportement normal du procédé (Nourelfath, 1997).

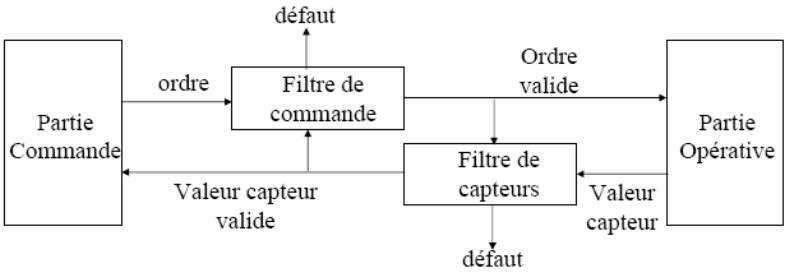


FIG. 2. 6 – Approche filtre

B. Approche comparateur

Cette approche repose sur la comparaison permanente de l'état réel du système déterminé à partir des informations des capteurs et de celui donné par le modèle de comportement du système (figure 2.7). Tout écart entre l'état réel du système et celui donné par le modèle signale une défaillance. Les travaux de (Holloway et Krogh, 1991) placent le modèle du procédé en tant que émulateur des évolutions normales de l'unité opérative. Son rôle est de calculer les fenêtres temporelles d'occurrence des comptes rendus émis par le procédé quand celui-ci est soumis à une commande particulière. Pour une consigne donnée, un bloc de comparaison permet de vérifier si un compte rendu émis par le procédé arrive bien à la date prévue par le modèle (Rayhane, 2004).

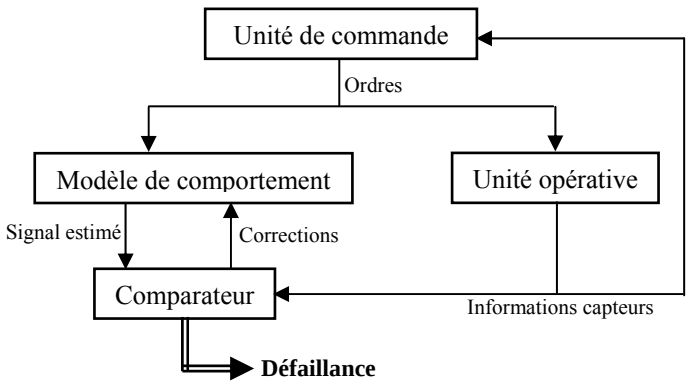


FIG. 2. 7 – Approche comparateur

C. Approche de modèle de référence

Cette approche nécessite l'existence d'un modèle de référence contenant tous les modèles de comportement normaux du système comme l'illustre la figure 2.8. Avant

l'envoi d'une instruction par l'unité de commande, celle-ci consulte le modèle de référence et s'il y n'a pas de concordance entre l'état du système et la nouvelle instruction alors une erreur de l'unité de commande est détectée. Le modèle de référence et l'unité opérative doivent évoluer simultanément, si il y a un décalage entre les deux modèles il y aura toujours une défaillance. Cette approche à l'avantage de vérifier l'état du procédé avant l'exécution d'une nouvelle instruction.

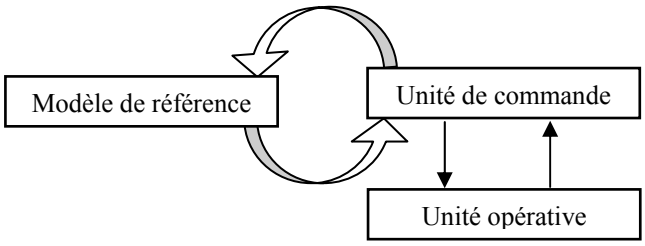


FIG. 2. 8 – Approche par modèle de référence

2.4.2. Les méthodes de surveillance

On peut différencier les méthodes de surveillance par plusieurs critères : l'évolution de la dynamique du système (continu, discret ou hybride), la mise en place du système de surveillance (en ligne ou hors ligne), la nature de l'information (quantitative ou qualitative) et sa distribution (décentralisée, centralisée) (Derbel, 2009). Ces méthodes sont généralement divisées en deux catégories (Zwingelstein, 1995) :

- les méthodes sans modèle (model-free methods)
- les méthodes à base d'un modèle (model-based methods)

A. Principe des méthodes de surveillance sans modèles

Il y a plusieurs systèmes industriels qui ne peuvent être modélisés. Différentes causes sont à l'origine de cette réalité, on cite par exemple la complexité du système ou bien la reconfiguration à plusieurs reprises en cours de fonctionnement. On ne peut appliquer sur ces systèmes que les méthodes de surveillance sans modèles. Ces méthodes nécessitent l'historique du procédé et s'appuient sur des règles heuristiques ou sur des exemples d'exécution. Parmi ces méthodes, on trouve :

- **La méthode du seuillage** : cette méthode s'appuie sur la comparaison entre les signaux fournis par le capteur avec des valeurs limites constantes ou adaptatives (Montmain, 1992). Le franchissement de ces valeurs seuils indique la présence d'une anomalie. La gravité de l'anomalie peut être déterminée par la mise en place de deux seuils de détection, le franchissement du premier correspond à la présence probable d'un défaut et le second caractérise sa gravité.

- **Les méthodes statistiques** : ces méthodes supposent que les informations transmises par les capteurs possèdent certaines propriétés statistiques, sur lesquelles des tests de seuil sont effectués (Basseville, 1998) (Zemouri, 2003). L'observation de la moyenne ou de la variance d'un signal peut détecter la présence d'une anomalie.

- **La reconnaissance des formes** : ces méthodes utilisent des algorithmes de classification des formes et des mesures (Dubuisson, 1990) (Ondel, 2006). L'utilisation de la méthode de reconnaissance des formes par un système de surveillance, se déroule en trois phases :

  - Phase d'analyse : durant cette phase l'espace de représentation des données est déterminé et réduit, ainsi que l'espace de décision qui permet la spécification de l'ensemble des classes possibles.

  - Phase de choix : durant cette phase une méthode de décision est retenue. Cette méthode permet de choisir une règle de décision pour classer les nouvelles observations selon leurs classes.

  - Phase d'exploitation : dans cette phase le mode de fonctionnement du système est déterminé grâce aux nouvelles observations obtenues sur le processus.

- **Les systèmes experts** : ces systèmes se basent sur une information heuristique pour relier les symptômes aux défauts (Zwingelstein, 1995). Ces systèmes font l'association empirique entre effet est cause grâce à des règles (Farreney, 1989). Ces liaisons sont généralement basées sur l'expérience de l'expert au lieu de se baser sur la connaissance de la structure et du comportement du système.

B. Principe des méthodes de surveillance à base de modèles

Les méthodes de surveillances à base de modèles ont été introduites au début des années 70. Plusieurs travaux ont été réalisés sur ces méthodes (Frank, 1990), (Patton and Chen, 1991a), (Patton, 1994), (Frank, 1996), (Dubuisson, 2001) et (Hamscher et al., 1992). Etant donnée l'importance de ces méthodes dans nos travaux la sous-section suivante leur est consacrée.

2.4.3. Méthodes de surveillance à base de modèles

A. Modélisation des systèmes

La modélisation d'un système consiste à établir des liens ou des relations de contraintes entre les différentes variables composant ce système. La nature et la complexité des modèles sont les principales différences qui peuvent différentier un modèle d'un autre. Ces modèles peuvent être prédictifs (équations d'état, automate à états finis), qualitatifs (équation de confiance), structurels ou analytiques (graphes structurels), explicatifs (graphes causaux temporels) ou associatifs (système experts, reconnaissance des scénarios). Généralement en automatique pour caractériser le comportement normal d'un système on utilise des modèles dits de bon fonctionnement, c'est-à-dire ne comportant aucun défaut. En surveillance, par contre il faut généraliser le modèle pour qu'il puisse prendre en compte tous les états possibles y compris le comportement défaillant du système. Comme illustré dans la figure 2.9 il y a trois niveaux de connaissance qui peuvent être considérés selon (Mezyani, 2005) :

- Le niveau 1 est le plus élémentaire des niveaux, il se résume à indiquer les contraintes qui ont été violées lors de l'occurrence d'une défaillance.
  - Dans le cas des systèmes continus, au niveau 1 le système peut être décrit par des équations d'état :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ y(t) = h(x(t), u(t)) \end{cases}$$

Avec $x(t) \in X \subseteq \mathbb{R}^n$ est la variable d'état continue,

$u \in U \subseteq \mathbb{R}^n$ est le vecteur des entrées de commande.

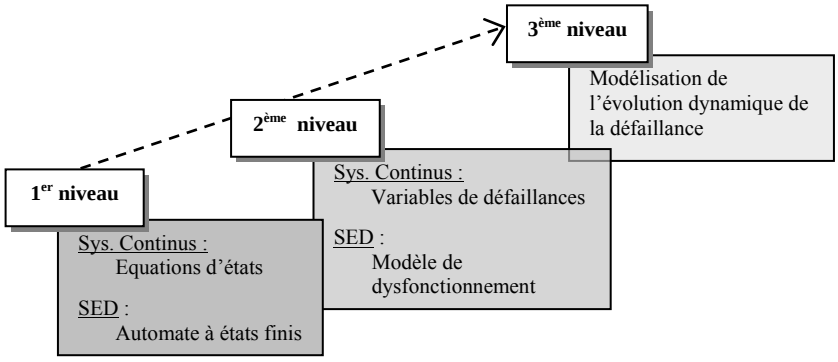


FIG. 2. 9 – Niveaux de connaissance pour la surveillance

- Dans le cas des systèmes discrets, au niveau 1 le système peut être décrit soit par l'automate à états finis, soit par réseaux de Petri
  - Automate à états finis:

$$A : \langle Q_n, \Sigma_n, \delta, q_0 \rangle$$

Où

- Q_n ensemble d'états discrets normal ;
- q_0 est la condition initiale ;
- Σ_n est l'ensemble d'événement correspondant aux transitions vers les états normaux $q_n \in Q_n$, $\Sigma_n = \Sigma_0 \cup \Sigma_{n0}$ peut être partitionné en deux ensemble d'événements $\Sigma_n = \Sigma_0 \cup \Sigma_{n0}$ où Σ_0 représente l'ensemble des événements observables et Σ_{n0} représente l'ensemble des événements non observables (Sampath et al., 1995).
- δ est la fonction de transition.

Un automate à états finis est dit *déterministe* si à partir d'un état donné, au plus une seule transition est possible sur l'occurrence d'un évènement. Le modèle d'automate à états finis de l'exemple 2.3.1.B est illustré par la figure 2.10.

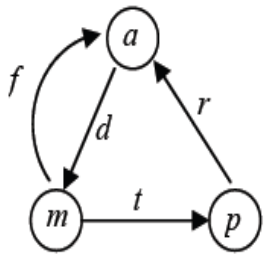


FIG. 2. 10 – Exemple d'un automate à états finis

- Réseaux de Petri :

  Le modèle Réseau de Petri (RdP) à été introduit en 1964 par C. A. Petri (Petri, 1962). Les réseaux de Petri est un outil graphique et mathématique applicable à une large variété de systèmes. Ils sont très prometteurs pour la description et l'étude des systèmes de processus concurrents et parallèles évoluant dans le temps de façon discrète, déterministe et/ou stochastique (David et Alla, 1995).

L'utilisation de ces modèles comme référence pour la surveillance permet la détection des défaillances dans le système.

- Le niveau 2 de connaissance est plus précis grâce à l'introduction de variables supplémentaires, variables de défaillance pour les systèmes continus ou modalité de l'état pour les SED.

  - Pour les systèmes continus les défaillances peuvent être additives ou multiplicatives suivant la manière dont les variables de défaillance influencent les équations du modèle. L'équation de l'état s'écrit sous la forme :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), d(t), \varphi(t)) \\ y(t) = h(x(t), u(t), d(t), \varphi(t)) \end{cases}$$

Où

- $d(t) \in D \subseteq \mathbb{R}^{\delta}$ est le vecteur des entrées inconnues et des perturbations,
- $\varphi(t) \in F \subseteq \mathbb{R}^{\zeta}$ est le vecteur des défaillances.

Ainsi le modèle d'état est qualifié de modèle défaillant. Le vecteur $\varphi(t)$ peut être scindé en trois sous vecteurs ϕ_{proc}, ϕ_{act}, ϕ_{capt} :

- ϕ_{proc} symbolise les défauts physiques du procédé. Ceux qui affectent la fonction F.
- ϕ_{act} symbolise les défauts actionneurs.
- ϕ_{capt} symbolise les défauts capteurs.

Le modèle s'écrit alors sous la forme :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), d(t), \varphi_{proc}(t), \varphi_{act}(t)) \\ y(t) = h(x(t), u(t), d(t), \varphi_{capt}(t)) \end{cases}$$

- Pour les SED, le système de niveau 2 peut être modélisé par un automate qui permet de prédire des évolutions normale et défaillante. Ce modèle appelé également modèle de dysfonctionnement, est formellement défini par (Sampath et al., 1996) :

$$G : \langle Q_n, \Sigma_n, \delta, q_0 \rangle$$

Avec,

- $Q = Q_n \cup Q_d$, Q_n l'ensemble d'état discrets normaux et Q_d est l'ensemble d'états discrets défaillants ;
- $\Sigma = \Sigma_n \cup \Sigma_d$, Σ_n l'ensemble d'événements correspondant aux transitions vers les états normaux et Σ_d l'ensemble d'événements correspondant aux transitions vers les états défaillants.

La figure 2.11 représente le modèle automate à états finis décrits à un niveau 2 de connaissance d'un fonctionnement d'un moteur. Le moteur peut êtres décrit par 4 états discrets : moteur en **Marche** (M), moteur à **l'Arrêt** (A), moteur **Bloqué en Marche** (BM) et moteur **Bloqué à l'Arrêt** (BA).

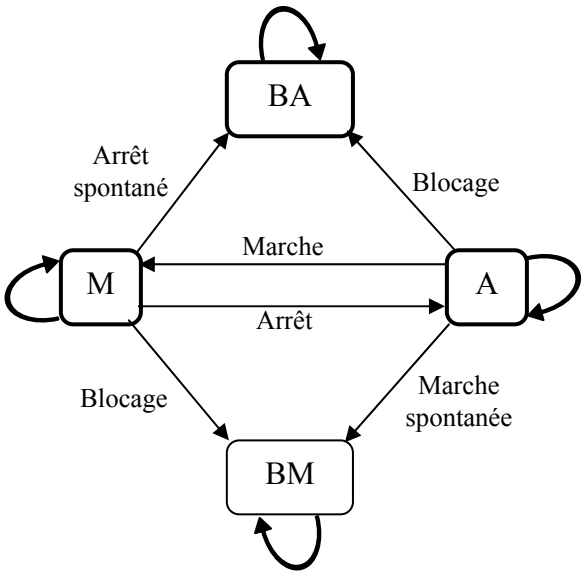


FIG. 2. 11 – Automate de dysfonctionnement

- Le niveau 3 de connaissance nécessite une modélisation de l'évolution dynamique de la défaillance. Des évolutions supplémentaires reliant les variables de défaillance sont ajoutées au modèle de bon fonctionnement. Une connaissance précise des phénomènes physiques est requise, et des données expérimentales du processus défectueux doivent pouvoir être utilisées.

B. Principe de la surveillance à base de modèle

Les méthodes de surveillance à base de modèle utilisent toutes les connaissances et les informations délivrées par le modèle. Le principe de ces méthodes est de comparer l'évolution réelle du système avec l'évolution prévue et établie selon le modèle. Afin de réaliser cette opération on utilise des modèles de bon fonctionnement. Ces méthodes de surveillance peuvent générer deux résultats possibles. Le premier, en vue de la détection, et cela par la confrontation des données relevées au cours du fonctionnement réel du système avec la connaissance que l'on a de son fonctionnement normal. Le

deuxième, en vue de la localisation, et cela par la confrontation des données relevées au cours du fonctionnement réel du système avec la connaissance que l'on a de son fonctionnement défaillant. La figure 2.12 illustre l'architecture générale de la surveillance à base de modèle.

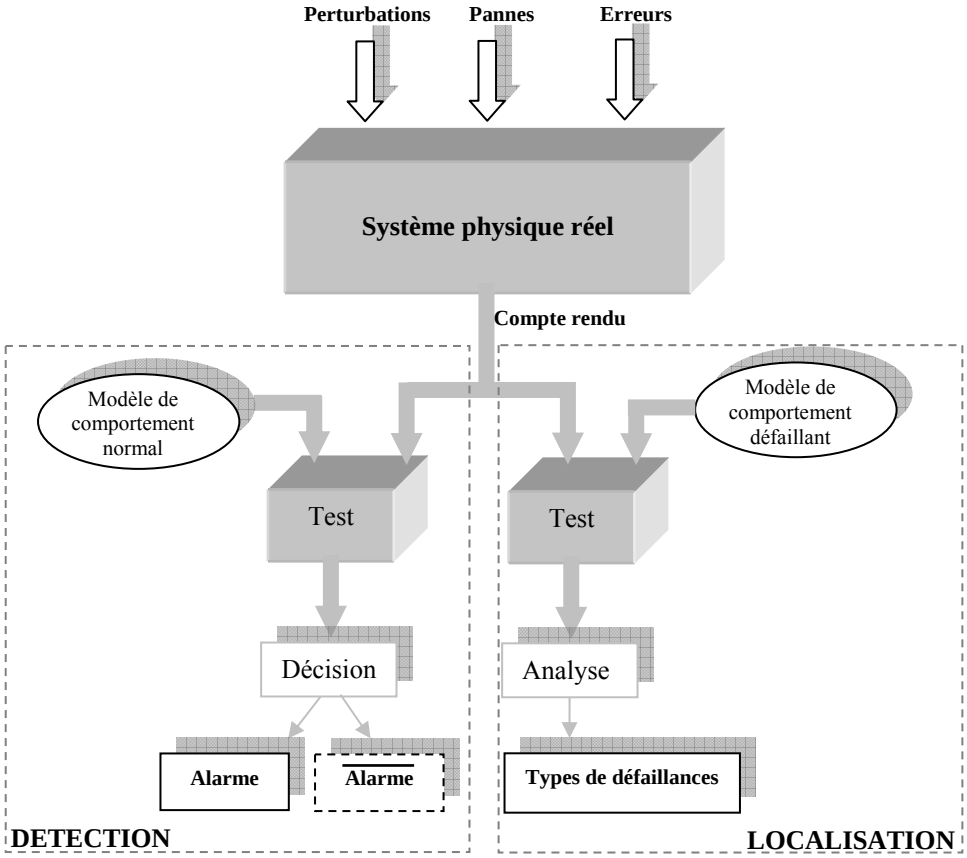


FIG. 2. 12 – Principe de surveillance à base de modèle

Concernant la surveillance des systèmes continus, on utilise un signal appelé *résidu* pour caractériser la comparaison entre le comportement réel du système et le comportement prévu.

Concernant la surveillance des systèmes à évènements discrets, la comparaison consiste à vérifier la concordance entre les évènements prévus par le modèle et les évènements observés grâce aux capteurs discrets.

Les deux types de surveillance cités seront explicités en détail dans les sections suivantes.

C. Surveillance à base de modèles des systèmes continus

Plusieurs travaux ont été consacrés aux méthodes de surveillance à base de modèles des systèmes continus notamment dans les années 70 (Model-based FDI : Fault Detection and Isolation) (Mehra and Peschon, 1971 ; Jones, 1973 et Willsky, 1976) et plus récemment (Isodori, 1995 ; Koutsoukos et al, 2001). Ces méthodes sont basées sur la génération de résidus.

Définition 2.1 : un résidu est un signal indicateur de défauts. Il reflète la cohérence des données mesurées sur le système avec un modèle comportemental de celui-ci sous l'hypothèse de fonctionnement normal (non défaillant) (Mezyani, 2005).

❑

Dans une évolution normale sans perturbations les résidus sont égaux à zéro. Lors de l'occurrence d'une défaillance les résidus ont alors une valeur significative. En réalité un résidu n'est jamais égal à zéro et cela pour cause les perturbations des bruits qui affectent le système et les erreurs de modélisation. Cependant ces influences peuvent être minimisées en améliorant la robustesse des résidus (Patton and Chen, 1991b). Si l'écart mesuré entre l'état du système et son modèle comportemental dépasse un seuil bien déterminé, il y a alors détection d'une défaillance. Il s'en suit une génération d'un résidu qui sera comparé avec toutes les signatures des défaillances connues, dans le but d'isoler et d'identifier la défaillance.

C.1 Méthodes de génération de résidus

L'algorithme utilisé pour obtenir les résidus est appelé générateur de résidus. La conception d'un générateur de résidus est appelé communément dans la littérature : le Problème Fondamental de Génération de Résidus (FPRG : Fundamental Problem of residual Generation). Parmi les méthodes utilisées pour la génération des résidus on trouve principalement la méthode d'estimation paramétrique, la méthode à base de relation de redondance analytique et la méthode à base d'observateur.

- La méthode d'estimation paramétrique : cette méthode s'appuie sur un modèle paramétrique décrivant l'évolution du système, ainsi on peut déterminer les

valeurs des paramètres du système en fonctionnement normal. Elle consiste à identifier les divers paramètres du système et à comparer ces estimations aux valeurs nominales des paramètres. l'écart résultant est utilisé comme résidu (Isermann, 1984 ; Willsky, 1976).

- La méthode à base de *Relation de Redondance Analytique* ou méthode *d'espace de parité*. Cette méthode repose sur la concordance entre le modèle du procédé avec les mesures provenant des capteurs (Chow and Willsky, 1984).
- La méthode à base d'observateurs : le principe général est de concevoir un système dynamique permettant de donner une image, ou une estimation, de certaines variables, ou combinaison de variables, nécessaire au bouclage. Lorsque le système est dynamique et que certaines variables (conditions initiales) sont connues, l'estimation n'est correcte qu'après un certain temps de convergence, fixé par la dynamique de l'observateur. Le principe général consiste à comparer des fonctions de sorties estimées avec les mêmes fonctions des sorties mesurées. L'écart entre ces fonctions est utilisé comme résidu (Jones, 1973 ; Mezyani, 2005).

C.2 La détection

Le rôle de la détection est de signaler l'apparition d'une défaillance, pour ce faire un modèle du fonctionnement normal du système est nécessaire. Les résidus générés par le modèle sont égaux à zéro si le modèle utilisé traduit parfaitement le comportement du système et ceci dans le cas d'absence de perturbations ou d'erreurs de modélisation. Une alarme est déclenchée si il y a au moins un résidu dont la valeur est différente de zéro.

C.3 La localisation

La localisation consiste à déterminer l'origine d'une défaillance qui a été détectée plutôt. Cette étape nécessite l'utilisation d'un modèle de comportement défaillant au niveau 2 de connaissance sur les défaillances, c'est-à-dire la connaissance des variables de défaillance pour les systèmes continus ou modalité de l'état pour les SED.

C.4 L'identification des défaillances

L'identification permet de déterminer les caractéristiques précises de la défaillance. Cette tâche est très complexe à réaliser car elle nécessite l'utilisation d'un modèle comportemental du système en présence des défaillances avec un niveau de connaissance 3 qui suppose la connaissance de la structure et de la dynamique de la défaillance. L'identification permet d'établir l'état précis du système et aussi la mise en place de procédures tolérantes aux défaillances.

C.5 La décision

La concordance entre le modèle de fonctionnement normal et le comportement continu du système est indiquée par les résidus. Concernant les résidus en fonctionnement normal, ces derniers ne sont pas parfaitement égaux à zéro à cause des bruits de mesures, de l'imprécision des capteurs ou des paramètres du modèle. Dans le cas d'un résidu non nul, la procédure de décision doit alors décider si les causes de cet écart par rapport à zéro sont le résultat d'une défaillance ou d'une simple perturbation ou erreur d'instrumentation. Afin de rendre cette action performante et réduire le taux de fausse alarme, les résidus doivent être optimisés en les rendant plus sensible aux défaillances et moins sensible aux perturbations ou erreurs de modèle. Cette surveillance nécessite une procédure de décision très performante. Dans la littérature plusieurs méthodes de décision ont été traitées ; celles utilisant des tests d'hypothèses statistiques, ou utilisant la logique floue (Kramer, 1987), ou encore des méthodes d'intelligence artificielle (Shafer, 1976; Dubuisson, 2001).

Pour notre part on considère deux niveaux de décision qui peuvent être mis en place. Le premier est similaire à la procédure de décision standard explicitée en haut de ce paragraphe. Le deuxième niveau de décision concerne le fonctionnement global du système, en effet le système peut avoir plusieurs comportements possibles hormis l'état de fonctionnement normal, ces comportements ne sont pas nécessairement défaillants. Les résidus dans ces situations sont forcément non nuls, dans ce cas on doit établir des critères de décision afin d'identifier les résidus correspondant au fonctionnement défaillants des autres résidus.

2.5. Surveillance à base de modèle des SED

Dans cette section nous allons présenter les méthodes de surveillance des systèmes à évènements discrets. Pour chaque méthode on présentera le modèle retenu ainsi que la méthode de détection ou diagnostic des défauts. Les informations contenues dans le modèle de base définissent la méthode de surveillance. Deux classes de méthodes se distinguent : le modèle de comportement global et le modèle de comportement de bon fonctionnement.

2.5.1. Surveillance à base de modèle de comportement global

Ce modèle illustre à la fois le comportement normal et le comportement défaillant du système. A partir de ce modèle, un autre modèle déterministe appelé *diagnostiqueur* est construit, celui-ci estimera l'état actuel du système, aussi il permettra d'identifier les défauts selon l'observation de l'occurrence des évènements. Le diagnostiquer ne peut détecter un défaut qu'à partir d'un nouvel évènement observable du système ou bien à la fin de la période de temps qui lui est alloué.

A. Diagnostic à base de modèles logiques

Pour présenter cette méthode nous allons nous baser sur les travaux de Sampath qui sont une référence dans le domaine du diagnostic à base des modèles des SED (Sampath et al, 1995). Cette méthode exposée aussi dans (Derbel, 2009) s'appuie sur une présentation logique des SED, elle déduit l'apparition des défauts non observables à partir des évènements observables affectant le système.

Le principe de cette méthode est présenté dans la figure 2.13.

Cette méthode se divise en deux parties ; la première génère un modèle global sous la forme d'un automate à états finis qui traduit le comportement normal et défaillant du système. On représente les défauts par des évènements non observables. Le diagnostiqueur est constitué hors-ligne à partir de ce modèle, sous la forme d'un automate à état finis déterministes. Dans la deuxième étape l'automate du

diagnostiqueur détecte en ligne les évènements observables générés par le système et donne une estimation de son état actuel et dés défauts altérant son exécution.

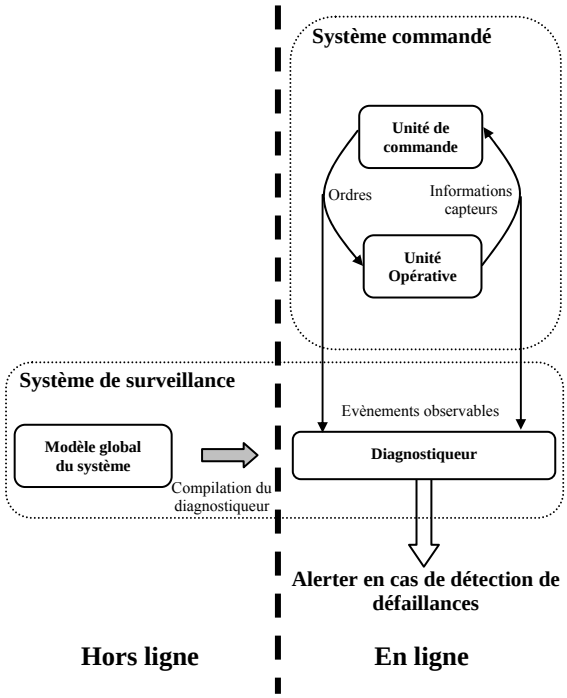


FIG. 2. 13 – Principe de l'approche de Sampath

L'automate du modèle complet est noté $G = (Q, \Sigma, \delta, q_0)$ avec Q l'ensemble des états composant le système et Σ l'ensemble des évènement qui se compose de deux sous ensemble Σ_o celui des évènements observable et Σ_{uo} celui des évènements non observables tel que $\Sigma = \Sigma_o \cup \Sigma_{uo}$. L'automate G doit vérifier deux conditions :

- 1ère condition : l'automate est vivant ; c'est-à-dire que pour chaque état q appartenant à Q, il existe une transition de sortie.
- 2ème condition : il n'y a pas de cycle contenant uniquement des évènements non observables.

On note par $\Sigma_f \subseteq \Sigma_{uo}$ l'ensemble des évènements des défauts à diagnostiquer. On suppose que l'ensemble des défauts Σ_f forme une partition de m sous-ensembles de défauts $\Sigma_f = \Sigma_{f1} \cup ... \cup \Sigma_{fm}$ notée $\Pi_f = \{\Sigma_{f1}, ..., \Sigma_{fm}\}$, avec $m \geq 1$. Chaque sous ensemble $\Sigma_{fi \in \{1,..., m\}}$ correspond à un groupe de défauts ayant le même effet sur le système, ou affectant le même composant de la partie opérative.

En se basant sur le modèle de l'automate *G*, on applique un algorithme de synthèse du diagnostiqueur en considérant la partition de défauts Π_f, afin d'obtenir l'automate du diagnostiqueur $G_d = (Q_d, \Sigma_o, \delta_d, q_{d0})$. A chaque état du diagnostiquer correspond un ensemble de paire de la forme (état, étiquettes). Dans chaque paire il y a un état estimé du système ainsi qu'un ensemble d'étiquettes $\{F_1, ..., F_2, N\}$. L'étiquette *N* indique que le système fonctionne en mode normal dans l'état estimé relié à cette étiquette. L'étiquette F_i correspond à un défaut de l'ensemble Σ_{fi}, si un état est associé à cette étiquette alors un défaut s'est produit avant d'atteindre cet état. L'état F_i-*certain* du diagnostiqueur est défini par l'association d'une étiquette à tous les éléments d'un état du diagnostiqueur. Si dans un état on ne trouve que des étiquettes de type N alors cet état du diagnostiqueur est dit normal, il est dit F_i-*incertain* s'il ne coïncide à aucun des deux cas précédents.

Pour mieux illustrer le fonctionnement du diagnostiqueur, nous considérons l'exemple suivant.

Exemple 2.5.1.A

Le modèle de l'automate à état finis ainsi que le modèle du diagnostiqueur sont présentés dans la figure 2.13. Les arcs en pointillés représentent les transitions sur les évènements non observables. Cet automate est caractérisé par l'ensemble des évènements observables Σ_o = {a, b, c, d, e, f} et l'ensemble des évènements non observables $\Sigma_{uo} = \{f_1, f_2\}$. La partition des défauts est définie comme suit : $\Pi_f = \{\Sigma_{f1}, \Sigma_{f2}\}$, avec $\Sigma_{f1} = \{f_1\}$ et $\Sigma_{f2} = \{f_2\}$.

L'état initial du diagnostiqueur est l'état 1*N*. À partir de cet état et suite à l'occurrence de l'évènement a, le diagnostiqueur évolue vers l'état 2*N*, après suite à l'occurrence de l'évènement b, il évolue soit vers l'état 4, soit l'état 6 ou l'état 7. Le diagnostiqueur indique alors que l'évènement b autorise l'accès soit à l'état 4 ou l'état 7 en détectant l'apparition des défauts f_1 et f_2 par les étiquettes respectives F_1 et F_2, soit l'état 6 en fonctionnement normal par l'étiquette *N*. Les états postérieurs à l'observation de l'évènement c sont soit l'état 5, soit l'état 8 ou l'état 10. Dans ces états le diagnostiqueur détecte l'état 10 comme celui du fonctionnement normal et les états défaillants 5 et 8 suite à la détection des défauts respectifs f_1 et f_2. Dans cet état l'isolation des défauts n'est pas réalisée, et le bon fonctionnement du procédé ne peut

être garantit. Les états {6*N*, 4f_1, 7f_2} et {10*N*, 5f_1, 8*C*} sont considérés comme des états incertains. L'occurrence de l'évènement e permettra d'isoler et de détecter le défaut f_2. L'état {9 f_2} est F_2-certain. En effet, la fonction de décision annonce l'apparition d'un défaut de l'ensemble Σ_{f2}.

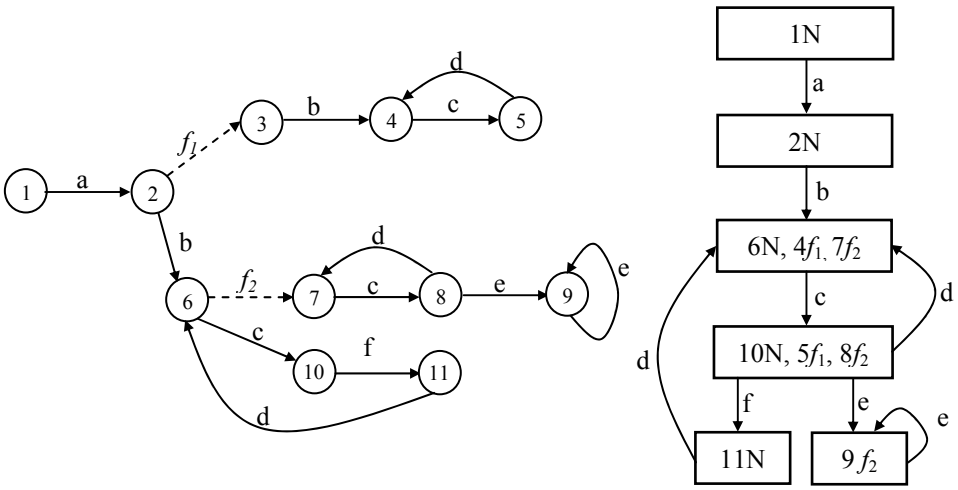


FIG. 2. 14 – Exemple d'un modèle d'automate à état finis G et son diagnostiqueur G_d

B. Diagnostic des défauts intermittents

Au cours du fonctionnement du système, un défaut peut surgir plusieurs fois, il est appelé « défaut intermittent ». Le nombre d'occurrences de ce défaut ainsi que sa durée peuvent le transformer d'un défaut intermittent vers un défaut permanent. Les travaux de (Correcher et al, 2003) proposent de construire un diagnostiqueur ainsi qu'un système d'organisation de l'information qui sera utilisé pour la sauvegarde des propriétés des défauts intermittents, c'est-à-dire leur durée et leur fréquence d'apparition. Ces informations seront obtenues lorsque le diagnostiqueur détectera un défaut. L'occurrence d'un défaut intermittent modifie le comportement du système, il bascule de l'état de fonctionnement normal vers un état défaillant. Le système revient à l'état de fonctionnement normal si l'évènement de réversion respectif au défaut survient. Le comportement du système est illustré par la figure 2.14. Dans cette figure l'état *E* correspond à tous les états possibles du système, E_F est l'ensemble des états défaillants et E_N correspond à l'ensemble des états de fonctionnement normal. On a ainsi la relation $E = E_F \cup E_N$. Les ensembles $E_{Fi} \subseteq E_F$ et $E_{Ni} \subseteq E_N$ correspondent aux

états atteignables après l'apparition d'un défaut et aux états atteignables suite à l'apparition d'un évènement de réversion. On note F_i l'ensemble des évènements conduisant à l'état défaillant et RF_i l'ensemble des évènements de réversion, ces évènements sont non-observables. Durant l'évolution du système il peut y avoir une alternance de comportements normaux et fautifs selon l'occurrence des évènements F_i et RF_i. Dans les travaux de (Sampath et al, 1996) on présente une solution (Fig. 2.15) qui consiste à modifier le cycle des évènements de défauts intermittent et de leur réversion, et cela en injectant dans ce cycle des évènements observables σ_i et σ_j. l'évènement σ_i représente le basculement vers l'état défaillant et l'évènement σ_j représente la réversion vers l'état de fonctionnement normal. Cette méthode nécessite parfois l'ajout de capteurs supplémentaires si ceux déjà présents dans le système ne permettent pas de détecter les évènements non-observables.

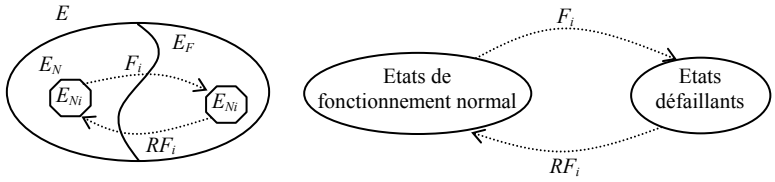


FIG. 2. 15 – Evolution du système

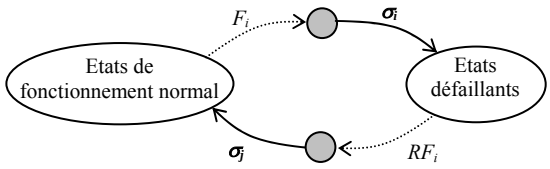


FIG. 2. 16 – Solution pour la surveillance du système

C. Diagnostic à base de modèles temporisés

Les modèles temporisés ont été utilisés dans le diagnostic et la surveillance afin d'exploiter leur apport dans l'exploitation quantitative du temps dans l'identification et la détection des défauts surtout dans les systèmes à caractère temporisé. Nous citons comme exemple de ces travaux, ceux à base de modèle d'automates temporisés (Derbel et al, 2006), (Tripakis, 2002), les travaux utilisant les RdP temporels (Ghazel et al,

2005) et les travaux utilisant les temps discrets (Zad et al, 2005), (Chen and Provan, 1997).

Soit le modèle d'un procédé illustré dans la figure 2.16. Les évènements *u, f* sont non observables par contre les évènements *a, b* sont observable. L'évènement *f* caractérise un défaut. Les deux séquences *a.f.b* et *a.u.b* ont la même projection sur l'ensemble des évènements observables *a, b*. donc nous ne pouvons pas différentier entre le comportement défaillant et normal en se basant uniquement sur la séquence d'évènements traduisant l'évolution du système. En se basant sur l'apparition de l'évènement *b* et le temps de réponse de celui-ci nous pouvons différencier une défaillance *f* d'un évènement non observable *u*. De ce fait en partant de l'état 2, l'évènement *b* doit amener à l'état 4 ou 6 et cela au bout d'un certain temps. S'il apparaît au delà de 3 unité de temps $x > 3$, la possibilité que l'évènement f est survenue est très probable. Pour cet exemple, le système n'est pas diagnosticable en cas d'absence de l'information temporelle.

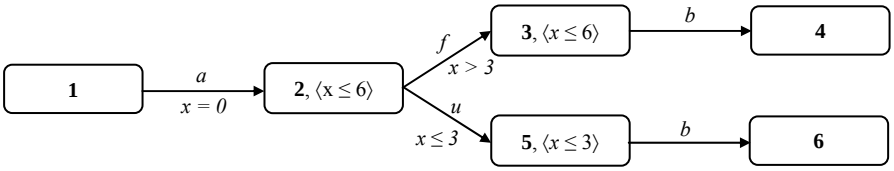


FIG. 2. 17 – Modèle temporel d'un procédé

2.5.2. Surveillance à base de modèle de comportement de bon fonctionnement

Le principe de la surveillance à base de modèle de comportement de bon fonctionnement est le suivant : Toute évolution hors du comportement normal, est une évolution anormale. Le modèle de bon fonctionnement d'un procédé expose les différentes relations entre les informations délivrées par les capteurs et les ordres fournis par l'unité de commande. Dans ces relations on trouve le niveau d'apparition des évènements ainsi que leurs dates d'apparition. La modélisation des apparitions des évènements peut être faite soit par les automates à états finis soit par les RdP. Cependant ces modèles logiques ne permettent pas la détection de tous les défauts

comme l'absence d'un retour capteur suite à un ordre de l'unité de commande. Pour palier à cet inconvénient, il est utile de connaître les liens temporels entre les différents évènements. Ces liens définissent un ensemble de contraintes régissant l'occurrence des évènements. Toute violation de ces contraintes est le résultat d'un défaut. C'est pourquoi il faut utiliser des modèles temporels pour modéliser les relations temporelles liant les différents évènements. Parmi les modèles largement utilisés dans la modélisation du comportement de bon fonctionnement on trouve les automates temporisés et les RdP temporels.

2.5.3. Surveillance des défauts interruptibles dans les SED

Cette méthode de surveillance a été présentée dans les travaux de (Allaham and Alla, 2006), (Allaham and Alla, 2007b), c'est une approche de détection pour les SED qui évoluent avec un comportement 'acceptable'. Ce comportement du procédé peut être affecté par des défauts intermittents, la modélisation du système se fera avec les automates à chronomètres. Cette méthode est basée sur l'utilisation des contraintes temporelles régissant le comportement acceptable du système et aussi sur l'utilisation de l'analyse d'atteignabilité de l'automate à chronomètres. L'estimation de l'état du système est faite en se basant sur les évènements observables. Cette étude est limitée à un type de défauts, qu'on appelle 'les défauts interruptibles' qui sont très présents dans les SED. L'objectif est de détecter ces défauts au plus tôt.

A. Principe de la surveillance des défauts interruptibles

Le modèle utilisé comme référence dans cette méthode est le modèle de comportement acceptable du procédé. Ce modèle étant un automate à chronomètre, les sommets symbolisent les différents états atteignables du système et les équations différentielles dans un sommet reflètent les dynamiques des tâches dans celui-ci. Selon les défauts interruptibles, une tâche peut être active ou interrompue. L'espace temporel dans un sommet contient toutes les trajectoires possibles du système, tout en respectant les contraintes temporelles dans ce sommet. Il est déterminé de telle manière qu'il restreint seulement l'exécution acceptable des tâches du procédé. Un événement du système à surveiller correspond à un évènement de l'automate. Cet évènement est soit

un signal provenant du capteur soit un ordre provenant de l'unité de commande. Certains chronomètres seront initialisés lors du franchissement des transitions. L'automate est synchronisé sur l'évènement du procédé commandé. Un évènement associé à une transition ne peut se produire que s'il respecte l'espace temporel du sommet source de cette transition. L'évolution d'un automate dans un sommet est possible tant que les paramètres des chronomètres respectent l'espace temporel de ce sommet. Ainsi, cet automate permettra de détecter un défaut suite à la violation de l'espace temporel associé à un sommet comme le montre l'automate illustré par la figure 2.18.a. Lorsque l'évènement σ_j se produit alors que les valeurs des chronomètres x_i, x_j, y_i et y_j appartiennent à l'espace E_i, la commutation $l_1 \rightarrow l_2$ aura lieu. Une alarme sera déclenchée lorsque la relation suivante est vérifiée : $x_i, x_j, y_i, y_j \notin E_i$. La structure du système de surveillance est schématisée par la figure 2.18.b. Le système de surveillance modélisé par l'automate à chronomètre, recueille les signaux provenant du système à surveiller. L'évolution discrète de cet automate se produit suite à l'observation d'un évènement généré par le système à surveiller. En cas de violation de l'espace temporel associé à un sommet, le système de surveillance déclenche une alarme.

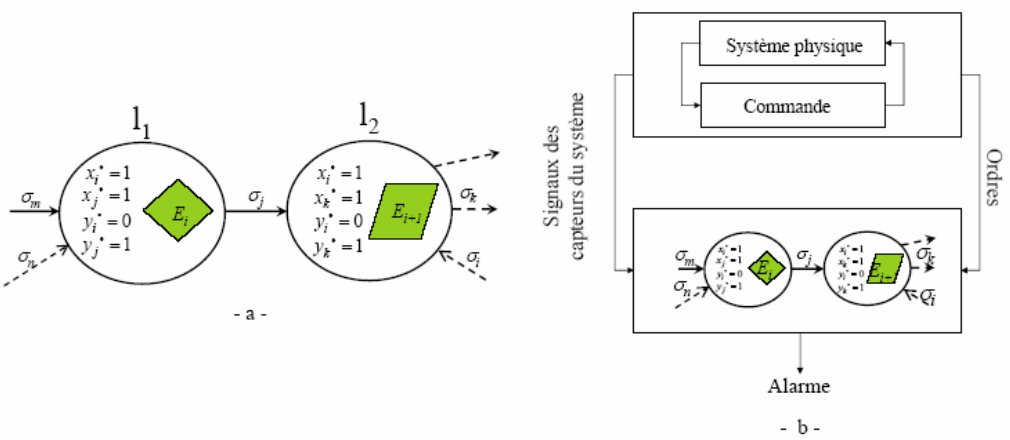


FIG. 2. 18 – a- le modèle du système commandé basé sur l'automate à chronomètre. b- La structure proposée afin d'implémenter le système de surveillance.

B. Exemple illustratif

Un système manufacturier est considéré (Allahham, 2008), il se compose d'un poste de travail, d'un robot et d'une station d'assemblage (Fig.2.19.a). Le poste de travail est

formé d'un poussoir pour embarquer les palettes, d'un convoyeur et d'un capteur de détection de fin de course. Le contrôleur de ce système est représenté dans la figure 2.19.b par un grafcet. Lors de l'occurrence de l'évènement *d*, le poussoir charge une palette sur le convoyeur. L'évènement d est généré automatiquement dans l'intervalle [0, 2] *u.t.* après l'activation de l'étape s_1 ou s_6. Lors de l'activation de l'étape s_2, le convoyeur se met en marche. L'évènement *b* est produit par le capteur si la palette arrive au point *B*, ce qui active l'étape s_3. Si le robot n'est pas occupé (l'étape s_4 est active), la transition t_3 est franchie instantanément et le robot commence à transférer la palette au poste d'assemblage. L'activation de l'étape s_5 signifie la mise en marche du robot. Après avoir atteint la station le robot dépose la palette dans celle-ci et redevient disponible pour un autre transfert. Lorsque le robot dépose la palette un capteur génère l'évènement *R*. Dans ce système, il y a deux tâches qui s'exécutent, la première exécutée par le convoyeur et le poussoir et la deuxième exécutée par le robot. On a recours aussi à deux capteurs logiques pour détecter la dynamique des tâches. Le premier capteur est associé au convoyeur, il génère les signaux s_c et r_c. Le deuxième est associé au robot est génère les signaux s_r et r_r.

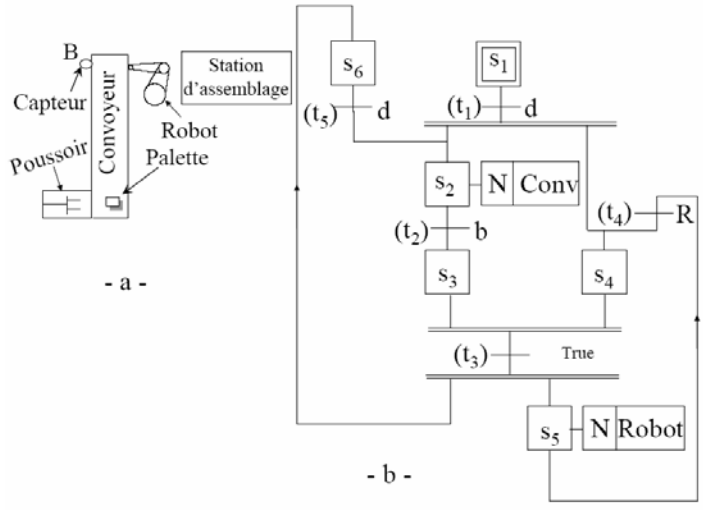


FIG. 2. 19 – a- Le système manufacturier. b- le modèle grafcet représentant la commande du système considéré.

La figure 2.20 représente les automates à chronomètre A_1 et A_2 des taches respectives 1 et 2, elle représente aussi l'automate *A* résultant de la composition des automates A_1 et A_2. L'analyse d'atteignabilité appliquée à chaque sommet nous donne les espaces

temporels correspondant à chaque sommet. Cette analyse est réalisée en ce basant sur l'analyse en avant et en arrière de l'automate. Le tableau 1 montre l'espace temporel de chaque sommet.

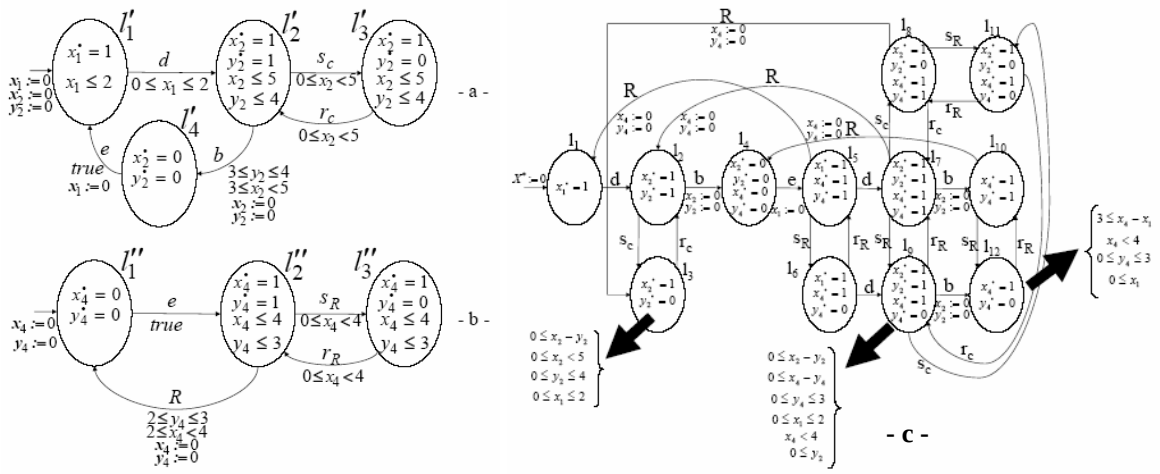


FIG. 2. 20 – a- L'automate A_1 modélisant la tâche du convoyeur. b- L'automate A_2 modélisant la tâche du robot. c- l'automate A du système considéré.

Soit la séquence d'évènement suivante : $d_{(0)} \rightarrow b_{(3)} \rightarrow d_{(3.5)} \rightarrow s_{r(4)} \rightarrow r_{r(4.5)} \rightarrow s_{r(5)} \rightarrow b_{(6.5)}$, illustrée dans la figure 2.21.a. Dans ce scénario le robot subit deux défauts interruptibles par deux fois. Le défaut surgit pour la première fois à l'instant t = 4 *u.t.* et sa réversion à t = 4.5 *u.t.* Et une deuxième fois à l'instant t = 5 *u.t.* La cause de ces interruptions peut être par exemple un défaut dans le bras manipulateur du robot. Le bras subit un défaut intermittent d'une durée de 0.5 *u.t.* Dans ce schéma de fonctionnement les évolutions des chronomètre x_2, y_2, x_4 *et* y_4 sont illustrés dans la figure 2.21.b. Dans ce chronogramme les variables x_4 et y_4 sont initialisées à l'instant t = 3 *u.t.* l'automate *A* atteint le sommet l_{12} du l_1 où les sommets fréquentés dans ce scénario sont :

$$l_1 \xrightarrow{d_{(0)}} l_2 \xrightarrow{b_{(3)}} l_4 \xrightarrow{e_{(3)}} l_5 \xrightarrow{d_{(3.5)}} l_7 \xrightarrow{s_{r(4.5)}} l_9 \xrightarrow{r_{r(4.5)}} l_7 \xrightarrow{s_{r(5)}} l_9 \xrightarrow{b_{(6.5)}} l_{12} \ldots$$

L'inégalité $0 \leq x_4 - y_4 \leq 2$ dans le sommet l_{12} détecte un défaut à l'instant $t_3 = 6.5$ *u.t.* et $y_4(t_4) = 1.5$ *u.t.* l'alarme est déclenchée car il faut au moins une durée $\alpha_2 - y_4(t_3) = 2 - 1.5 = 0.5$ *u.t.* pour accomplir la tâche de la palette correctement jusqu'à l'arrivée au poste d'assemblage. Donc la valeur correspondante de x_4 est $x_4 = x_4(t_3) +$

($\alpha_2 - y_4(t_3)$) = 3.5 + 0.5 = 4 *u.t.* Cette période de temps est supérieure à la période maximale pour le transfert de la palette au poste d'assemblage.

Sommet	Espace atteingable résultant de l'analyse en avant de $\mathbb{A}$	Espace atteingable résultant de l'analyse en avant de $\mathbb{A}^{-1}$	Espace résultant de l'intersection
l_1	$x_2 = x_4 = y_2 = y_4 == 0 \wedge 0 \leq x_1 \leq 2$	$x_4 = y_4 == 0 \wedge x_2 - y_2 < 2 \wedge 0 \leq x_1 \leq 2 \wedge 0 \leq x_2 < 5 \wedge 0 \leq y_2 \leq 4$	$x_2 = x_4 = y_2 = y_4 == 0 \wedge 0 \leq x_1 \leq 2$
l_2, l_3	$x_4 = y_4 == 0 \wedge 0 \leq x_2 - y_2 \wedge 0 \leq x_1 \leq 2 \wedge 0 \leq y_2 \leq 4 \wedge 0 \leq x_2 < 5$	$x_4 = y_4 == 0 \wedge x_2 - y_2 < 2 \wedge 0 \leq y_2 \leq 4 \wedge 0 \leq x_2 < 5$	$0 \leq x_2 - y_2 < 2 \wedge x_2 < 5 \wedge y_2 \leq 4 \wedge 0 \leq x_1 \leq 2$
l_4	$x_2 = x_4 = y_2 = y_4 == 0 \wedge 0 \leq x_1 \leq 2$	$x_4 = y_4 == 0 \wedge x_2 - y_2 < 2 \wedge 0 \leq x_2 < 5 \wedge 0 \leq y_2 \leq 4$	$x_2 = x_4 = y_2 = y_4 == 0 \wedge 0 \leq x_1 \leq 2$
l_5, l_6	$x_1 - x_4 == 0 \wedge x_2 = y_2 == 0 \wedge x_1 \leq 2 \wedge 0 \leq y_4 \wedge 0 \leq x_1 - y_4$	$0 \leq x_1 \leq 2 \wedge 0 \leq y_2 \leq 4 \wedge 0 \leq x_2 < 5 \wedge 0 \leq x_4 < 4 \wedge 0 \leq y_4 \leq 3 \wedge x_2 - y_2 < 2 \wedge x_4 - y_4 < 2$	$x_1 - x_4 == 0 \wedge x_2 = y_2 == 0 \wedge x_1 \leq 2 \wedge 0 \leq y_4 \wedge 0 \leq x_4 - y_4$
l_7, l_8, l_9, l_{11}	$x_1 + x_2 - x_4 == 0 \wedge 0 \leq x_2 - y_2 \wedge 0 \leq y_4 \leq 3 \wedge 0 \leq x_1 \leq 2 \wedge 0 \leq x_4 - y_4 \wedge x_4 < 4 \wedge y_2 \geq 0$	$0 \leq y_2 \leq 4 \wedge 0 \leq x_2 < 5 \wedge 0 \leq y_4 \leq 3 \wedge 0 \leq x_4 < 4 \wedge x_2 - y_2 < 2 \wedge x_4 - y_4 < 2$	$x_1 + x_2 - x_4 == 0 \wedge 0 \leq x_2 - y_2 < 2 \wedge 0 \leq x_4 - y_4 < 2 \wedge 0 \leq x_4 - x_2 \leq 2 \wedge y_2 \geq 0 \wedge x_4 < 4 \wedge 0 \leq y_4 \leq 3$
l_{10}, l_{12}	$x_2 = y_2 == 0 \wedge x_4 < 4 \wedge -x_1 + x_4 \geq 3 \wedge 0 \leq y_4 \leq 3 \wedge 0 \leq x_1$	$0 \leq y_2 \leq 4 \wedge 0 \leq y_4 \leq 3 \wedge 0 \leq x_2 < 5 \wedge 0 \leq x_4 < 4 \wedge x_4 - y_4 < 2 \wedge x_2 - y_2 < 2$	$x_2 = y_2 == 0 \wedge 0 \leq x_1 \wedge 0 \leq x_4 - y_4 < 2 \wedge 3 \leq x_4 - x_1 \wedge x_4 < 4 \wedge y_4 \leq 3$

Tableau 1 – L'espace temporel dans chaque sommet de l'automate A résultant de l'analyse d'atteignabilité

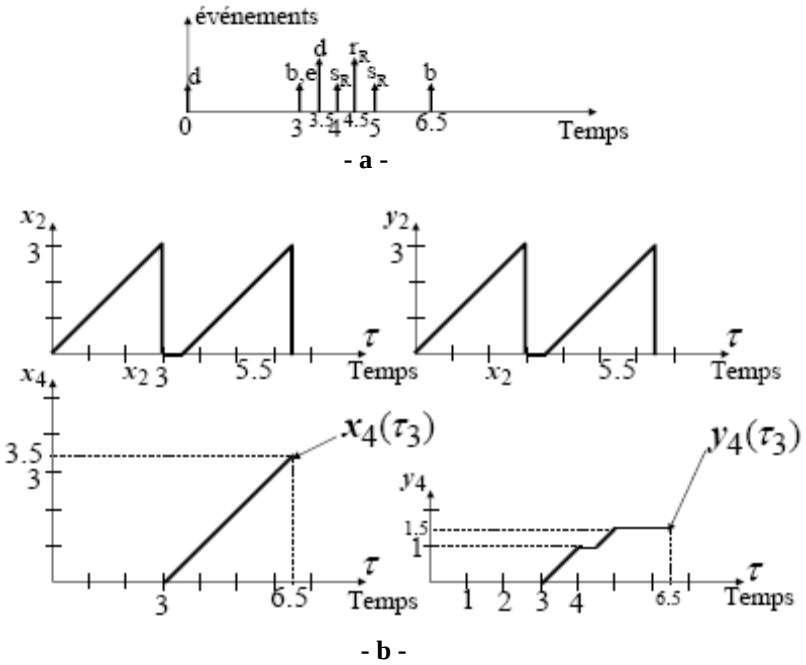


FIG. 2. 21 – Exemple de scénario d'évolution et chronogrammes correspondants

2.6. Conclusion

Ce chapitre a présenté un état de l'art sur la surveillance des systèmes dynamiques hybrides. Comme on l'a vu toute étude sur les systèmes dynamiques hybrides traite à la fois des composantes continus et discrètes du système étudié. C'est pour cette raison que les différentes approches et méthodes présentées dans ce chapitre sont décrites pour les deux types de système continu et à événements discrets. Les méthodes ont été classées selon plusieurs critères : le modèle de comportement du système, la structure de prise de décision et l'approche de l'implémentation.

Nous avons aussi présenté les deux catégories de modèle : le modèle représentant le comportement global d'un système et le modèle de comportement de bon fonctionnement. Nous pouvons alors synthétiser notre étude la manière suivante :

1. le choix du modèle dépend du niveau de connaissance que l'on a sur le système. De même la sélection du modèle représentant le comportement du système dépend des objectifs à atteindre. En effet pour la détection et le diagnostic des défauts le modèle global est plus indiqué, tandis que le modèle de bon fonctionnement est plus apte à la détection des seuls défauts.

2. Les travaux de (Rayhane, 2004) ont introduit le comportement dégradé. Ceci a permis de donner au système un degré d'évolution supplémentaire afin de réaliser sa fonction de ne pas être sanctionné dés le moindre écart.

3. Trois approches d'implémentation du système de surveillance sont utilisées dans la littérature : l'approche filtre, l'approche comparateur et l'approche modèle de référence.

4. A notre connaissance il existe peu de contribution de la surveillance et du diagnostic des systèmes dynamiques hybrides.

L'introduction du temps permet de rendre plus efficace les systèmes de surveillance. Cette information additionnelle permet de détecter des défauts alors que ce n'est pas le cas pour le modèle logique. Mais peut-on aller plus en considérant des dynamiques plus riches ? L'étude présentée dans ce chapitre permet de situer l'approche de surveillance que nous allons développer dans les chapitres 3 et 4 et qui permettra de prendre en compte une classe de systèmes plus importante.

chapitre 3

INTRODUCTION A LA SURVEILLANCE DES SYSTEMES DYNAMIQUES : APPROCHE DE LA SURVEILLANCE PAR AUTOMATE HYBRIDE LINEAIRE

3.1. Introduction

Dans ce chapitre nous allons présenter dans un premier temps l'outil de modélisation utilisé dans notre approche de la surveillance des systèmes dynamiques hybrides à savoir les automates hybrides linéaires. Dans la littérature la plupart des études de surveillance et de diagnostic ont étés réalisées sur les systèmes temporisés et cela en utilisant les automates temporisés. Dans notre approche de la surveillance des systèmes dynamiques nous allons développer ces études et introduire par la suite la surveillance par automate hybride linéaire.

Dans un deuxième temps, une approche de la surveillance des SDH sera explicitée.

3.2. Modélisation par automates

Dans cette section, plusieurs classes d'automates seront présentées. Tout d'abord les automates temporisés avec l'étude d'atteignabilité correspondante ensuite, cette étude sera développée sur les automates hybrides linéaires.

3.2.1. Les automates temporisés

Le modèle de l'automate temporisé se base sur une machine à états finis composée d'un ensemble de variables continues par morceaux, appelées horloges (Alur et Dill, 1994) (Bengtsson et Yi, 2004). Ces variables ont pour rôle de compter le temps écoulé. Lorsque le système évolue dans un sommet, les variables sont incrémentées uniformément avec le passage du temps. Toutes les horloges faisant partie du modèle sont synchronisées et avancent avec la même période. L'automate temporisé se caractérise par l'uniformité des dynamiques de ces horloges, elles ont toutes la même valeur, à savoir $\dot{x} = 1$. On appelle *invariant du sommet* le prédicat sur les valeurs des horloges associé à chaque sommet de l'automate. La présence d'un automate dans un sommet est conditionnée par le respect des valeurs des horloges à l'invariant correspondant à ce sommet. Le franchissement d'une transition se fait instantanément. Certaines horloges sont réinitialisées à zéro suite à ce franchissement. Cette

réinitialisation est réalisée par une *affectation*. On associe à chaque transition une *garde,* c'est un prédicat sur la valeur des horloges. Si la valeur des horloges satisfait la garde alors la transition qui lui est associée peut être franchie. Les gardes modélisent les contraintes temporelles qui régissent l'évolution du système.

Un automate temporisé est défini formellement comme suit (Alur et Dill, 1994).

Définition 3.1. Un automate temporisé est un 6-uplet $A = (S, S_0, X, \sum, I, T)$, où :

- S est un ensemble fini de sommets ;
- $S_0 \subset S$ est le sommet initial ;
- X est l'ensemble fini des horloges ;
- $\sum$ est l'ensemble des symboles ;
- I est une application qui associe un invariant du sommet $I(S_m)$ à chaque sommet $S_m \subset S$;
- T est l'ensemble des transitions. Une transition est un 5-uplet $(S_m, a, g, g_{m,m+1}, A_{m,m+1}, S_{m+1})$, où :
  - S_m est le sommet source ;
  - $a \in \sum$ est un ensemble associé à un événement ;
  - $g_{m,m+1}$ est la condition de franchissement i.e. la garde ;
  - $A_{m,m+1}$ est l'affectation ;
  - S_{m+1} est le sommet destination.

❑

Définition 3.2. L'état d'un automate temporisé est défini par le couple (S_m, v), où S_m désigne le sommet et v une valuation d'horloges qui vérifie l'invariant du sommet $I(S_m)$.

❑

Tant que les valeurs d'horloges respectent l'invariant, on peut séjourner dans le sommet associé à cet invariant. Un sommet ne défini pas un seul état mais tout un espace qui est décrit par l'ensemble des valeurs que peuvent prendre les horloges. L'espace d'horloge Q_n appartenant au sommet S_m est appelé région dans un sommet et noté (S_m, Q_n).

L'étude d'un processus modélisé par un automate est basée généralement sur une analyse d'atteignabilité des états de cet automate. Pour savoir si on peut atteindre une région Q à partir d'une région Q_0, on peut utiliser deux méthodes.

La première consiste à calculer l'espace de tous les états qui peuvent être atteint depuis des états appartenant à la région Q_0. L'ensemble de ces états est appelé successeur de la région Q_0. Si l'espace calculé contient des états qui appartiennent également à la région Q, alors on peut conclure que cette région est atteignable depuis Q_0. Cette méthode est appelée *méthode d'analyse en avant*.

La deuxième méthode est basée sur le calcul de l'ensemble de tous les états à partir desquels on peut atteindre des états de la région Q. L'ensemble de ces états est appelé prédécesseur de la région Q. Si l'espace calculé contient des états qui appartiennent également à la région Q_0, alors on peut conclure que la région Q est atteignable depuis Q_0. Cette méthode duale à la méthode d'analyse en avant, est appelée méthode d'analyse en arrière. Ces deux méthodes sont explicitées dans (Alu, 1999), (Sava, 2001),. Ces procédures d'analyse d'atteignabilité ont été implémentées dans les logiciels dédiés à la vérification des systèmes temporisés et hybrides. Parmi ces logiciels nous citons Hytech (Henzinger et al., 1997) et PHAVer (Frehse, 2005), (Asarin et al., 2006). Dans notre travail, nous nous intéressons seulement aux procédures de calcul des successeurs et prédécesseurs d'une région, que nous présentons par la suite.

3.2.2. Principe de calcul des successeurs d'une région

Il y a deux possibilités d'évolution dans un automate à partir d'un état, soit il reste dans le même sommet en laissant le temps d'écouler, soit il y a franchissement d'une transition de sortie vers un autre sommet. Ces deux types de successeurs sont appelés respectivement : successeur continus et discrets.

En laissant le temps s'écouler et en restant dans le même sommet on obtient un *successeur continu*.

Définition 3.3. L'état (S_m, $v+t$) est un successeur continu de l'état (S_m, v) si :

$$\exists\, t \in \Re^+ \, t.q. \; \forall\, t' \le t, \; v + t' \models I(S_m)$$
$$(S_m, v) \rightarrow (S_m, \; v + t)$$

❑

L'automate demeure dans le sommet S_m, mais les horloges sont incrémentées pendant une durée t. Pendant cette période t les horloges vérifient l'invariant $I(S_m)$ du sommet S_m.

On obtient un successeur discret en franchissant une transition de l'automate.

Définition 3.4. L'état (S_m, $v+t$) est un successeur discret de l'état (S_m, v) par le franchissement d'une transition $T_{m,m+1} = (S_m, a, g_{m,m+1}, A_{m,m+1}, S_{m+1})$ si :

$$T_{m,m+1} = (S_m, a, g_{m,m+1}, A_{m,m+1}, S_{m+1}) \in T \wedge (v \models g_{m,m+1}) \wedge$$
$$(v' = v\,[A_{m,m+1}]) \wedge (v\,[A_{m,m+1}] \models I(S_m))$$
$$(S_m, v) \rightarrow (S_{m+1}, \; v')$$

❑

Une transition ne peut être franchie que si les valeurs des horloges vérifient sa garde ($v \models g_{m,m+1}$). La valeur des horloges est modifiée lors du franchissement d'une transition selon l'affectation associée ($v' = v\,[A_{m,m+1}]$). Après franchissement de la transition, la valeur des horloge doit vérifier l'invariant du nouveau sommet atteint ($v\,[A_{m,m+1}] \models I(S_m)$). Les successeurs d'une région peuvent être déterminés de la même manière que les successeurs des états.

Définition 3.5. On appelle successeur continu de la région (S_m, Q_n) l'ensemble des états atteignables à partir de tout état (S_m, v) en laissant le temps s'écouler tout en restant dans le même sommet. On note cet ensemble $Suc_t(Q_n)$ qui est défini par l'expression :

$$v' \models Suc_t(Q_n) \; ssi \; \exists\, t \in \Re, \; v' - t \models Q_n \wedge \forall t' \in \Re^+, \; t' \le t \Rightarrow v' - t' \models I(S_m)$$

❑

Exemple 3.1. Soit l'automate temporisé illustré dans la figure 3.1.

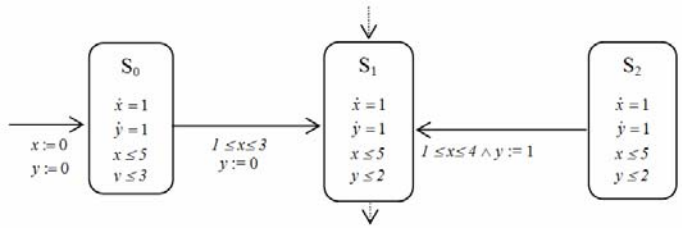


FIG. 3. 1 – Automate temporisé

Considérons l'automate temporisé de la figure 3.1. Soit (S_1, Q_1) une région dans le sommet S_1. Cette région est représentée par la figure 3.2, elle est décrite par les équations suivante : $Q_1 = 1 \leq x \leq 2 \wedge 0 \leq y \leq 1$. Le successeur continu de la région (S_1, Q_1) est :

$$\begin{aligned} \mathrm{Suc_t}(S_1, Q_1) = \quad & \exists\, t \in \Re^+.\, Q_1\,[x - t,\, y - t].I\,(S_1) \\ & \exists\, t \in \Re^+.\, [1 \leq x - t \leq 2 \wedge 0 \leq y - t \leq 1] \wedge x \leq 5 \wedge y \leq 2 \\ & 0 \leq x - y \leq 2 \wedge 1 \leq x \wedge 0 \leq y \leq 2 \end{aligned}$$

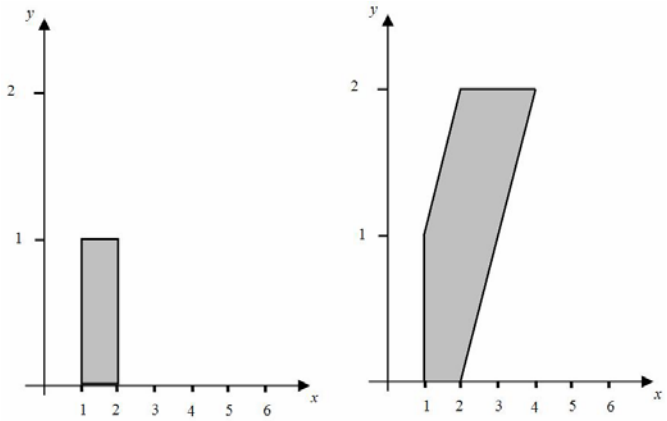


FIG. 3. 2 – Successeur continu d'une région : a – L'espace Q_1, b – Successeur (S_1, Q_1)

Définition 3.6. Soit (S_m, Q_n) une région et $T_{m,m+1} = (S_m, a, g_{m,m+1}, A_{m,m+1}, S_{m+1})$ une transition. L'ensemble des états atteignables depuis tout état $(S_m, v) \in (S_m, Q_n)$ en

franchissant la transition $T_{m,m+1}$ est appelé successeur discret de la région (S_m, Q_n). Cet ensemble noté $Succ_d\,(Q_n) = Succ_{m,m+1}\,(Q_n)$ est défini par l'expression :

$$v' \models Succ_d\,(Q_n)\ ssi\ \exists\ v\ \ tel\ que\ v \models (Q_n \wedge g_{m,m+1}) \wedge v' = v\,[A_{m,m+1}]$$

❑

Exemple 3.2. Le successeur discret de la région (S_1, Q_2), représenté par la figure 3.4.a, correspondant au franchissement de la transition $T = S_1 \rightarrow S_2$ de l'automate illustré par la figure 3.3 est :

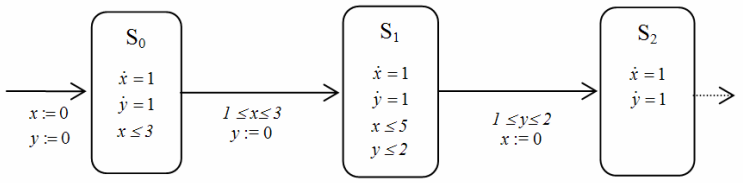


FIG. 3. 3 – Automate temporisé

$$Succ_d\,(S_1, Q_2) = Succ_{1,2}\,(Q_2) = \exists\, x', y'.\ Q_2[x',y'] \wedge x = 0 \wedge y = y'$$

$$= \exists\, x', y'.[0 \le x' - y' \le 3 \wedge 1 \le x \wedge 1 \le y \le 2 \wedge x = 0 \wedge y = y']$$

$$= \exists\, x'.[0 \le x' - y' \le 3 \wedge 1 \le x' \wedge 1 \le y \le 2 \ \wedge x = 0]$$

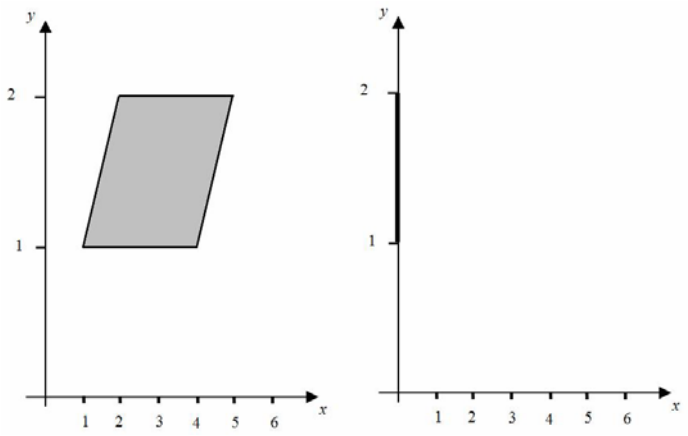
$$= [-3 \le y - x' \le 0 \wedge 1 \le x' \wedge 1 \le y \le 2 \wedge x = 0]$$

$$= [-3 + x' \le y \le 0 + x' \wedge 1 \le x \wedge 1 \le y \le 2 \wedge x = 0]$$

$$= [1 \le y \le 2 \wedge x = 0]$$

L'espace $Succ_d\,(Q_2)$ est représenté dans la figure 3.4.b

Les variables x et y représentent la valeur des horloges dans l'espace $Succ_d\,(Q_2)$ après le franchissement de la transition $T_{1,2}$, tandis que, les variables x' et y' représentent les anciennes valeurs des horloges et cela avant le franchissement de la transition $T_{1,2}$.



- a - b

FIG. 3. 4 – Successeur discret d'une région : a- Région Q_2, b- Espace successeur discret de la région Q_2

3.2.3. Principe de calcul des prédécesseurs d'une région

Tout état depuis lequel on peut atteindre un état donné est un prédécesseur de cet état. Il y a deux types de prédécesseurs : continus et discrets.

Un état depuis lequel on peut atteindre un état donné en laissant le temps s'écouler tout en restant dans le même sommet est un prédécesseur continu de cet état.

Définition 3.7. L'état (S_m, v) est un prédécesseur continu de l'état $(S_m, v + t)$ si :

$$\exists\, t \in \Re^+ \, t.q. \; \forall\, t' \leq t, \; v + t' \models I(S_m)$$
$$(S_m, v) \rightarrow (S_m, v + t)$$

❑

La notion de prédécesseur continu est duale à celle de successeur continu.

Tout état depuis lequel on peut atteindre un état donné par le franchissement d'une transition est un prédécesseur discret de cet état.

Définition 3.8. L'état (S_m, v) est un prédécesseur discret de l'état (S_{m+1}, v') par le franchissement d'une transition $T_{m,m+1} = (S_m, a, g_{m,m+1}, A_{m,m+1}, S_{m+1})$ si :

$$(S_{m,\, m+1}, a, g_{m,m+1}, A_{m,m+1}, S_{m+1}) \in T \wedge (v \models g_{m,m+1}) \wedge$$

$$(v' = v\,[A_{m,m+1}]) \wedge (v\,[A_{m,m+1}] \models I(S_m))$$
$$(S_m, v) \rightarrow (S_{m+1},\ v')$$

❑

La notion de prédécesseur discret est duale à celle de successeur discret. De la même manière que pour les états on peut définir les prédécesseurs d'une région.

Définition 3.9. L'ensemble des états à partir desquels on peut atteindre n'importe quel état $(S_m,v) \in (S_M,\ Q_n)$ en laissant le temps s'écouler, tout en restant dans le même sommet, est appelé prédécesseur continu de la région $(S_M,\ Q_n)$. Cet ensemble, noté $Pre_t(Q_n)$, est défini par l'expression :

$$v' \models Pre_t(Q_n)\ ssi\ \exists\, t \in \Re,\ v' + t \models Q_n \wedge \forall t' \in \Re^+,\ t' \leq t \Rightarrow v' + t' \models I(S_m)$$

❑

Exemple 3.3. Considérons maintenant la région $(S_2,\ Q_3)$ dans le sommet S_2 de l'automate représenté par la figure 3.3. La région Q_3 illustrée par la figure 3.5.a est définie par les équations suivantes :

$$Q_3 = [0 \leq x \leq 2 \wedge 1 \leq y \leq 3]$$

Le prédécesseur continu de cette région est calculé comme suit :

$$\begin{aligned} Pre_t(Q_3) &= \exists\, t \in \Re^+.\ Q_3\,[x + t,\ y + t] \\ &= \exists\, t \in \Re^+.\ [0 \leq t \wedge 0 \leq x + t \leq 2 \wedge\ 1 \leq y + t \leq 3] \\ &= [0 \leq x \leq 2 \wedge 0 \leq y \leq 3 \wedge\ y - x\ \leq 3 \wedge\ x - y \leq 1] \end{aligned}$$

L'espace d'horloge $Pre_t(Q_3)$ est représenté par la figure 3.5.b.

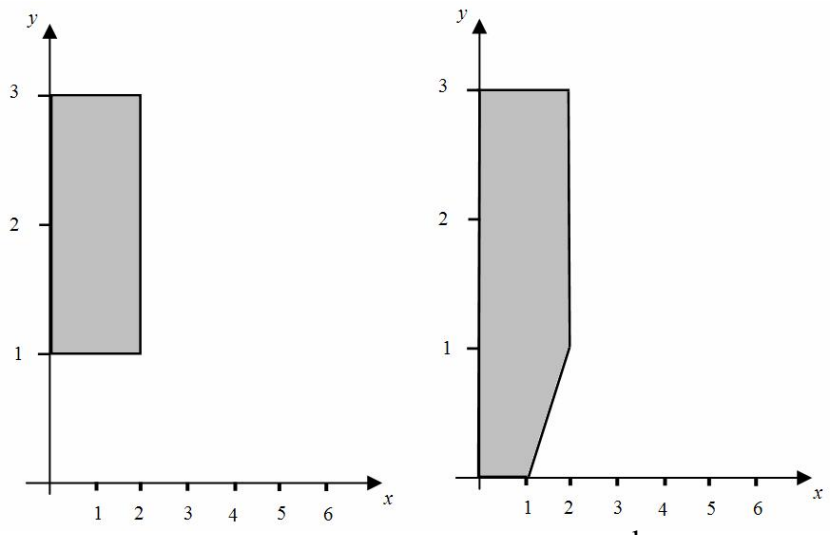


FIG. 3. 5 – Prédécesseur continu d'une région : a – espace Q3, b – Espace $Pre_t(Q_3)$

Définition 3.10. Soit (S_{m+1}, Q_n) une région et $T_{m,m+1} = (S_m, a, g_{m,m+1}, A_{m,m+1}, S_{m+1})$ une transition. L'ensemble des états à partir desquels on peut atteindre n'importe quel état $(S_{m+1}, v) \in (S_{m+1}, Q_n)$ en franchissant la transition $T_{m,m+1}$ est appelé prédécesseur discret de la région (S_{m+1}, Q_n). Cet ensemble noté $Pre_{m,\, m+1}(Q_n)$ est défini par l'expression :

$$v' \models Pre_{m,\, m+1}(Q_n) \; ssi \; v' \models (g_{m,m+1} \wedge I(S_m)) \wedge v'[A_{m,m+1}] \models Q_n$$

❑

3.2.4. Automate hybride linéaire

L'automate hybride (AH) est un outil de modélisation défini comme une machine à états finis munie d'un ensemble de variables continues et leur représentation dynamique. Le modèle résultant est un graphe états-transitions élargi avec un ensemble de variables. Les états discrets sont modélisés par les sommets du graphe où la dynamique continue a lieu. A chaque sommet est associée une fonction d'évolution qui décrit l'évolution continue des variables relative à ce sommet.

Nos travaux de recherches se basent sur les systèmes dynamiques hybrides. Le choix des automates hybrides en tant que outil de modélisation est plus qu'indiqué, en effet les *AH* permettent à la fois une modélisation des paramètres continus et discrets qui

composent les *SDH*. La composante continue est décrite par un ensemble d'équations différentielles ordinaires et la composante discrète par un automate fini.

La dynamique d'un système peut être représentée de différentes manières. Cependant plus cette dynamique est complexe, plus les possibilités d'analyse formelle sont faibles. Il s'agit donc de trouver un compromis entre la qualité du résultat et la complexité du modèle. Pour notre part, nous considérons une vision événementielle dans laquelle nous injectons un comportement continu. Allahham a déjà abordé cette problématique dans (Allahham et al, 2007b) et (Allahham et al, 2008) en modélisant les dynamiques par des horloges qui peuvent être arrêtées, d'où notre présentation plus haut des automates temporisés. La dynamique d'une horloge dans un sommet de l'automate temporisé est décrite par l'équation $\dot{x} = 1$. Ces travaux ont permis alors de synthétiser un système de surveillance pour des processus interruptibles. Nous considérons dans nos recherches des dynamiques données par des équations du type :

$$\frac{dx}{dt} = C, \text{ avec } C \in \Re$$

Ce changement de la dynamique du système n'est pas modélisable par les automates temporisés, c'est pour cette raison qu'on a recours aux automates hybrides linéaires.

Le fait de d'envisager des dynamiques variables pour un seul système nous permet d'augmenter la classe de systèmes étudiés.

Définition 3.11. Un automate hybride linéaire est un 7-uplet $H = (S, X, T, \Sigma, dif, Inv, S_0)$ (Alur et al, 1995) où :

- S : ensemble fini de sommets (appelés aussi localités, situations);
- X : ensemble fini de variables réelles (vecteur d'état à composantes continues) ;
- T : est un ensemble fini de transitions. a = $(s, g, \sigma, R, s') \in T$, avec s le sommet source, g la garde, σ l'événement associé, R l'affectation et s' le somment destination.
- Σ : ensemble fini d'étiquettes (i.e. ensemble d'actions événementielles liées aux franchissements de transitions)

- *Dif* : fonction associant à chaque sommet s ∈ S un ensemble de comportements continus (appelés aussi activités) *Dif*(*s*) :

$$\left.\frac{dx_i(u)}{du}\right|_t = x_i(t) = cste_l$$

- *Inv* : fonction associant à chaque sommet $s \in S$ un invariant *inv*(*s*) (un prédicat sur les variables ; collection d'espaces de définition du vecteur d'état associés à chaque sommet);

- $S_0 \in S$: Sommet initial.

❑

3.3. Présentation de l'approche de surveillance

Afin de résoudre la problématique présentée au début de ce mémoire, nous avons développé une méthode de surveillance pour les systèmes hybrides commandés. Nous avons pu voir dans les chapitres précédant que le temps est un des éléments principaux qui caractérisent en général l'évolution d'un processus et en particulier le système de surveillance de ce processus. Notre approche de la surveillance se base essentiellement sur ce facteur temps, dans laquelle on élabore un système de détection pour les systèmes dynamiques.

Nous considérons que les systèmes étudiés peuvent évoluer dans plusieurs modes de fonctionnement différents comme illustré par la figure 3.6. Chacun de ces modes à une dynamique propre. Par ailleurs on peut distinguer parmi eux trois catégories ;

Les modes de fonctionnement initial : ces modes de fonctionnement sont ceux à partir desquels le système démarre. Les paramètres qui caractérisent l'évolution du système sont initialisés dans ces modes.

Les modes de fonctionnement normal : ce sont les modes de fonctionnement prévu par l'opérateur. La présence du système dans ces modes le conduit inexorablement vers une exécution correcte et sans violation du cahier des charges.

Les modes de dysfonctionnement : ce sont les modes où le système évolue avec des dynamiques pouvant l'amener soit à une défaillance soit à une violation du cahier des charges. La présence du système dans ces modes de fonctionnement n'est pas toujours synonyme de défaillance, parfois on a recours à ce type de fonctionnement pour corriger les effets de certains imprévus qui apparaissent en cours de fonctionnement. Donc la présence du système de ce mode de fonctionnement ne déclenche pas systématiquement l'alarme. Nous pouvons citer comme exemple deux applications ; la première celle d'un moteur qui voit sa vitesse de rotation varier de sa vitesse nominale de fonctionnement soit elle augmente soit elle diminue suite à un dysfonctionnement ou même à une mauvaise manipulation de l'opérateur. La deuxième application est celle d'une vanne dont le débit peut varier selon la présence ou non de résidus dans les tuyaux.

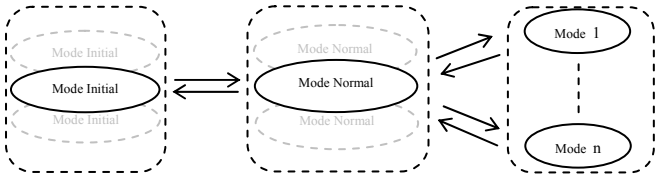


FIG. 3. 6 – Modes de fonctionnement d'un système hybride

L'élaboration d'une méthode de surveillance des systèmes hybrides à l'avantage d'avoir un champ d'application élargi. Ceci est rendu possible par la nature même des systèmes à surveiller. Par exemple, contrairement aux SED on peut envisager des vitesses d'exécution variables. Nous allons par la suite présenter la démarche de surveillance que nous avons élaborée (karoui et al, 2010b). Dans un premier temps nous modélisons le comportement du système en utilisant l'automate hybride linéaire. On établit les propriétés caractérisant l'exécution du fonctionnement normal. Par la suite on applique une procédure de synthèse à cet automate pour garder uniquement les trajectoires qui satisfont les propriétés obligatoires. Les sommets de l'automate représentent les états atteignables du système, les équations différentielles relatives à chaque sommet reflètent la situation du système dans cet état. Le passage d'un sommet à un autre est réalisé par le franchissement de la transition qui les lie, ce franchissement est conditionné par l'occurrence de l'événement correspondant à cette transition. L'espace des états atteignables synthétisé dans chaque sommet comporte toutes les situations correspondant au comportement normal.

La combinaison entre les contraintes qui régissent l'évolution des dynamiques du système et le pouvoir de l'analyse d'atteignabilité de l'automate constitue l'originalité de notre méthode de surveillance. Ceci nous permet de déterminer des espaces temporels décrits par des inégalités algébriques relatives à chaque sommet de l'automate. La violation de ces espaces nous permet de détecter une défaillance de fonctionnement au plus tôt. La notion de détection « au plus tôt » et le fait de détecter un comportement anormal et ceci sans attendre l'expiration de la date au plus tard prévue de son apparition.

3.4. Construction du système de surveillance par automate hybride linéaire

3.4.1. Présentation intuitive de l'approche

Pour bien illustrer le problème nous considérons un exemple simple de système hybride commandé, illustré dans figure 3.7. Ce système représente un atelier de collage qui se compose de deux postes, d'un convoyeur, de deux capteurs et d'un contrôleur. Au départ et dans le poste P1 il y a un dépôt de colle qui est effectué sur la pièce à traiter. Le convoyeur va transporter cette pièce vers le poste P2 où il y aura l'opération de collage. Le convoyeur débute sa tâche suite à l'ordre '***St***'(*start*) du contrôleur, après que la colle ait été déposée sur la pièce. L'événement '***Ed***' (*End*) survient quand la pièce atteint le poste P2, cet événement est détecté par le premier capteur.

Le convoyeur à deux fonctions ; la première c'est de transporter la pièce de P1 à P2 et la deuxième c'est qu'il sert de retardateur. En effet le trajet de la pièce est déterminé afin que la colle soit prête pour être utilisée. Le convoyeur a une vitesse nominale V_0 qui définit l'état normal, avec cette vitesse il devrait porter la pièce à destination en 8 u.t. (unités de temps). La durée acceptable du trajet est comprise dans l'intervalle [7, 8] u.t. Cet intervalle correspond à la période de temps durant laquelle la colle est exploitable. Le deuxième capteur logique produit les événements c_i et r_i, ces événements indiquent un changement de vitesse du convoyeur. Le fait que le transport de la pièce vers le poste P2 se fait pendant se durée et sans occurrence de défaut, représente le comportement normal du système. Dans nos travaux les défauts intermittents sont

considérés comme un cas particulier de défaillance, en effet notre système de surveillance se veut plus large dans la détection des défaillances. Dans les travaux de (Allahham, 2008) les défauts intermittents sont définis comme pouvant arrêter l'exécution du convoyeur pendant une durée indéterminée, lorsque ce défaut disparaît le système reprend son exécution normale. Pour notre part l'arrêt du système est un cas particulier parmi d'autre. Nous considérons que celui-ci peut évoluer dans plusieurs modes de fonctionnements différents, chaque mode à une dynamique de fonctionnement et dans le cas d'un défaut intermittent la dynamique correspondant au fonctionnement du convoyeur sera nulle.

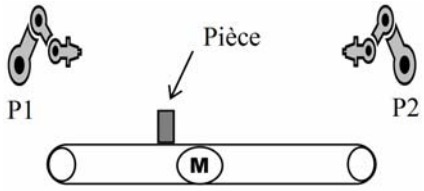


FIG. 3. 7 – Atelier de collage

Pour présenter notre approche, on construit le modèle de surveillance de l'exemple illustré par la figure 3.7. Dés que le contrôleur donne l'ordre '***St***' deux variables x et y sont initialisées (Figure 3.8). La variable x est un compteur qui mesure le temps écoulé depuis l'instant de la mise en route du convoyeur jusqu'à son arrêt correspondant à l'occurrence de l'événement '***Ed***'. La dynamique de x prendra comme valeur : $\dot{x} = 1$, Le convoyeur est en marche. Lorsque cet événement '***Ed***' se produit, la valeur de x indique si la tâche a été exécutée pendant la durée acceptable [7, 8]. Dans un fonctionnement normal la pièce se déplace sur le convoyeur et arrive au poste P2 en 8 u.t.

Dans cet exemple plusieurs modes de fonctionnement normaux peuvent être envisagés. Dans cette étape introductive on a fait le choix de ne modéliser qu'un seul. On définit le mode normal par l'état où les dynamiques caractérisants les variables x et y sont conformes aux valeurs prédéfinies avant le début de l'exécution. Dans notre cas la dynamique de x vaut 1 et la dynamique de y vaut 2. Tout écart de ces deux valeurs entraînera l'apparition d'un autre mode fonctionnement. Le système peut évoluer dans

le mode normal tant que les valeurs des compteurs x et y vérifient les invariants du sommet. Dans le fonctionnement normal, le système exécutera forcément sa tâche tout en respectant les contraintes qui lui sont imposées, à savoir $7 \leq x \leq 8$ et $y = 16$.

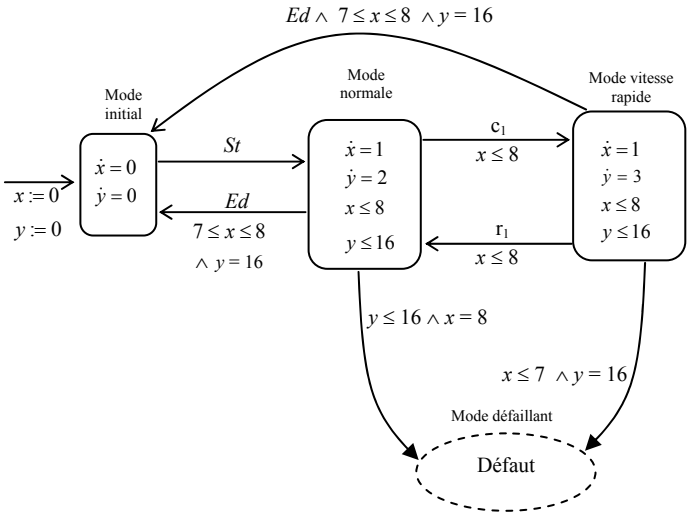


FIG. 3. 8 – Modèle de la surveillance du convoyeur

Lors du transport de la pièce du poste P1 au poste P2, la vitesse du convoyeur et donc celle de la pièce peut s'écarter de la vitesse nominale à cause de dysfonctionnements. Nous considérons dans cet exemple un changement vers une vitesse supérieure à la vitesse nominale. Ce changement de vitesse impliquera un changement d'état du système de l'état normal vers le mode vitesse rapide. Ce changement doit être observable afin d'être détecté par le capteur logique. Concernant les modes de défaillances plusieurs possibilités peuvent êtres envisagées, une accélération, une décélération ou bien un arrêt du convoyeur. Le système illustré par la figure 3.7 est présenté comme exemple introductif pour résoudre la problématique de la surveillance, il nous a paru judicieux de ne pas présenter toutes les possibilités de défaillance à l'état actuel de l'avancement des travaux.

La figure 3.9 illustre l'espace acceptable correspondant au couple de variables (x, y) cette espace est noté A. Dans cette figure on trouve également l'espace des états atteignables par l'automate. Tant que les paramètres (x, y) respectent cet espace le

système pourra respecter les contraintes temporelles et géographiques. Par ailleurs si ces paramètres dépassent cet espace une alarme se déclenche aussitôt.

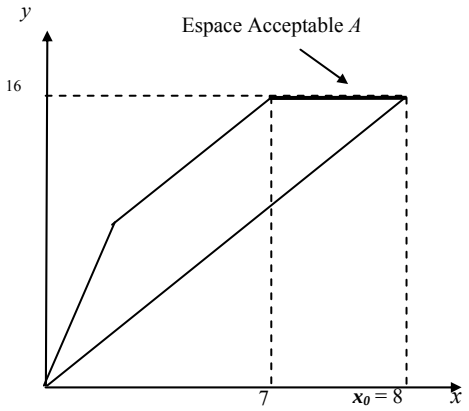


FIG 3. 9 – Espace des états atteignables

Le compteur *y* indique la position de la pièce sur le convoyeur, la longueur du trajet est égale à 16m. Lors d'une exécution normale quand l'événement ***'Ed'*** se présente on aura $y = 16$. La valeur de *y* indiquera si la pièce est arrivée au poste P2 ou pas. La dynamique de *y* reflète la vitesse de la pièce, autrement dit :

- $\dot{y} = 2$ quand la pièce évolue avec une vitesse nominale égale à 2 m/*u.t.*
- $\dot{y} = 3$ lorsque la pièce à une vitesse de 3 m/*u.t.*

On définit le mode vitesse rapide par le changement de la dynamique de *y* vers $\dot{y} = 3$.

Quand l'événement ***'Ed'*** a lieu, la valeur de *x* indique si la pièce a été amenée vers le poste P2 dans l'intervalle acceptable [7, 8]. Normalement le convoyeur mettra 8 *u.t.* pour faire ce trajet. Au cours du trajet le convoyeur peut changer de vitesse, cette mutation est indiquée par l'événement $\boldsymbol{c_1}$, la pièce évoluera dans ce cas avec une vitesse supérieure à la vitesse nominale (v = 3 m/ *u.t.*). C'est le passage au mode vitesse rapide, dans cet état la dynamique de *x* ne change pas par contre celle de *y* sera égale à 3. La condition associée à ce changement d'état et que $x \leq 8$, puisque le changement de dynamique peut avoir lieu tant qu'on ne dépasse pas l'intervalle acceptable (avant alarme). Le retour à l'état normal est indiqué par l'événement $\boldsymbol{r_1}$, la dynamique de *y* sera de nouveau égale à 2, la garde associée à cet événement est $x \leq 8$. Les événements $\boldsymbol{c_1}$ et $\boldsymbol{r_1}$ sont commandables.

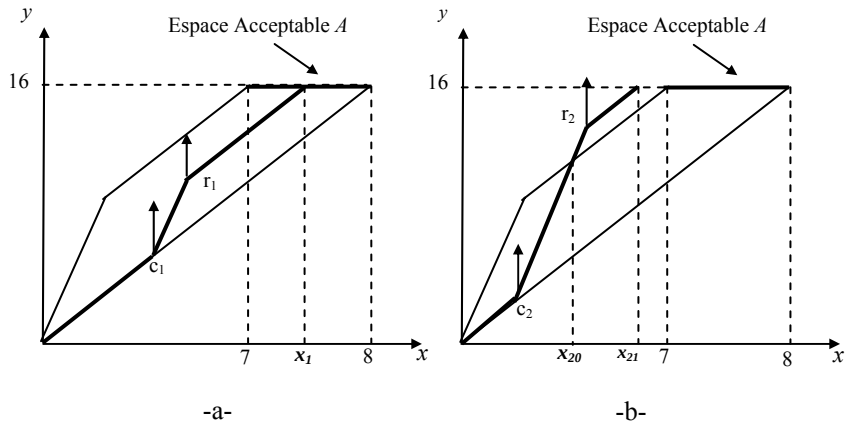


FIG. 3. 10 – (a) trajectoire 1, (b) trajectoire 2

Un retour du mode vitesse rapide à l'état initial est possible si les conditions suivantes sont satisfaites ; $7 \leq x \leq 8$ et $y = 16$. Cela signifie que la tâche a été effectuée et dans les délais.

Le passage du mode normal au mode défaut est régi par la contrainte suivante, l'horloge x indique que le temps maximal autorisé est atteint mais la pièce n'est pas encore arrivée au bout du convoyeur c'est-à-dire au poste P2. Le passage du mode vitesse rapide au mode défaut est régi par la contrainte suivante ; la pièce a atteint le poste P2 mais avec un temps inférieur à la durée minimale autorisée.

Notre objectif est de déterminer toutes les trajectoires possibles qui permettent d'atteindre le poste P2 tout en respectant les contraintes citées ci-dessus. Cela équivaut à déterminer l'espace d'état des variables (x, y) qui satisfait les conditions de terminaison correcte de la tâche.

La figure 3.10 présente deux trajectoires : la première (Figure 3.10.a) définie par les événements (c_1, r_1) la pièce atteint le poste P2 à x_1 qui est compris dans l'intervalle acceptable [7, 8]. Pour la deuxième trajectoire (Figure 3.4.b) qui est définie par les événements (c_2, r_2) la pièce atteint le poste P2 à x_{21} qui n'appartient pas à l'espace des états atteignables. Le système de surveillance déclenche une alarme à l'instant x_{20}, car même si le système commandé retourne vers le mode de fonctionnement normal, la pièce n'atteindra pas le poste P2 dans l'intervalle acceptable. Ainsi il n'est pas nécessaire d'attendre l'instant x_{21} pour déclencher l'alarme.

3.4.2. Surveillance du système par le temps enveloppe

Considérons les modèles temporels qui modélisent l'apparition d'événements produits par un procédé par des contraintes temporelles. Soit X, L'ensemble des compteurs utilisés par le modèle en vue de la surveillance de ces contraintes, avec $X = \{x_i, x_j, x_k, \ldots\}$. Les bornes supérieures des contraintes surveillées sont respectivement les suivantes : β_i, β_j et β_k, en supposons que ces horloges soient initialisées en même temps. Selon l'étude bibliographique qui a été faite sur la surveillance à base d'un modèle de comportement normal, nous remarquons que l'espace temporel atteignable des compteurs pour tout le modèle est un hypercube. Les bornes supérieures des compteurs représentent les bornes de cet hypercube (figure 3.11). Une défaillance est détectée lorsque la valeur d'un compteur dépasse les bornes de cet hypercube.

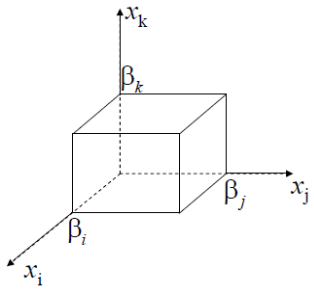


FIG. 3. 11 – Espace atteignable limitant le comportement normal d'un système caractérisé par un ensemble de contrainte temporelle

En appliquant les méthodes de surveillance présentée dans le chapitre 2 sur l'atelier de collage présenté ci-dessus, on obtient les modèles de surveillance illustrés par la figure 3.12. La figure 3.12.a représente un RdP temporel décrivant le comportement normal du système étudié. Les événements d et b sont modélisés par les transitions t_1 et t_2. La surveillance de l'espace atteignable associé au marquage de P2 détecte le défaut présenté par la trajectoire 3 (Fig 3.10.b) à l'instant 8 *u.t*. Cet instant est la valeur maximale au delà de laquelle le franchissement de la transition t_2 est considéré comme une défaillance. C'est-à-dire que la détection de la défaillance se fera à l'instant d'expiration de la date maximale prévue de l'occurrence de l'événement b qui représente la fin d'exécution. La figure 3.12.c illustre l'instant de détection de la défaillance en cas de la trajectoire n°2 utilisée précédemment. On note également que la

surveillance de l'espace atteignable dans le sommet S_2 de la figure 3.12.b détecte le défaut à l'instant 8 *u.t.* pour la même trajectoire.

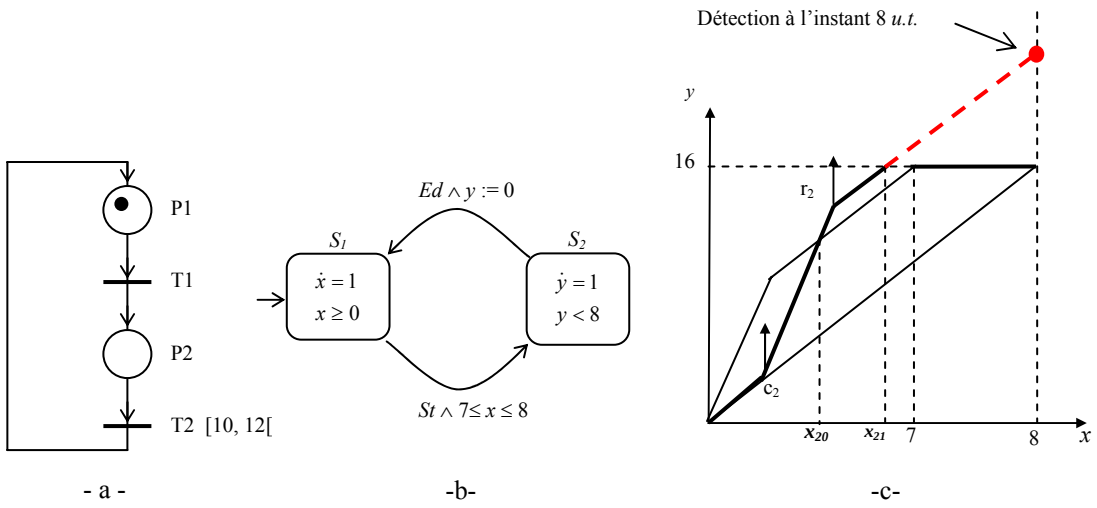


FIG. 3. 12 – Surveillance de l'atelier de collage par le temps enveloppe :
a- RdP temporel, b- par automate temporisé, c- espace d'état atteignable et instant de détection de la défaillance.

3.4.3. Démarche formelle de la construction du système de surveillance.

A. Comportement du système

On s'intéresse à la surveillance des systèmes dynamiques événementiels commandés présentés dans les chapitres précédents. L'aspect dynamique du système à surveiller est pris en compte à travers les changements de modes d'évolution. Le système est étudié dans sa totalité, seule les performances globales seront considérées, toutefois une intégration du système de surveillance sur des composants du processus est toujours possible. Donc l'évolution du système sera répartie en plusieurs modes de fonctionnement.

On considère un système de surveillance comprenant plusieurs modes de fonctionnement : mode initial, mode normal et les modes de dysfonctionnement i. Chaque mode est défini par une dynamique distincte. Dans la figure 3.6 le mode i est représenté de manière générique. L'évolution du système est caractérisée par les

variables x et y qui sont des paramètres observables, elles sont dans le système de surveillance et décrivent l'état du système. La variable x représente la durée d'exécution totale du processus et la variable y reflète l'état d'avancement du processus. Plusieurs contraintes sont appliquées à ces variables afin de vérifier que l'exécution du système s'est faite en respect des règles constituant le cahier des charges. Pour la variable x il faut que $x \in [\alpha, \beta]$ que l'on note durée acceptable, où α correspond à la durée minimale nécessaire pour réaliser la totalité du processus, aussi β correspond à sa durée maximale. Ces durées tiennent compte de l'avancement ou du retard qui peuvent être dus aux conditions d'exploitation des ressources utilisées par le processus. Autrement dit cet intervalle tient compte du changement de dynamique adopté par le système à surveiller en cours de fonctionnement. La variable y quand à elle, doit atteindre une valeur limite δ. Si $y < \delta$, l'exécution est alors incomplète, si $y > \delta$ la valeur limite est dépassée.

Le fait d'utiliser deux paramètres pour la surveillance renforce système de surveillance en qualité de précision et de fiabilité.

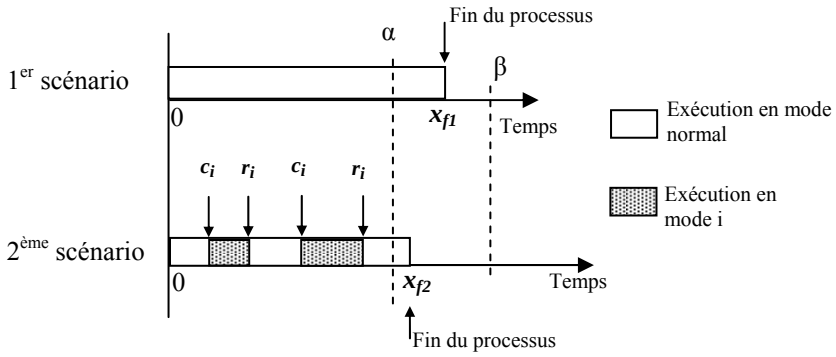


FIG. 3. 13 – Comportement du processus

Durant son évolution le processus peut basculer du mode normal vers le mode défaillant i selon l'occurrence des événements (c_i, r_i). Ces événements sont supposés être observables. L'événement c_i consiste à faire migrer le système du mode normal vers un autre mode i différent, quand à l'événement r_i son occurrence ramène le système à un mode de fonctionnement normal. Un retour à l'état initial est possible à partir du mode i si le processus a terminé son exécution tout en respectant les contraintes sur les

variables x et y. Le comportement du processus est illustré par le chronogramme de la figure 3.13.

Le comportement présenté dans cette figure décrit dans un premier temps un fonctionnement normal du système, en effet dés le départ et jusqu'à l'instant $x_{f1} \in [\alpha, \beta]$ le système n'évolue que dans un seul mode normal. C'est un scénario parfait qui ne contient aucune perturbation au fonctionnement prévu. Dans une deuxième étape la figure 3.13 décrit une autre évolution du processus, celle-ci elle comprend plusieurs modes de fonctionnement, la fin du processus se produit à l'instant $x_{f2} \neq x_{f1}$ mais $x_{f2} \in [\alpha, \beta]$, malgré les perturbations survenues lors du fonctionnement normal et qui ont amené le système vers d'autre modes de fonctionnement le processus s'est exécuté dans les délais requis.

B. Modélisation du système de surveillance

Pour modéliser le comportement du processus de la figure 3.6, nous avons recours à un modèle d'automate à trois états plus le mode Alarme. Pour simplifier la présentation on ne considère ici qu'un seul mode de dysfonctionnement. La généralisation de l'approche est une de nos perspectives de recherche. Le modèle de l'automate hybride linéaire est illustré dans la figure 3.14. Ce modèle constitué de 3 sommets S_1, S_2 et S_3 modélisant respectivement les états : initial, fonctionnement normal et mode de fonctionnement i. Les compteurs x et y sont initialisés à l'entrée du sommet S_1. Les compteurs de ce sommet sont à l'arrêt, leurs dynamiques sont nulles. Après l'occurrence de l'événement ***'St'*** de début d'exécution, le système évolue du sommet initial S_1 vers le sommet S_2 de fonctionnement normal. Dans ce sommet les dynamiques sont non nulles, la dynamique de x est 1 et celle de y est égal à λ_0. L'invariant relatif à ce sommet est $Inv(S_2) = (x \leq \beta \wedge y \leq \delta)$ où β représente la durée maximale autorisée pour l'exécution du processus et δ représente la valeur finale de y stipulant que l'exécution est achevée. Dans le troisième sommet la dynamique de x est toujours égale à 1 car x représente le temps, par contre celle de y a une valeur égale à $\lambda_i \neq \lambda_0$.

La transition $T_1 : S_1 \xrightarrow{G_{1,2}} S_2$, représente le début d'exécution de l'état initial vers le mode normal, lié à l'événement *St* émanant du contrôleur du système. Donc la garde de cette transition est $G_{1,2} = (St)$.

La transition T_2 : $S_2 \xrightarrow{G_{2,3}} S_3$, représente la migration vers le mode de dysfonctionnement i, suite à l'occurrence de l'événement c_i, la garde de cette transition est $G_{2,3} = (x \leq \beta)$.

La transition T_3 : $S_3 \xrightarrow{G_{3,2}} S_2$, représente le retour au mode de fonctionnement normal, suite à l'occurrence de l'événement r_i, la garde de cette transition est $G_{3,2} = (x \leq \beta)$.

La transition T_4 : $S_2 \xrightarrow{G_{2,1}} S_1$, représente le retour à l'état initial à partir du mode normal, c'est-à-dire la fin d'exécution du processus qui est indiquée par l'événement *Ed*, la garde de cette transition est $G_{2,1} = (Ed \wedge \alpha \leq x \leq \beta \wedge y = \delta)$.

La transition T_5 : $S_3 \xrightarrow{G_{3,1}} S_1$, représente le retour à l'état initial à partir du mode de dysfonctionnement i, c'est-à-dire la fin d'exécution du processus qui est indiquée par l'événement *Ed*, la garde de cette transition est $G_{3,1} = (Ed \wedge \alpha \leq x \leq \beta \wedge y = \delta)$.

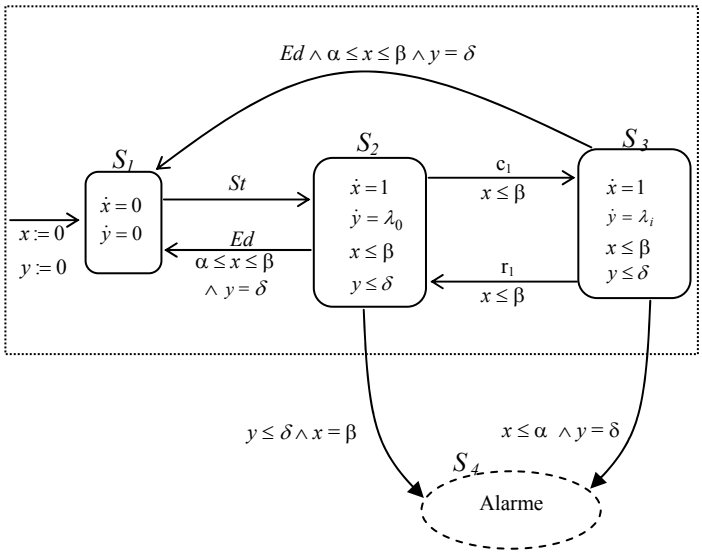


FIG. 3. 14 – Modèle de l'automate représentant le système de surveillance

Dans le modèle de l'automate (fig. 3.14), on peut avoir aussi un basculement vers un état faute impliquant le déclenchement d'une alarme. Ce basculement peut être réalisé par les transitions suivantes :

La transition T_6 : $S_2 \xrightarrow{G_{2,4}} S_4$, représente le basculement vers l'état fautif à partir de l'état de fonctionnement normal. La garde de cette transition est $G_{2,4} = (x = \beta \wedge y \leq \delta)$.

Le franchissement de cette transition indique que le temps alloué au processus a atteint sa limite maximale et que le système n'a pas encore achevé son exécution.

La transition $T_7 : S_3 \xrightarrow{G_{3,4}} S_4$, représente le basculement vers l'état fautif à partir de l'état de dysfonctionnement i. La garde de cette transition est $G_{3,4} = (x \leq \alpha \wedge y = \delta)$. Le franchissement de cette transition indique que le système a achevé son exécution trop tôt, la limite minimale n'est pas encore atteinte par x.

Le comportement de ce système hybride événementiel peut être défini par un automate hybride linéaire comme suit :

Définition 3.12. L'automate hybride linéaire représentant le système de surveillance est $H = (S, S_1, X, \Sigma_i, T, Inv, Dif)$, où :

- $S = \{S_1, S_2, S3\}$; et S_1 le sommet de départ ;
- $X = \{x, y\}$;
- $\Sigma_i = \{b, d, c_i, r_i\}$;
- $T = \{T_1, T_2, T_3, T_4, T_5\}$
- $Inv\ (S_2) = Inv\ (S_3) = \{x \leq \beta, y \leq \delta\}$;
- $Dif\ (S_1)(x) = 0$, $Dif\ (S_2)(x) = Dif\ (S_3)(x) = 1$, $Dif\ (S_1)(y) = 0$, $Dif\ (S_2)(y) = \lambda_0$, $Dif\ (S_3)(y) = \lambda_i$

❑

Remarque 1. La seule différence entre les sommets S_2 et S_3 réside dans la valeur de la dynamique de y. Par définition les deux modes sont semblables, ils correspondent à des modes de fonctionnement distincts. Cependant pour des raisons pratiques nous avons voulu différentier les modes qui permettent au système de terminer sa tâche dans les délais s'il n'utilisait que ces modes là, des modes de dysfonctionnements qui conduisent le système systématiquement vers un état défaillant.

Remarque 2. La dynamique du mode de dysfonctionnement est $\lambda_i \in [\lambda_{min}, \lambda_{max}]$, avec $\lambda_{max} \geq \lambda_{min} \geq 0$, de même pour la dynamique du mode de fonctionnement normal $\lambda_0 \in [\lambda_{min}, \lambda_{max}]$. Le fait de supposer que les dynamiques sont positives est un choix résultant

de notre expérience, les processus physiques peuvent s'exécuter plus rapidement ou plus lentement mais rarement en inversant leur sens d'exécution. Prenons l'exemple d'une vanne qui laisse passer un liquide, selon l'ouverture de la vanne le débit peut augmenter ou diminuer. Cependant il n'y a aucune contrainte théorique qui empêche d'avoir des dynamiques négatives.

Remarque 3. La trajectoire passant par les sommets $S_1 \rightarrow S_2 \rightarrow S_4$, n'est pas possible car le fait de rester uniquement dans le sommet S_2 n'amène pas le système à un état défaillant. Par conséquent la transition T_6 ne sera jamais franchie si la transition T_3 n'est pas franchie auparavant. Si une alarme se déclenche, c'est que l'évolution du système est passée forcément par un mode de dysfonctionnement.

C. Espace des états atteignables

Dans cette partie, nous allons décrire la méthode de calcul de l'espace temporel atteignable, qui représente toutes les évolutions possibles su système. Chaque trajectoire de cet espace doit amener le système à atteindre ses objectifs dans les délais requis et sans déclenchement de l'alarme. Pour résoudre se problème nous allons calculer l'espace d'analyse en avant et l'espace d'analyse en arrière.

C1. L'analyse en avant

L'analyse en avant permet de calculer toutes les trajectoires possibles du système à partir d'un état initial, y compris celles qui aboutissent à une alarme. Ceci consiste à calculer les espaces des variables relatives aux séjours du système dans chaque sommet du modèle.

Les variables x et y sont initialisées au départ, l'espace de temps à l'entrée du sommet S_2 est $E_2 = \{x = y = 0\}$, l'évolution de l'état (S_2, E_2) est déterminée en utilisant l'analyse en avant. Soit E'_2 le successeur discret de la région E_2.

On note $E'_2 = Suc_d(E_2)$, avec :

$$Suc_d(E_2) \Rightarrow \quad x_0 \in [0, \beta]$$
$$y_0 = \lambda_0.x_0$$
$$x = x_0 + t$$
$$y = \lambda_i.t + \lambda_0.x_0$$
$$y = \lambda_i.(x - x_0) + \lambda_0.\ x_0$$
$$y = \lambda_i.x + (\lambda_0 - \lambda_i).\ x_0$$

$$Suc_d(E_2) = (0 \le \frac{y_i - \lambda_i x}{\lambda_0 - \lambda_i} \le \beta \wedge y \le \delta \wedge 0 \le x \le \beta)$$

L'espace E_2 résultant de ce calcul est illustré par la figure 3.15. Cet espace regroupe tous les ensembles atteignables des valeurs x et y dans S_2 et S_3. Cela veut dire toutes les trajectoires possibles dans ces sommets. Certaines de ces trajectoires sont acceptables car elles permettent de vérifier la garde $G_{1,3}$, mais ce n'est pas le cas de toutes les trajectoires. Notre objectif est de caractériser l'espace atteignable qui permet au système d'atteindre ces objectifs tout en respectant ses contraintes.

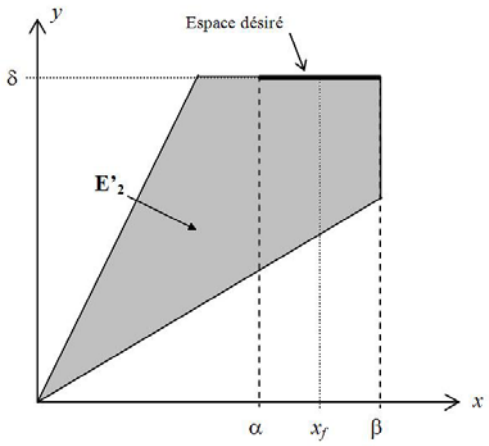


FIG. 3. 15 – Espace d'état calculé par l'analyse en avant (E'_2)

C2. L'analyse en arrière

Pour éliminer les trajectoires qui ne correspondent pas à une exécution correcte, on effectue une analyse arrière en partant de l'espace D désiré qui représente le

comportement du processus exécuté correctement, c'est-à-dire qui vérifie les contraintes finales décrites par les inégalités : $\alpha \le x \le \beta$ et $y = \delta$.

Nous calculons l'espace E''$_2$, il nous permettra d'atteindre cet espace D par la méthode d'analyse arrière. Cela revient à inverser l'automate et à faire une analyse avant.

On a :

$$\alpha \le x_f \le \beta$$

$$\alpha \le x + t \le \beta$$

$$y_f = \lambda_i.t + y = \delta$$

$$\Rightarrow \quad t = \frac{\delta - y}{\lambda_i}$$

$$\Rightarrow \alpha \le x + \frac{\delta - y}{\lambda_i} \le \beta$$

Dans le cas où il y a plusieurs dynamiques λ_i avec lesquelles le système évolue, avec $\lambda_i \in [\lambda_{min}, \lambda_{max}]$, et $\lambda_{max} \ge \lambda_{min} \ge 0$. Pour calculer toutes les trajectoires avec lesquelles on peut réaliser une exécution complète du processus avec la limite de temps inférieure α, la dynamique utilisée est λ_{min}. Pour calculer toutes les trajectoires avec lesquelles on peut réaliser une exécution complète du processus avec la limite de temps supérieure β, la dynamique utilisée est λ_{max}.

Analyse arrière à partir de la limite de temps inférieure α :

$$\alpha \le x + \frac{\delta - y}{\lambda_{min}}$$

$$\alpha.\lambda_{min} \le \lambda_{min}.x + \delta - y$$

$$\Rightarrow \frac{\alpha.\lambda_{min}}{\delta} \le \lambda_{min}.x - y$$

Analyse arrière à partir de la limite de temps supérieure β :

$$x + \frac{\delta - y}{\lambda_{max}} \le \beta$$

$$\Rightarrow \qquad \lambda_{\max}.x - y \le \frac{\beta.\lambda_{\max}}{\delta}$$

L'espace E''$_2$ (Figure 3.16) est décrit par l'inégalité suivante :

$$Pre_t(\mathrm{D}) = (\frac{\alpha.\lambda_{\min}}{\delta} \le \lambda_{\min}.x - y \ \wedge \ \lambda_{\max}.x - y \le \frac{\beta.\lambda_{\max}}{\delta})$$

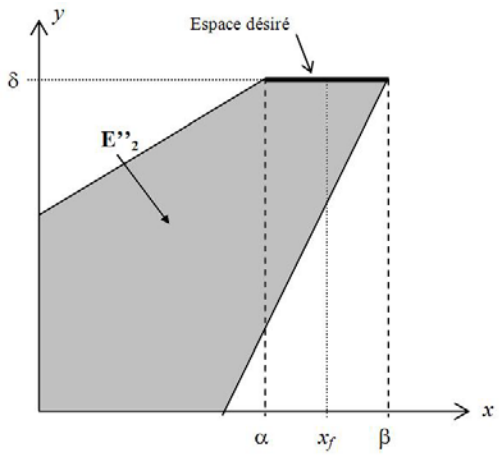


FIG. 3. 16 – Espace d'état calculé par l'analyse en arrière (E'$_2$)

Pour obtenir l'espace caractérisant l'évolution correcte du système, on calcule la région *E* qui est l'intersection des régions *E'$_2$* et *E''$_2$*, cet espace contient toutes les trajectoire possibles qui permettent en processus de s'exécuter correctement en respectant les contraintes physiques imposées par le processus et en tenant compte de toutes les dynamiques avec lesquelles le système peut évoluer. Cet espace est illustré dans Figure 3.17.

$$E = E'_2 \cap E''_2$$

Inévitablement tout écart de l'espace *E*, indiquera à l'opérateur que le système ne pourra pas achever son exécution dans les délais requis et entraînera immédiatement le déclenchement d'une alarme.

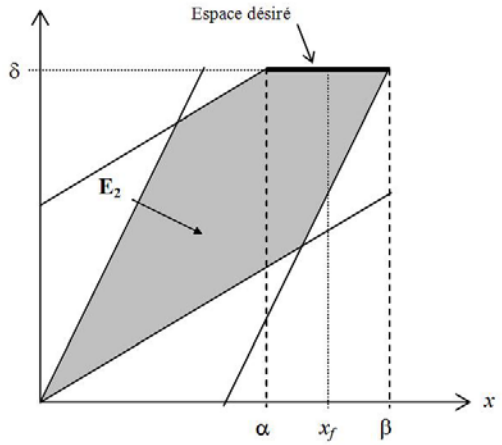


FIG. 3. 17 – Espace des états du processus commandé (E)

3.5. Conclusion

Dans ce chapitre nous avons présenté en premier lieu l'outil de modélisation utilisé dans nos travaux à savoir les automates. Deux classes d'automates ont été présentées ; la première étant celle des automates temporisés à cause de son importante utilisation dans les études de surveillance des systèmes à événements discrets. La deuxième classe qui a été introduite et qui est l'objet de notre contribution dans ce travail de recherche est l'automate hybride linéaire. L'utilisation de cet outil est justifiée par sa flexibilité par rapport à l'automate temporisé ainsi que par l'élargissement du champ de ses composantes.

Garce à l'automate temporisé nous avons calculé un espace d'état ou le système évoluera dans le respect des contraintes du cahier des charges. Nous avons défini un automate représentant le système de surveillance de processus industriels, il comporte à trois états : état initial, état de fonctionnement normal et état de dysfonctionnement, chaque état comportant une dynamique différente. A ce stade des travaux on peut considérer qu'il nous reste encore une étape à franchir. En effet notre objectif final vise à trouver un modèle d'automate pouvant définir n'importe quel processus évoluant avec plusieurs dynamiques, mais ne distinguant aucun mode par rapport a un autre. C'est le but du chapitre suivant, dans lequel nous allons définir cet automate. Nous en profiterons aussi pour élargir notre champ d'application en augmentant la classe de l'automate utilisé à savoir les automates hybrides rectangulaire afin de pouvoir étudier

des processus plus complexes que ceux possibles avec l'automate hybride linéaire. L'analyse d'atteignabilité sera approfondie avec le calcul des espaces d'analyse en avant et d'analyse en arrière. On établira également les espaces d'implémentation de la méthode de surveillance pour chaque sommet.

Chapitre 4
METHODOLOGIE DE SURVEILLANCE DES SYSTEMES DYNAMIQUES HYBRIDES

4.1. Introduction

Dans ce chapitre une méthodologie de surveillance des systèmes dynamiques hybrides sera établie. Cette méthode est basée sur l'utilisation des automates hybrides rectangulaires (AHR). Une approche qui repose sur les automates hybrides linéaire a été introduite dans le chapitre précédent. Ici, cette approche sera développée, généralisée et enrichie par l'utilisation de l'automate hybride rectangulaire. Dans une première partie nous allons présenter l'outil de modélisation utilisé à savoir les AHR. Ensuite nous introduirons la méthode de surveillance en utilisant l'atelier de collage déjà présenté. Par la suite une synthèse de la méthode sera faite et nous terminerons par une application sur deux exemples illustratifs de la méthode de surveillance avec implémentation et simulation.

4.2. Automate hybride rectangulaire

Dans cette partie nous introduisons le modèle automate hybride rectangulaire (*AHR*) (Henzinger et al, 1998), qui représente le cadre de modélisation de l'approche de surveillance que nous allons développer dans ce chapitre. Cet automate peut être considéré comme une généralisation des modèles automate temporisé et automates hybrides linéaire, il est aussi utilisé pour le diagnostic des systèmes hybrides rectangulaires (Derbel, 2009). Ce modèle permet d'approximer les comportements de la dynamique continue du système à travers l'utilisation de conditions de flux rectangulaires de la forme $\dot{x} \in [a, b]$. Dans la suite une présentation formelle de l'automate hybride rectangulaire sera faite, avec ses aspects syntaxiques et sémantiques.

4.2.1. Syntaxe

Soit $X = \{x_1, x_2, \ldots, x_n\}$ un ensemble de variables réelles. Nous notons par $\dot{X} = \{\dot{x} | x \in X\}$ l'ensemble des dérivées premières des variables dans X par rapport au temps.

Définition 4.1.

- Une inégalité rectangulaire sur X est une inégalité de la forme $x \sim c$, où $x \in X$, $c \in \mathbf{Z}$ et $\sim \in \{<, \leq, \geq, >\}$.
- Un prédicat rectangulaire sur X est une conjonction d'inégalités rectangulaires sur X. Notons par $Rect(X)$ l'ensemble des prédicats rectangulaires sur X.

❑

Définition 4.2. Une valuation sur X est un vecteur de $\mathfrak{R}^n$, v = $\{v_1, v_2, ..., v_n\}$, qui définit une valeur $v_i \in \mathfrak{R}$ pour chaque variable $x_i \in$ X.

❑

Définition 4.3. Un Automate Hybride Rectangulaire (Henzinger et al, 1998) est un septuplet H = ($S, X, \Sigma, E, inv, flux, init, M$), où :

- $S = \{s_1,\ldots, s_k\}$ est un ensemble fini de sommets représentant les états discrets du système;
- X est un ensemble fini de variables réelles. L'état continu du système est caractérisé à tout instant par un vecteur v = $\{v_1, v_2, ..., v_n\}$, dans l'espace euclidien $\mathbb{R}^n$, ou v_i correspond à la valeur du vecteur x_i ;
- Σ est un ensemble d'événements ;
- $E \subseteq S \times \Sigma \times Rect(X) \times Rect(X) \times 2^X \times S$ est un ensemble fini de transitions. Une transition (s, σ, g, r, R, s') correspond à un changement de sommet de s à s', sur l'occurrence de l'événement σ et sous la condition $v \in [g]$, où le vecteur v correspond aux valeurs courantes des variables de X. Après le franchissement de la transition, chaque variable $x_i \in R$ est réinitialisée, d'une manière non déterministe, par une valeur de l'intervalle $[r](x)$. Les autres variables qui ne sont pas dans R, restent inchangées.
- $inv : S \rightarrow Rect\ (X)$ est une fonction qui associe à chaque sommet $s \in S$ une contrainte rectangulaire pour chaque variable $x_i \in X$. Le système peut séjourner dans un sommet tant que l'invariant du sommet est satisfait ;

- *flux* : S → *Rect*($\dot{x}$) est la fonction qui affecte à chaque sommet une représentation pour l'évolution continue. Durant le séjour dans un sommet *s* de l'automate, l'évolution des variables continues est exprimée généralement sous la forme d'une équation d'état *flux*(*s*), où pour chaque variable $x_i \in X$, sa première dérivée par rapport au temps $\dot{x}_i$ doit évoluer dans l'intervalle [*flux*(*s*)] ($\dot{x}_i$);
- *init* $\subseteq S \times$ *Rect*(X), désigne la condition initiale de l'automate ;
- $M \subseteq S$ correspond à l'ensemble des sommets marqués de l'automate.

❑

4.2.2. Sémantique

L'état d'un automate hybride rectangulaire est indiqué à chaque instant par une paire (*s*, v) avec *s* l'état discret du système et v un vecteur indiquant la valeur courante de chaque variable.

La sémantique d'un automate hybride rectangulaire est présentée comme étant l'ensemble des comportements qui peuvent êtres produits par le système modélisé. Dans ce cas nous définissons la sémantique d'un système continu par un ensemble d'équations différentielles.

Il y a deux types de transitions qui peuvent faire évoluer un automate hybride rectangulaire :

- Les transitions continues : en franchissant cette transition la partie discrète de l'état reste constante dans le même sommet s, par contre il y a évolution de la partie continue de la valuation v vers la valuation v'. Et ceci grâce à une trajectoire continue, caractérisée par les dynamiques des variables qui évoluent dans les intervalles indiqué par *flux*(*s*), avec le respect des contraintes d'invariants spécifiées par *inv*(*s*).

  La transition continue est notée par (*s*, v) $\xrightarrow{\delta}$ (*s*, v'), où $\delta \in \mathfrak{R}_+$ désigne le temps passé pendant la transition.

- Les transitions discrètes : ils correspondent à une évolution discrète, qui amène l'automate d'un sommet à un autre ; elles sont décrites par un quintuplet $u = (s, \sigma, g, r, R, s') \in E$, l'ensemble des transitions défini précédemment.

  La transition discrète est notée par $(s, v) \xrightarrow{u} (s, v')$.

4.2.3. Analyse d'atteignabilité

L'analyse d'atteignabilité cherche à vérifier si on peut accéder à une région de l'espace d'état à partir d'une région initiale de donnée. Une région d'un AHR est définie comme étant une paire (s, z), où s correspond à un sommet de l'automate et z une région de l'espace d'état continu. Un état (s', v) est inclus dans une région (s, z) si $s = s'$ et $v \in z$. l'état symbolique (s, z) décrit les états $\{(s, v) \mid v \in z\}$. Elle permet de voir l'existence ou non d'une trajectoire qui amène l'état du système à partir d'une région (s_0, z_0) vers une région (s_m, z_m).

Généralement on utilise la méthode d'analyse en avant pour vérifier la l'accessibilité d'une région dans l'espace d'état. Les successeurs d'un état symbolique sont calculés itérativement, en alternant le calcul de successeurs continus et discrets (Roux et Rusu, 1996).

Le calcul du successeur continu d'un état symbolique (s, z) est réalisé en utilisant l'opérateur Suc_t, qu'on définit comme suit :

$$Suc_t((s,z)) = \left\{ (s,v') \middle| (s,v) \xrightarrow{\delta} (s',v'), v \in z, \delta \in R_+ \right\}$$

De même l'opérateur de calcul des successeurs discrets d'un région (s, z) suite au franchissement d'une transition $u \in E$, est noté $Suc_d((s, z), u)$ tel que :

$$Suc_d((s,z),u) = \left\{ (s',v') \middle| (s,v) \xrightarrow{u} (s',v'), v \in z \right\}$$

L'opérateur Suc est défini tel qu'il correspond à la composition de l'opérateur successeur discret, sur le franchissement d'une transition $u \in E$, et l'opérateur successeur continu :

$$Suc((s,z),u) = Suc_t \circ Suc_d((s,z),u)$$

La notion de prédécesseur continu est duale à celle de successeur continu. Tout état *W* définit par (*s*, *z*) depuis lequel on peut atteindre un état donné *R'* exprimé par (s', *z*') en laissant le temps s'écouler tout en restant dans le même sommet est un prédécesseur continu de cet état. De même que la notion de prédécesseur discret est duale à celle de successeur discret.

De la même façon on définit l'opérateur Pré(*s*, *z*) permettant le calcul des états prédécesseurs de l'état (*s*, *z*).

On note Pré(*s'*, *z'*) = (*s*, *z*)

4.3. Surveillance par automate hybride rectangulaire

Notre travail de recherche nous a conduit à présenter plusieurs modèles d'automates hybrides dédiés à la surveillance des systèmes dynamiques évènementiels. Dans un premier temps on eu recours aux automates hybrides linéaires (Karoui et al 2010, a), ensuite et afin d'enrichir notre modèle de surveillance nous proposons un modèle d'automate hybride rectangulaire caractérisant le fonctionnement normal du système [Karoui et al 2010, b]. A partir des ordres reçus par l'unité de commande le système évoluera dans plusieurs modes de fonctionnements différents, la transition entre ces modes est gérée par des gardes dépendant des commandes. Il est important de surveiller les actions du système en observant le déroulement des actions de commande. Un système de surveillance est efficace s'il anticipe le dysfonctionnement, c'est ce que nous appelons détection au plus tôt. Les techniques classiques de suivi des systèmes à événement discret consistent à attendre l'échéance d'une ou plusieurs temporisations, cela correspond à caractériser l'espace des trajectoires atteignable par un hyper cube et constitue une approximation grossière. Dans nos travaux la connaissance de l'espace atteignable d'un AHR nous permettra de déterminer les trajectoires dynamiques exactes du système et de détecter au plus tôt le dysfonctionnement. L'ensemble de ces trajectoires constitue le domaine d'évolution du système, dans lequel tous les critères de l'utilisateur ainsi que les contraintes physiques du système sont respectées.

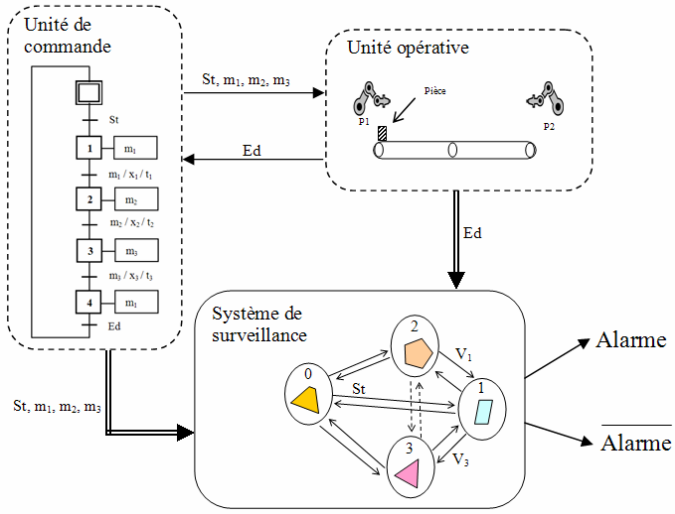


FIG.4. 1 – Modèle de surveillance d'un système réel commandé

Dans ce chapitre nous allons démontrer que la complexité des conditions de commutation entre les différents modes de fonctionnement engendre une diversité des domaines d'évolution (Système de surveillance, Fig. 4.1). Aussi nous démontrerons que plus le fonctionnement est complexe plus le système de surveillance est fin. La méthodologie de surveillance que nous proposons dans ce travail est indépendante du système de commande, le modèle AHR de surveillance va suivre les trajectoires proposées par le module de commande et dés le premier écart par rapport aux trajectoires préétablis il y a déclenchement d'une alarme ; cela sous entend que le modèle de surveillance doit calculer au préalable toutes les trajectoires possibles du système et de ce fait la séquence de commande devient un simple scénario parmi d'autre du système de surveillance.

4.3.1. Spécification de l'exemple

Considérons l'exemple de l'atelier de collage décrit dans le chapitre précédant (Unité Opérative, figure 4.1). Cet atelier est composé d'un poste P1 où il y a un dépôt de colle effectué sur une pièce. Un convoyeur va transporter cette pièce vers le poste P2 où il y aura l'opération de collage. Le convoyeur débute sa tâche suite à l'ordre Start 'St' du contrôleur (après que la colle ait été déposée sur la pièce). L'évènement End 'Ed'

survient quand la pièce atteint le poste P2. Le tapis roulant à deux fonctions ; la première consiste à transporter la pièce de P1 à P2 et la deuxième sert de retardateur.

Dans un premier temps nous allons considérer l'atelier comme un simple système de transport, nous allons établir le mécanisme de surveillance entre le point de départ le poste P1 et le point d'arrivée le poste P2.

Lors du transport de la pièce le convoyeur peut être utilisé en plusieurs modes de fonctionnement, la différence entre eux réside dans la vitesse de convoyage.

Pour présenter notre approche, on construit le modèle de surveillance de l'exemple illustré par l'unité opérative de la figure 4.1. Dés que le contrôleur donne l'ordre *Start* '***St***' deux variables x et y sont initialisés. La variable x mesure le temps écoulé depuis l'instant de la mise en route du convoyeur jusqu'à son arrêt, la dynamique de x prendra comme valeur : $\dot{x} = 1$; Le convoyeur est en marche. La variable y indique la position de la pièce sur le convoyeur, la longueur du trajet est égale à 16m. La dynamique de y peut prendre plusieurs valeurs. En effet le système a été conçu pour une vitesse nominale égale à 2 m/*u.t.* mais il y a d'autres modes (accélération ou décélération). On limite le système à une vitesse maximale égale à 3 m/*u.t.* et à une vitesse minimale égale à 1 m/*u.t.* la valeur de la dynamique de y appartient à l'ensemble {[1, 2[, 2,]2, 3]}, ainsi on aura 4 modes de fonctionnement du système. Le modèle surveillance AHR du convoyeur est illustré dans la figure 4.2.

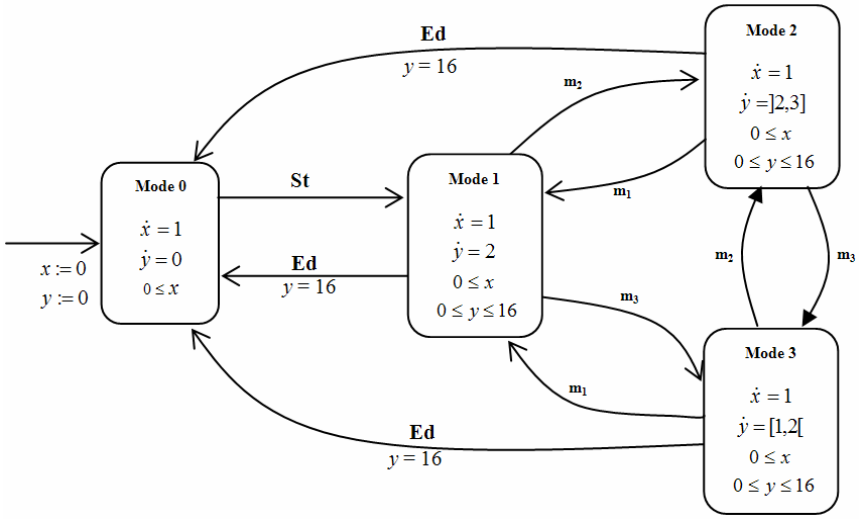


FIG.4. 2 – Modèle de surveillance du convoyeur

Les gardes m_i des transitions entre les modes 1, 2 et 3 sont générées par le calculateur. Chaque garde m_i correspond à une paire (s_i, t_i) où s_i correspond au mode suivant et t_i à la durée que le convoyeur va rester dans ce mode.

Tel qu'il est conçu le système de surveillance ne déclenchera une alarme qu'a la seule condition où le convoyeur est bloqué définitivement et la pièce transportée n'atteindra dans ce cas jamais le poste P2.

Dans certains travaux déjà réalisés [Allahham, 2008], les dynamiques sont données par des horloges qui peuvent être arrêtées. Il a alors été possible de synthétiser un système de surveillance pour des processus interruptibles. Cette surveillance se fait au plutôt grâce à la construction de l'espace atteignable exact.

Afin de rendre notre système de surveillance plus précis, des contraintes peuvent être rajoutées sur les variables qui caractérisent le fonctionnement du convoyeur.

Dans un premier temps, on impose à *x* une borne supérieure et une borne inférieure. Ces limites temporelles peuvent s'expliquer par le fait que la colle disposée sur la pièce ne peut être utilisée que pendant une période de temps bien précise. Avant cette période la colle n'est pas prête pour l'utilisation et au-delà la colle s'est détériorée. La pièce doit atteindre le poste P2 durant l'intervalle [7, 9]. On peut envisager dans un deuxième temps que des contraintes de fonctionnement ont été appliquées sur le convoyeur ; au démarrage tous les modes de fonctionnements sont accessibles sauf le mode 3 qui le sera après 2 *u.t.*, au-delà de 6 *u.t.* le mode 2 ne sera plus accessible. Ces contraintes peuvent être traduites par le fait que le convoyeur ne peut démarrer avec une vitesse faible et ceci pour ne pas entamer sa tache avec un retard, aussi il ne peut pas terminer sa course avec une vitesse très élevée afin de faciliter son arrêt. Le nouveau modèle de surveillance AHR du convoyeur modifié est illustré dans la figure 4.3.

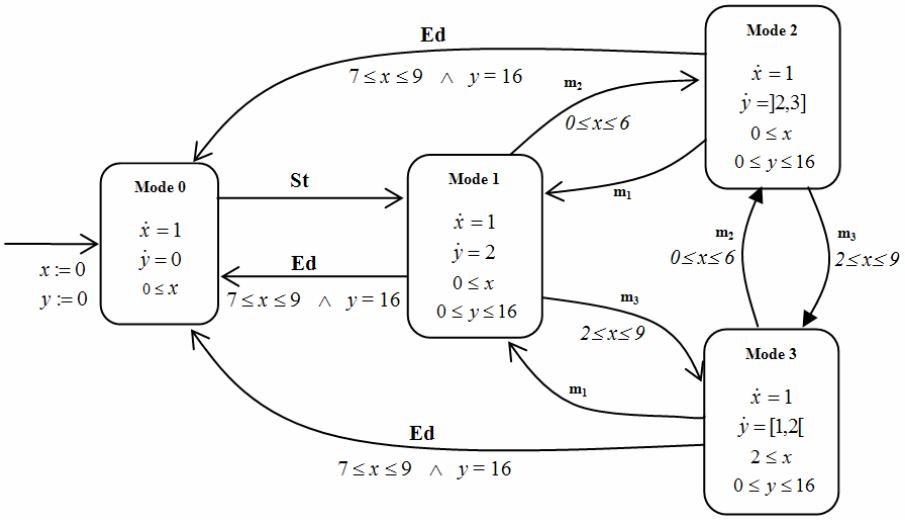


FIG.4. 3 – Modèle de surveillance AHR enrichi

4.3.2. Analyse d'atteignabilité de l'exemple

A. Analyse en avant

Rappelons que l'analyse en avant est basée sur le calcul de l'espace de tous les états qui peuvent êtres atteints depuis des états appartenant à R_0. L'ensemble de ces états est appelé successeur de la région R_0. Si l'espace calculé contient des états qui appartiennent également à la région R, alors on peut conclure que cette région est atteignable depuis R_0.

Le calcul d'une région (s', z') successeur d'une région (s, z) suite a une transition u = (s, σ, g, r, R, s'), est réalisée en appliquant les étapes suivantes (Guéguen et zaytoon, 2004) :

1. Calculer z_1 l'espace correspondant à l'intersection de l'invariant du sommet s et le futur de la région z. le futur d'une région correspond à l'ensemble des états accessibles à partir de la région de départ en appliquant la fonction de flux.
2. Calculer z_2 : c'est l'intersection de z_1 avec la condition de garde g ;
3. Calculer z_3 en appliquant la fonction de réinitialisation définie par R et r sur la région z_2 ;
4. Calculer z_4 : c'est l'intersection de la région z_3 et l'invariant du sommet s' ;

5. Enfin calculer *z*' l'espace correspondant à l'intersection de l'invariant du sommet s' et le futur de la région z_4.

Soit la région *z* du sommet s_1 tel que $z = (3 \leq x \leq 4 \wedge 4 \leq y \leq 6)$. Le successeur continu de la région *z* est donné par la figure 4.4.a, ce successeur est calculé en laissant le temps s'écouler les variables *x* et *y* évoluent selon le flux du sommet s_1, $flux(s_1) = (\dot{x} = 1 \wedge \dot{y} = 2)$. L'espace illustré dans la figure 4.4.a représente toutes les régions accessibles à partir de la région *z* avec une évolution dans le sommet s_1 et uniquement avec les dynamiques de celui-ci. En appliquant les règles de calcul citées précédemment à cette région nous trouvons la région *z*' successeur de la région *z* illustrée par la figure 4.4.b. Cette figure représente toutes les régions accessibles à partir de la région z et ce la en utilisant les dynamiques de tous les sommets atteignables par le sommet s_1. Ces dynamiques sont : $flux(s_1) = (\dot{x} = 1 \wedge \dot{y} = 2)$ et $flux(s_2) = (\dot{x} = 1 \wedge \dot{y} \in]2,3])$ et $flux(s_3) = (\dot{x} = 1 \wedge \dot{y} \in [1,2[)$.

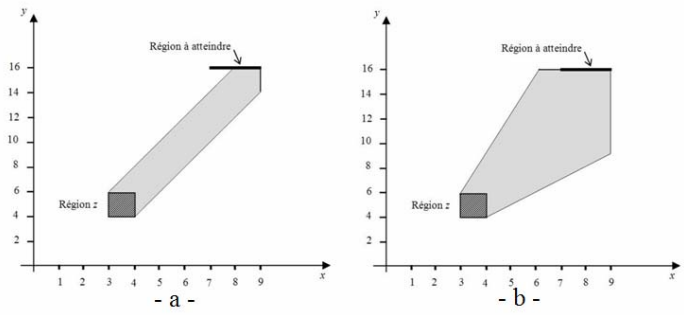


FIG.4. 4 – a – successeur continu de la région *z*, b – successeur de la région *z*

Pour alléger la présentation des régions, seul l'espace atteignable du mode de fonctionnement 1 de l'atelier de collage sera présenté, une étude analogue sur les autres modes de fonctionnement est faite. Le calcul des régions est fait en utilisant le logiciel PHAVer qui repose sur des techniques de vérification formelle qui sont détaillées dans (Frehse, 2005) et (Asarin et al., 2006).

L'ensemble des trajectoires pouvant êtres calculées à partir de l'état initial est représenté dans la figure 4.5.a, on remarquera que certaines trajectoires n'acheminent pas la pièce durant l'intervalle de temps prédéfini ou l'acheminent mais en dehors de la durée souhaitée (Fig. 4.5.b).

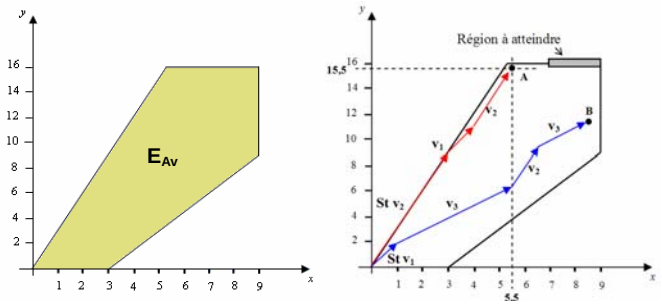


FIG.4. 5 – a - Espace de l'analyse avant, b - Exemple de trajectoires.

Dans l'analyse avant, certaines trajectoires aboutissent en des points où une transition n'est plus franchissable et indéfiniment non franchissable. La trajectoire 1, définie par l'ensemble des commandes suivante :{*St*, v_2, v_1, v_2}, aboutie en un point A où $x = 5{,}5$ et $y = 15{,}5$. A partir de ce point la transition associée à l'événement *Ed* n'est plus jamais franchissable car la garde $x \leq 7$ et $y = 16$ ne sera plus jamais atteinte. Il en sera de même pour la deuxième trajectoire aboutissant vers le point B de coordonnées (8.5, 11.5).

Quand la transition *Ed* est indéfiniment non franchissable le système se divise en sous-systèmes dans lesquelles les invariants sont respectés, mais il y a des transitions qui ne peuvent plus êtres franchies. La figure 4.6 illustre cet état avec une évolution du système dans un ensemble composé de 3 modes de fonctionnement différents. A partir de cet ensemble les modes 0 n'est plus atteignable.

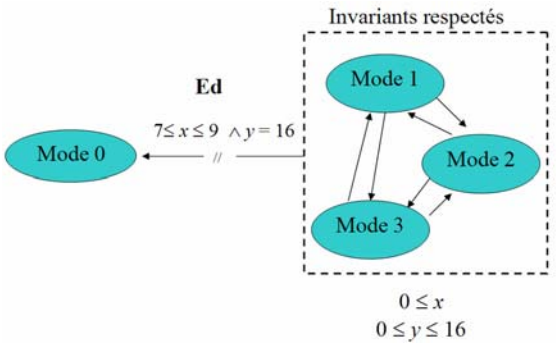


FIG.4. 6 – Modèle de surveillance du convoyeur

B. Analyse en arrière

On rappelle aussi que cette méthode est basée sur le calcul de l'ensemble de tous les états à partir desquels on peut atteindre des états de la région R. Si l'espace calculé contient des états qui appartiennent également à la région R_0, alors on peut conclure que la région R est atteignable depuis R_0. Cette méthode est duale à l'analyse en avant et est appelée méthode d'analyse en arrière. L'espace déterminé par l'analyse arrière nous permet de connaître tous les états à partir desquels on peut atteindre l'état final. Certaines zones appartenant à l'espace calculé par l'analyse arrière ne sont pas atteignables par le convoyeur dont le point C (Fig. 4.7.b). Ce point de coordonnées (1, 8) ne fait pas parti de l'espace des états calculés par l'analyse en avant, et donc le convoyeur ne pourra pas atteindre le point C.

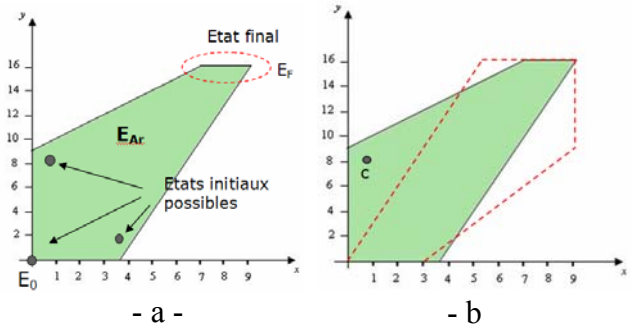
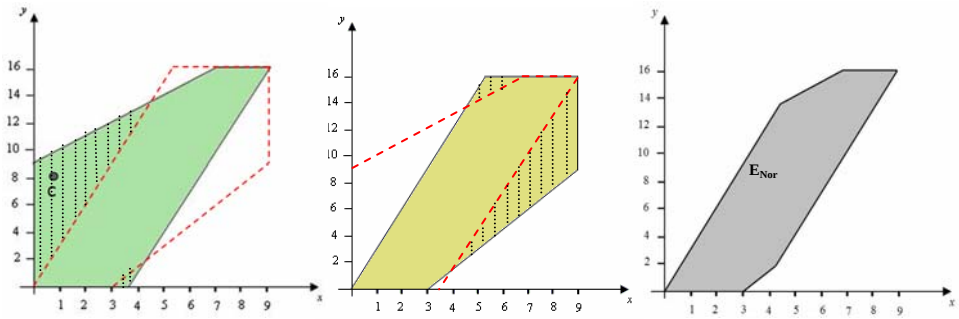


FIG.4. 7 – a - Espace de l'analyse arrière, b - Espace des états initiaux non accessibles.

C. Intersection des espaces

Nous allons ici déterminer l'espace du fonctionnement normal, espace qui garantira; qu'à partir de n'importe quel état appartenant à cet espace on peut atteindre l'état final tout en respectant les contraintes imposées.

Cet espace que nous notons E_{Nor} est défini tel que chaque région de cet espace est à la fois un successeur et un prédécesseur d'une autre région appartenant à cet espace. L'espace E_{Nor} calculé par la relation $E_{Nor} = E_{AV} \cap E_{AR}$ satisfait cette condition (Fig 4.8.c).



a b c

FIG.4. 8 – a - Etats non accessibles par l'état initial, b - Etats ne conduisant pas vers l'état final, c - Espace du fonctionnement normal.

L'intersection entre l'espace calculé par l'analyse en avant et celui calculé par l'analyse arrière permet d'une part d'éliminer les états qui ne seront jamais atteints par le convoyeur. En tenant compte des diverses dynamiques que le processus peut utiliser il y a certaines régions qui ne seront pas accessibles par le processus (Fig. 4.8.a). D'autre part cette intersection permet aussi d'écarter toutes les régions qui à partir desquelles l'état final n'est jamais atteint (Fig. 4.8.b).

4.4. Synthèse de la méthodologie de surveillance

4.4.1. Modélisation du processus

La structure générale d'un modèle de surveillance est un automate hybride rectangulaire donné dans la figure 4.9. Dans cette figure les modes correspondent au fonctionnement normal. Il y a une garde entre chaque sommet de fonctionnement normal vers le sommet de défaillance, cette garde est donnée par l'expression : $\forall i \in [1,n]\ Gd_i = I_i$, avec I_i l'invariant relatif à chaque sommet i.

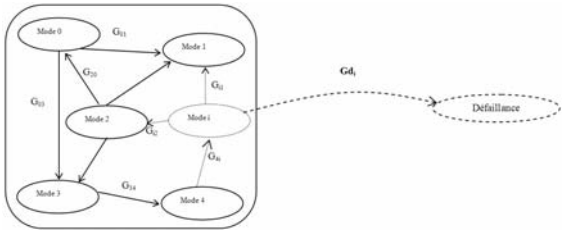


FIG.4. 9 – Structure générale d'un modèle de surveillance

Définition 4.4 : [Cassandras, 2008] le système est vivant s'il peut toujours exécuter un évènement. Quand il y a un ensemble d'états non marqués qui forment un groupe fortement connexe (i.e. ces états sont atteignable entre eux) et il n'y a aucune transition permettant de sortir de cet ensemble, si l'état du système appartient à cet ensemble d'états alors nous avons une situation d'interblocage.

❑

Définition 4.5 : Un modèle de surveillance est un système de surveillance s'il est vivant.

❑

Soit E_{Av} l'espace des états calculés par la méthode de l'analyse avant, cette région est définie par les paires $\langle s_A, z_A \rangle$. Et soit E_{Nor} l'espace d'états caractérisant le fonctionnement normal, dans cet espace le modèle AHR est vivant. Comme l'espace E_{Av} caractérise toutes les trajectoires possibles à partir de l'état initial $\langle s_0, z_0 \rangle$, on peut écrire la relation suivante : $E_{Nor} \subseteq E_{Av}$, qui est illustrée par la figure 4.10.

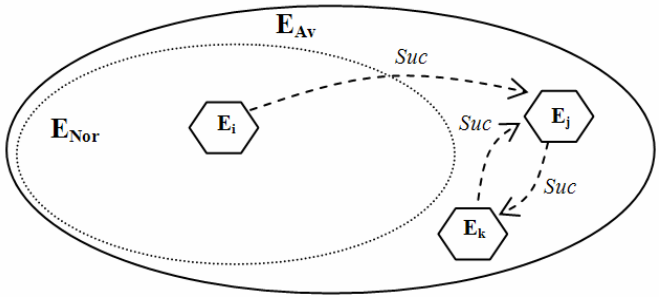


FIG.4. 10 – Espace d'état par l'analyse avant et espace d'état du fonctionnement normal

Soit l'hypothèse suivante : Il existe une région $E_i \in \{E_{Nor}, E_{Av}\}$ tel que $Suc(E_i) = E_j \notin E_{Nor}$, c'est-à-dire que la région E_{Nor} n'inclut pas un des successeurs de la région E_i. Le successeur de la région E_j sera la région E_k et le successeur de celle-ci sera E_j. On se retrouve dans un système interbloqué dans plusieurs états seulement. Selon cette hypothèse il n'existe pas de trajectoire amenant le système de l'état E_j à un

état du fonctionnement normal E_{Nor}. De ce fait le système n'est pas considéré comme un système vivant.

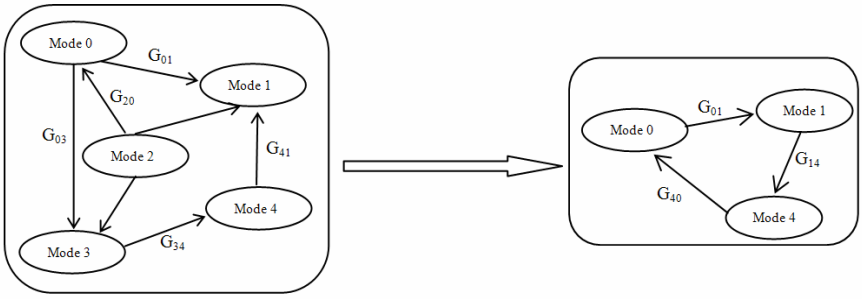


FIG.4. 11 – Interblocage de l'AHR dans un sous-espace

⇒ L'espace des états calculés par l'analyse avant n'est pas suffisant pour caractériser l'espace des états du fonctionnement normale. Le modèle de surveillance AHR ne constitue pas un système vivant, tôt ou tard le système entrera dans un sous espace restreint ou il ne pourra plus en sortir (FIG 4.11).

Dans ce sous-espace Il y a des transitions qui ne peuvent plus êtres franchies, et donc un comportement différent du fonctionnement normal.

L'espace des états calculé par la méthode de l'analyse arrière, est déterminé à partir d'un état final appartenant au fonctionnement normal et en inversant les arcs des transitions du modèle AHR.

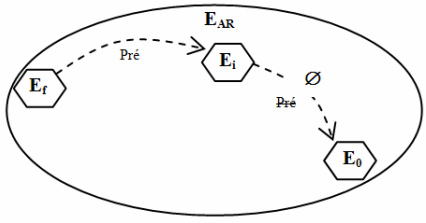


FIG.4. 12 – Espace des états à partir de l'analyse arrière

Soit la région E_f définit par $\langle s_f, z_f \rangle$ (Fig. 4.12); état final du fonctionnement normal, les prédécesseurs de $E_f \in E_{AR}$ (Espace des états calculés par l'analyse arrière). Hypothèse : soit la région E_i un des prédécesseurs de E_f et l'espace de départ initialisé (s_0, z_0) du fonctionnement normale E_0 n'est pas un prédécesseur de E_i. On peut conclure

sur cette hypothèse que l'espace obtenu par l'analyse arrière ne suffit pas à définir l'espace du fonctionnement normal.

L'espace des états du fonctionnement normal est défini tel que chaque région de cet espace est à la fois un successeur et un prédécesseur d'une autre région appartenant à cet espace. L'espace E_{Nor} calculé par la relation $E_{Nor} = E_{AV} \cap E_{AR}$ satisfait cette condition.

Propriété 4.1 : Un AHR est vivant si l'espace d'analyse avant est égale à l'espace d'analyse arrière.

❑

Propriété 4.2 : Un système de surveillance est obtenu à partir du modèle de surveillance en remplaçant tous les invariants de départ par les espaces trouvés en faisant l'intersection des espaces de l'analyse avant et de l'analyse arrière.

❑

Justification intuitive : Toute région $(s_i, z_i) \in E_{Nor}$ est à la fois un successeur de (s_0, z_0) et un prédécesseur de (s_f, z_f). Il existe toujours une trajectoire L pouvant amener le système à n'importe quel état (s_i, z_i) à partir de l'état initial (s_0, z_0).

L'ensemble T des transitions et l'ensemble L finie de trajectoires amenant d'un état initial (s_0, z_0) à tout état (s_i, z_i) appartenant à l'espace E_{Nor} est un système vivant.

4.5. Exemple illustratif 1

4.5.1. Spécification de l'exemple

Afin de montrer la pertinence de notre approche, nous considérons un système de transmission de données en isolation. La fonction de ce système est d'envoyer les données rassemblées par celui-ci. L'envoie des données est tributaire de l'énergie consommée ainsi que du débit d'émission. Le mécanisme de transmission peut adopter deux états de fonctionnement ; l'état « marche » ou « arrêt ». Durant la phase arrêt le système se met en mode de chargement et pendant la phase émission il est en mode

consommation d'énergie. Le modèle AHR du système est développé dans l'automate *A* dans la figure 4.13.

Pour caractériser le fonctionnement du système, on utilise trois variables *h*, *x* et *y*. La variable *h* représente le temps d'émission, en effet nous considérons que l'émission des données est limitée dans le temps. La variable *x* représente la réserve d'énergie contenue dans le système, pour émettre convenablement les données, le système doit avoir un stock d'énergie compris entre x_{min} et x_{max}. Les valeurs de cet intervalle sont déduites des contraintes physiques du système. x_{min} représente la valeur minimale de l'énergie en dessous de laquelle le système serait incapable de fonctionner. x_{max} indique la limite maximale d'énergie que le système pourrait emmagasiner. La variable *y* exprime la quantité de données émise.

Les dynamiques respectives aux variables *h*, *x* et *y* sont :

- $\dot{h}$: elle représente l'état de fonctionnement du système « marche » ou « arrêt ».
- $\dot{x}$: représente le taux de consommation d'énergie ; nous adoptons une dynamique positive lors de la phase arrêt du système et des dynamiques négatives pour refléter la diminution de la quantité d'énergie au cours du fonctionnement.
- $\dot{y}$: représente le débit d'émission des données.

4.5.2. Construction du modèle de surveillance

Le système de transmission peut évoluer en 5 modes de fonctionnement. Initialement le système en mode 0 est en chargement, à partir de là il peut évoluer soit dans le mode1 lorsque la quantité d'énergie emmagasinée atteint la limite supérieure autorisée, soit vers les modes d'émission (mode 2, mode 3 et mode 4). La fonction assignée à ce système est d'émettre une série de données d'une taille $y = 20$ en une durée ne dépassant pas $h = 10$ u.t.

Le mode 0 correspond à l'état de repos du système, dans ce mode de fonctionnement il n'y a en effet ni émission de données ni consommation d'énergie, bien au contraire dans ce type de fonctionnement le système se recharge d'énergie ; nous faisons l'hypothèse que le système ne peut à la fois consommer et emmagasiner de l'énergie en même temps. À partir de ce mode deux possibilités d'évolutions s'offrent au système ;

- la première est que la capacité de stockage de l'énergie a atteint sa limite et que l'ordre *St* de l'opérateur n'est pas encore arrivé, alors le système se met en mode 1 en attente par l'exécution de la garde Of ;

- La deuxième possibilité correspond à l'avènement de l'ordre *St*, le système peut ainsi débuter sa fonction d'émission des données.

Cette fonction d'émission est réalisée à travers plusieurs modes de fonctionnement :

- Mode 2 : relatif au sommet 2, dans ce mode le système évolue avec les flux suivants $flux(S_2) = (\dot{x} \in [-4,-3] \wedge \dot{y} = 2)$, les invariants de ce sommet sont $inv(S_2) = (0 \leq h \leq 10 \wedge 5 \leq x \leq 50 \wedge 0 \leq y \leq 50)$;

- Mode 3 : relatif au sommet 3, dans ce mode le système évolue avec les flux suivants $flux(S_3) = (\dot{x} \in [-12,-11] \wedge \dot{y} = 4)$, les invariants de ce sommet sont $inv(S_3) = (0 \leq h \leq 10 \wedge 5 \leq x \leq 50 \wedge 0 \leq y \leq 50)$. On remarque que plus le débit d'émission des données et grand plus la consommation d'énergie est grande ;

- Mode 4 : relatif au sommet 4, dans ce mode le système évolue avec les flux suivants $flux(S_4) = (\dot{x} \in [-2,-1] \wedge \dot{y} = 1)$, les invariants de ce sommet sont $inv(S_4) = (0 \leq h \leq 10 \wedge 5 \leq x \leq 50 \wedge 0 \leq y \leq 50)$. Pareillement on remarque que plus le débit d'émission des données et faible, plus la consommation d'énergie est faible aussi.

La problématique de ce système de surveillance est de pouvoir envoyer toutes les données à temps et avec la quantité d'énergie disponible. Le système de surveillance est modélisé par l'automate *A* de la figure 4.13 nous permet de signaler au plutôt si la transmission va s'effectuer dans les délais indiqués.

Les garde G_i, avec $i \in \{1, 2, 3, 4, 5\}$ amène le système à un mode défaillant. Le système évolue dans ce mode si :

- *h* atteint la valeur limite 10 *u.t.* sans que la totalité des données ne soit envoyé.

- Il ne reste plus assez d'énergie pour envoyer le reste des données.

Dés que le système évolue dans le mode défaillant une alarme se déclenche indiquant à l'opérateur que le système a failli dans la réalisation de sa tâche.

Remarque : les invariants des sommets 2, 3 et 4 sont les mêmes, ce qui implique que les espaces atteignables trouvés pour les modes 2, 3 et 5 sont identiques.

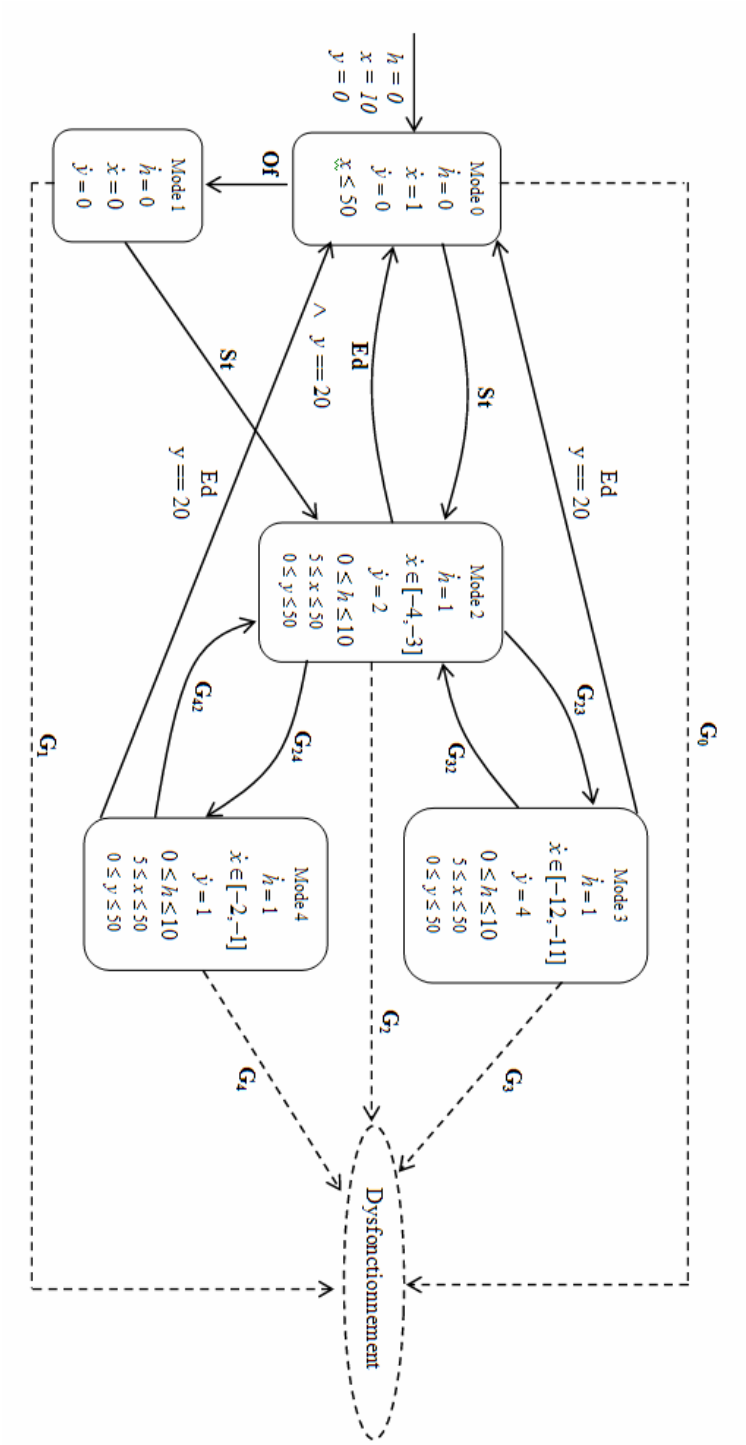


FIG.4. 13 – Modèle *A*HR du système de transmission

Sommet	Espace atteignable résultant de l'analyse en avant de l'automate	Espace atteignable résultant de l'analyse en arrière de l'automate	Espace résultant de l'intersection
S_0	$h = y = = 0 \wedge 0 \leq x$	$h = y == 0$ $\wedge$ -50 $\leq$ - x	$h = y = = 0 \wedge 0 \leq x$ $\wedge$ -50 $\leq$ - x
S_1	$h = y = x == 0$	$h = y = x == 0$	$h = y = x == 0$
S_2, S_3, S_4	$5 \leq x \wedge -10 \leq -h$ $\wedge 150 \leq -4h + 3x + 10y$ $\wedge 0 \leq -h + y$ $\wedge 0 \leq 4h - y$ $\wedge -50 \leq h - x - 2y$ $\wedge 50 \leq 5h - x - 4y$	$25 \leq x + y$ $\wedge 35 \leq -5h + x + 4y$ $\wedge -20 \leq -4h + y$ $\wedge 35 \leq -h + x + 2y$ $\wedge -50 \leq -x$ $\wedge -20 \leq -y$	$150 \leq -4h + 3x + 10y$ $\wedge 25 \leq x + y \wedge 0 \leq -h + y$ $\wedge -20 \leq -y$ $\wedge -50 \leq h - x - 2y$ $\wedge 0 \leq 4h - y \wedge -20 \leq -4h + y$ $\wedge -50 \leq 5h - x - 4y$ $\wedge 35 \leq -5h + x + 4y$ $\wedge 35 \leq -h + x + 2y$

Tab. 4.1. L'espace temporel dans chaque sommet de l'automate *A* résultant de l'analyse en avant, l'analyse en arrière et l'intersection de ces espaces.

4.5.3. Implémentation du système de surveillance

Considérons l'exemple de la trajectoire 1 défini par les commandes suivantes (St, C_{23}, C_{34}, C_{42}) et illustré dans la figure 4.14.a, l'émission des données à été effectuée sans violer les invariants. Les commandes C_{ij} indiquent la commutation de l'automate du sommet i vers le sommet j. la commande ***St*** indique que le système démarre avec le mode 2 et reste dans celui-ci pendant 3 *u.t.* ensuite il y a commutation vers les modes 3 pendant 2 u.t. puis vers le mode 4 pendant 1 u.t. puis retour vers le mode de fonctionnement 2 et c'est dans ce mode que le système termine l'envoi des donnée avec h = 8,5 u.t.

Par contre la trajectoire 2 (Fig. 4.14.b) définit par la séquence de commande (St, C23). Au départ le système commute suite à la commande St vers le mode 2 et y reste pendant 3 u.t. ensuite il commute vers le mode 3. Dans ce mode la trajectoire du système coupe les frontière de l'espace temporel définit par x, y et Z au point A de coordonnées (5.8, 7), Ce qui amène à un déclenchement d'alarme en ce point.

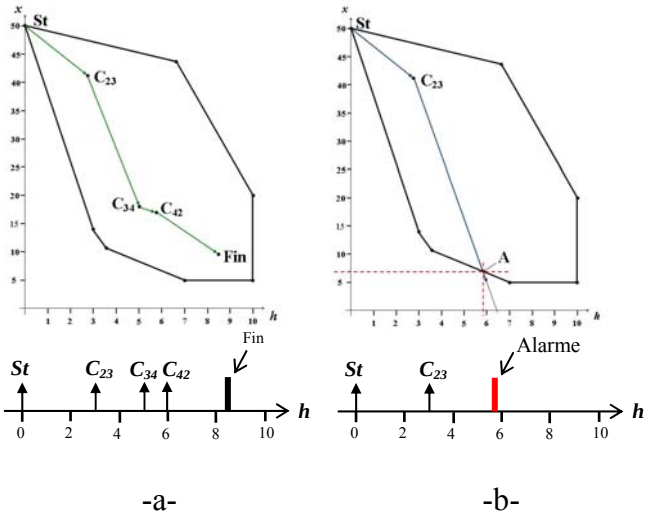


FIG.4. 14 – Exemple de fonctionnement : a - Normale, b - Dysfonctionnement

En utilisant les techniques de surveillance traditionnelles, l'alarme aurait été déclenchée pour h = 6 u.t. La méthode que nous proposons a permis dans ce cas un gain de temps de 0,2 u.t. Les espaces atteignables des variables *x* et *y* sont représentés respectivement dans la figure 4.15.a. et 4.15.b. Ces espaces ont été calculés par logiciel PHAVER [Frehse, 2005].

Toute trajectoire comprise dans ces espaces satisfait les contraintes d'émission des données et toute trajectoire n'appartenant pas aux espaces atteignables de *x* et *y* est une trajectoire qui amène le système à un dysfonctionnement.

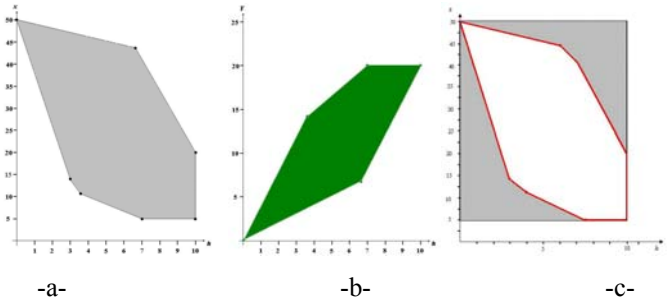


FIG.4. 15 – Espace des états atteignables : a- Paramètre *x* en fonction de *h*, b- Paramètre *y* en fonction de *h*, c- superposition des deux espaces du paramètre *x*

Dans la figure 4.15.c nous superposons deux espaces d'état ; le premier, celui de la figure 4.15.a : espace des états atteignables de la variable *x* et le deuxième (cadre)

représentant l'espace totale d'évolution de la variable *x*. Cette figure illustre bien la pertinence de notre approche. En effet l'espace que nous calculons et bien inférieure à celui établi par des systèmes de surveillance déjà en place.

4.6. Exemple illustratif 2

Reprenons l'exemple de l'atelier de collage qui a été modélisé au paravent dans ce chapitre par l'automate illustré dans la figure 4.3. Contrairement à l'exemple du système d'émission autonome, nous avons varié les invariants entre les différents sommets en incluant des conditions de commutation sur les transitions de l'automate. Ce qui engendre une différence entre les espaces temporels de fonctionnement acceptable du système dans chaque sommet de l'automate. Le système de surveillance est obtenu à partir du modèle de surveillance en remplaçant tous les invariants de départ par les espaces trouvés en faisant l'intersection des espaces de l'analyse avant et de l'analyse arrière.

Sommet	Espace atteignable résultant de l'analyse en avant de l'automate	Espace atteignable résultant de l'analyse en arrière de l'automate	Espace résultant de l'intersection
S_0	$y == 0 \wedge x \geq 0$	$y == 0 \wedge -x \geq -3$	$y = 0 \wedge x \geq 0 \wedge -x \geq -3$
S_1	$3x - y \geq 0 \wedge -x \geq 9$ $\wedge -y \geq -16$ $\wedge -x + y \geq -3$ $\wedge y \geq 0$	$-3x + y \geq -1 \wedge x \geq 0$ $\wedge -y \geq -16$ $\wedge x - y \geq -9$ $\wedge 2x - y \geq -7$	$x - y \geq -9 \wedge -x + y \geq -3$ $\wedge -y \geq -16 \wedge y \geq 0$ $\wedge 3x - y \geq 0 \wedge 3x + y \geq 10$
S_2	$3x - y \geq 0 \wedge -x \geq -6$ $\wedge -y \geq -16$ $\wedge -x + y \geq -3$ $\wedge y \geq 0$	$-3x + y \geq -1 \wedge x \geq 0$ $\wedge -x \geq -6$ $\wedge x - y \geq -9$ $\wedge 2x - y \geq -7$	$x - y \geq -9 \wedge -x + y \geq -3$ $\wedge -x \geq -6 \wedge y \geq 0$ $\wedge 3x - y \geq 0 \wedge 3x + y \geq 10$
S_3	$3x - y \geq 0 \wedge x \geq 2$ $\wedge -y \geq -16$ $\wedge -x + y \geq -3$ $\wedge -x \geq -9 \wedge y \geq 0$	$-3x + y \geq -1 \wedge x \geq 2$ $\wedge -y \geq -16$ $\wedge x - y \geq -9$	$x - y \geq -9 \wedge -x + y \geq -3$ $\wedge -y \geq -16 \wedge y \geq 0$ $\wedge 3x - y \geq 0 \wedge x \geq 2$ $\wedge 3x + y \geq 10$

Tab. 4.2. L'espace temporel dans chaque sommet de l'automate de l'atelier de collage résultant de l'analyse en avant, l'analyse en arrière et l'intersection de ces espaces.

Les espaces atteignables des modes de fonctionnements qui ont été définis par les inégalités dans le tableau 4.2. sont illustrés dans la figure 4.16.

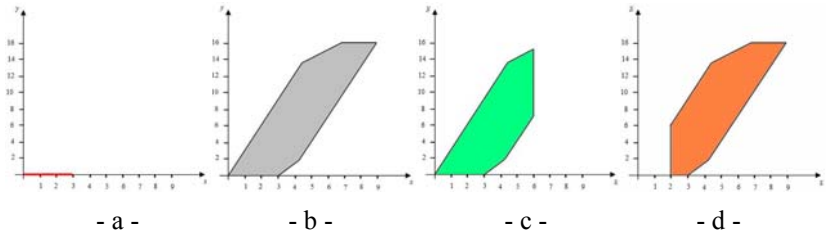


FIG.4. 16 – Espvace des états atteignables : a- mode 0, b- mode 1, c- mode 2, d- mode 3

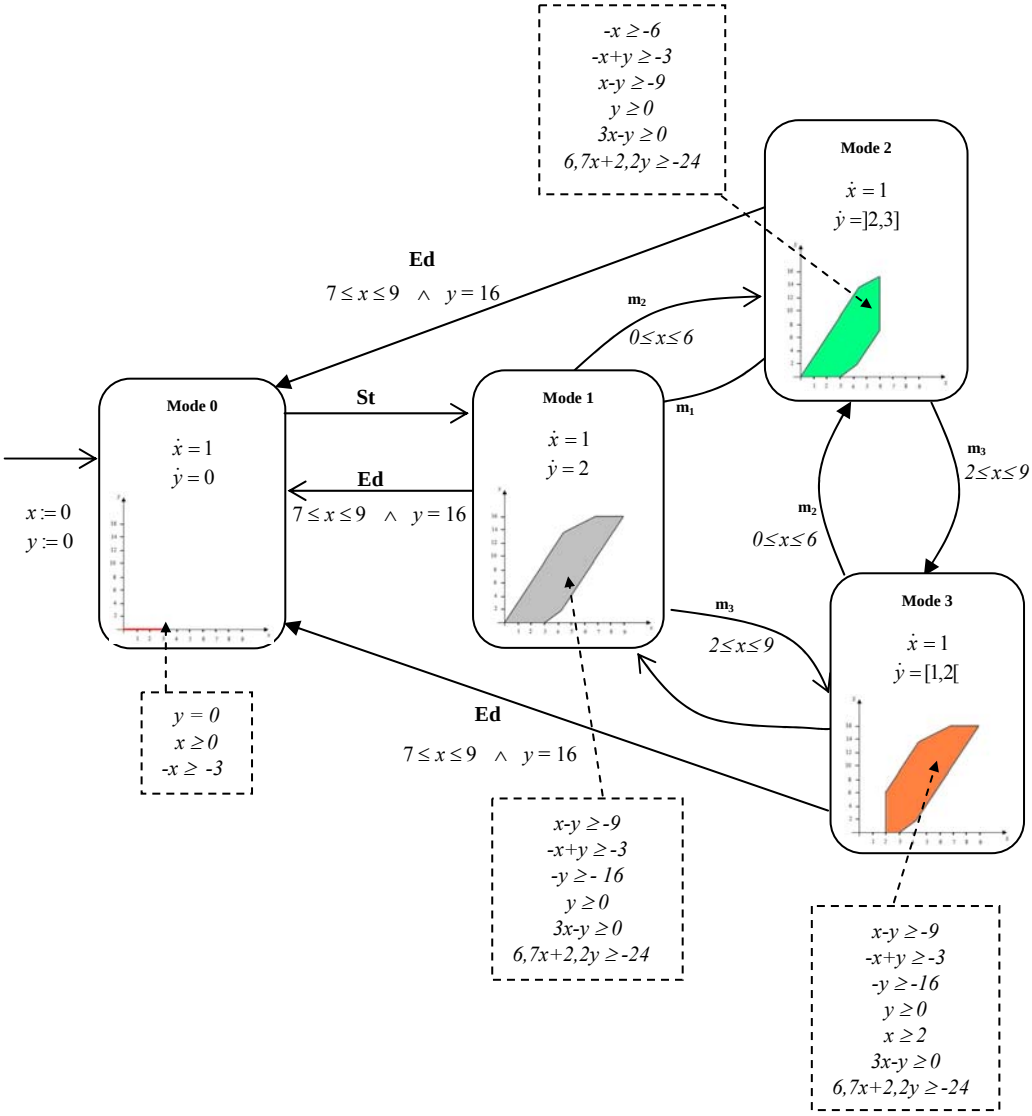


FIG.4. 17 – Modèle implémenté de l'automate de l'atelier de collage

Les invariants de chaque sommet du model AHR sont remplacés par les espaces déterminées par l'intersection des espaces des méthodes de l'analyse en avant et celle de l'analyse en arrière. Le modèle implémenté de l'automate de l'atelier de collage est illustré par la figure 4.17

La figure 4.18 illustre le changement introduit sur le mode de fonctionnement 2.

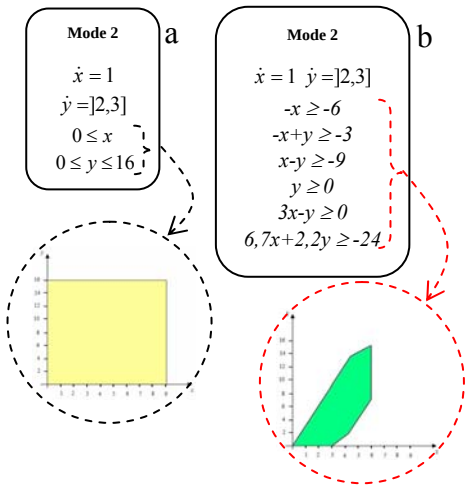


FIG.4. 18 – a - Modèle de surveillance du mode de fonctionnement 2, b - Modèle de surveillance implémenté du mode 2.

La figure 4.18 montre ainsi l'apport de notre méthodologie de surveillance qui consiste à restreindre la zone de fonctionnement dans les différents modes. En effet un écart par rapport au fonctionnement nominal est détecté dès qu'une frontière de cet espace est franchie.

4.7. Conclusion

Dans ce chapitre nous avons proposé un système de surveillance d'un système commandé, le modèle de surveillance de ce système est un automate hybride rectangulaire. Il tient compte des changements dynamiques qui peuvent apparaître au cours de l'exécution du processus tout en gardant une dominante événementielle. Les sommets de l'automate représentent les différentes dynamiques que peut avoir le système commandé, le passage entre les modes de fonctionnent est synchronisé par des évènements liés à ces diverses dynamiques.

Le comportement autorisé du système est contrôlé par des variables sur lesquelles sont appliquées des contraintes, ces contraintes exprimées par des inégalités définissent l'espace acceptable d'évolution du système commandé. Toute violation de cet espace acceptable déclenche une alarme, qui indique qu'il n'y a plus de trajectoire pouvant amener le système à réaliser sa tache en respectant les contraintes. La contribution majeure de nos travaux consiste au signalement de l'apparition d'une défaillance au plus tôt par rapport aux autres techniques de surveillance. Ceci a été possible avec la restriction de l'espace acceptable de fonctionnement du système étudié.

CONCLUSION GENERALE

L'objectif de cette thèse est la mise en œuvre d'une méthodologie de surveillance pour les systèmes dynamiques hybrides. Dans l'élaboration de nos travaux, nous nous sommes basés sur un ensemble de travaux de référence (Sampath et al., 1995), (Allahham et al., 2008).

La méthode de surveillance proposée a pour objectif de détecter au plus tôt les défauts qui peuvent affecter les systèmes dynamiques évènementiels. La détermination du système de surveillance à tenu compte des propriétés décrites ci-dessous.

Dans un premier temps, nous avons considéré les spécifications temporelles associés aux évènements qui se produisent dans un système dynamiques hybrides. L'apparition d'un évènement fait basculer le système dans plusieurs modes de fonctionnements différents, la différence entre eux, réside dans la dynamique d'exécution du système. Cette dynamique est observable, c'est-à-dire que le basculement entre les différents modes est observable. Dans notre démarche nous avons dans une première étape différencié entre le mode de fonctionnement normal qui amène le système à réaliser sa tâche dans les délais requis, des autres modes de fonctionnement. Nous avons ensuite généralisé l'approche en considérons ce mode normal comme un mode de fonctionnement parmi d'autres.

Le système de surveillance développé exploite d'une part les contraintes temporelles appliquées au système dynamique, et d'autre part, il utilise les techniques d'analyse d'atteignabilité de l'automate hybride modélisant le comportement du système. En effet l'outil de modélisation choisi pour l'établissement de la méthode de surveillance est l'automate hybride, ce choix a été dicté par la capacité d'analyse formelle qu'offre cet outil. Deux sous classes d'automate hybride ont été utilisées ; l'automate hybride linéaire pour introduire la démarche de surveillance et par la suite l'automate hybride rectangulaire pour généraliser la méthode et la rendre applicable à des systèmes plus complexes.

Le modèle du système hybride est obtenu par la composition des différents modes de fonctionnement qui le constitue. Chaque sommet de l'automate modélise un mode de fonctionnement particulier. Une procédure de synthèse est appliquée à l'automate afin de délimiter l'espace temporel relatif à chaque sommet. Cet espace défini sous la forme d'inéquations algébriques conditionne l'évolution des paramètres qui caractérisent le système. Tant que les valeurs des paramètres satisfont les contraintes, le système peut évoluer dans ce sommet sans craindre la violation du cahier de charges, par contre tout écart de l'un des paramètres du domaine établi conduit le système à une défaillance. Le basculement d'un mode de fonctionnement à un autre est associé à l'occurrence d'un évènement par le franchissement d'une transition, ce franchissement n'est autorisé que si les valeurs des paramètres vérifient l'espace temporel associé au sommet source. Une action de l'automate correspond à un évènement dans le système à surveiller. Elle peut être soit un signal de capteur ou un ordre provenant d'une unité de commande.

Le système de surveillance que nous avons proposé dans nos travaux permet de définir un espace délimitant les paramètres qui caractérisent l'évolution du système. Cet espace basé à la fois sur des contraintes temporelles et physiques conditionne le bon fonctionnement du système. Parmi les perspectives que nous envisageons, il y a l'utilisation de cet espace dans l'établissement de la commande des systèmes dynamiques hybrides. En effet les informations disponibles sur le système, à savoir les différents modes de fonctionnement possibles ainsi que les contraintes établies dans le cahier des charges, permettent de déterminer l'espace de fonctionnement non défaillant, donnant ainsi à l'opérateur la possibilité de corriger en temps réel les trajectoires

utilisées par le systèmes si celles-ci l'amène à une violation de l'espace de fonctionnement établi.

Egalement, parmi les extensions possibles de nos travaux, On peut envisager de déterminer le modèle réseau de Petri adéquat qui permettra de modéliser le comportement du procédé décrit par l'automate hybride. Ce modèle permettra alors de modéliser des systèmes complexes de manière concise et lisible.

BIBLIOGRAPHIE

(Alla, H. et al, 1998) Alla, H. et R. David. Continuous and hybrid petri nets, Journal of Circuits, Systems and Computer, 1998, vol. 8, no 1, p. 159 – 188.

(Allahham, A et al, 2006) Allahham, A. and Alla, H. Monitoring of timed discrete events systems: Application to manufacturing systems. In The 32nd Annual conference of IEEE Industrial Electronics Society, 2006, pages 3609-3614.

(Allahham, A et al, 2007a) Allahham, A. and Alla, H. Design and implementation of a monitoring system using grafcet. In 4th International Conference on Informatics in Control, Automation and Robotics, 2007, Angers, France.

(Allahham, A et al, 2007b) Allahham, A. and Alla, H. Monitoring of a class of timed discrete events systems. In IEEE International Conference On Robotics and Automation, 2007, Rome, Italy, p. 3609-3614.

(Allahham, A, 2008) Allaham, A., Surveillance des systèmes à événements discrets commandés : Conception et implémentation en utilisant l'automate programmable industriel, thèse de doctorat, L'université Joseph Fourier, Grenoble, 2008.

(Alur, R et al, 1993) Alur, R., C. Courcoubetis, T. A. Henzinger et P.-H. Ho. «Hybrid automata : An algorithmic approach to the specification and verification of hybrid systems», Hybrid Systems, LNCS, 1993, p. 209 – 229.

(Alur, R et al, 1994) Alur, R. and Dill, D.. A theory of timed automata. Theoretical computer Science, 1994, vol. 126, p. 183-235.

(Alur, R et al, 1995) Alur, R., C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P. h. Ho, X. Nicollin, A. Olivero, J. Sifakis et S. Yovine. «The algorithmic

analysis of hybrid systems», Theoretical Computer Science, 1995, vol. 138, p. 3–34.

(Antsaklis, P et al, 1993) Antsaklis, P. J. Lemmon, M. D. and Stiver, J. A. Hybrid system modeling and event identification. Technical report, Technical Report of the ISIS Group at the University of Notre Dame ISIS-93-002, Notre Dame, January 1993.

(Asarin, E et al, 2006) Asarin, E., Dang, T., Frehse, G., Girard, A., Guernic, C. L., and Maler, O. Recent progress in continuous and hybrid reachability analysis. In Proceedings of the IEEE International Symposium on Computer-Aided Control Systems Design, Technische Universität München, 2006, Munich, Germany.

(Bemporad, A et al, 1999) Bemporad, A. and Morari, M. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 1999, vol. 35(3): p. 407 – 428.

(Bengtsson, J et al, 2004) Bengtsson, J. and Yi, W. Timed automata : Semantics, algorithms, tools. In Lectures on Concurrency and Petri Nets, volume LNCS 3098, Springer-Verlag, 2004.

(Branicky, M et al, 1994) Branicky, M. S. Borkar, V. S. and Mitter, S. K. A unified framework for hybrid control. In *IEEE Conference Decision and Control*, December 1994, p. 4228 – 4234.

(Branicky, M et al, 1995) Branicky, M. S. *Studies in hybrid systems: Modling, Analysis, and control*. Thèse de doctorat, Massachusetts Institute of Technologies, 1995.

(Cassandras, C et al, 1999) Cassandras, C. et S. Lafortune. Introduction to Discrete Event Systems, Kluwer Academic Publisher, 1999.

(Chen, Y.-L et al, 1997) Chen, Y.-L. and Provan, G. Modeling and diagnosis of timed discrete event systems- a factory automation example. In The American Control Conference, New Mexico, 1997, p. 31 – 36.

(Chow, E et al, 1984) Chow, E. Y. and Willsky, A. S. Analytical redundancy and the design of robust failure detection systems. *IEEE Transactions on Automatic Control*, July 1984, vol. 29(7), p. 602 – 615.

(Combacau, M et al, 1998) Combacau, M. Contribution à la surveillance hiérarchisée des systèmes complexes. Habilitation à diriger les recherches, 1998.

(Combacau, M et al, 2000) Combacau, M., Berrut, P., Charbonnaud, F., and Khatab, A. Réflexions sur la terminologie : Surveillance - supervision. In Groupement pour la recherche en Productique, Systèmes de Production Sûrs de Fonctionnement. 2000.

(Correcher, A et al, 2003) Correcher, A., Garcia, E., Morant, F., Quiles, E., and Blasco-Gimenez, R. Intermittent failure diagnosis in indstrial processes. In IEEE International Symposium on Industrial Electronics, 2003, vol. 2, p. 723 – 728.

(David, R et al, 1995) David, R. et H. Alla. Du Grafcet aux réseaux de Petri. Paris, Hermes Science Publications, 1995.

(David, R et al, 1995) David, R. et H. Alla., Discrete, Continuous, and Hybrid Petri Nets, Berlin, Heidelberg Springer, 2004.

(Derbel, H et al, 2006) Derbel, H., Yeddes, M., Hadj-Alouane, N. B., and Alla, H.. Diagnosis of a class of timed discrete event systems. In Proceedings of the 8th International Workshop on Discrete event systems, WODES'06, 2006, Michigan, USA. Published by IEEE (ISBN 1-4244-0053-8 and IEEE catalog number 06EX1259).

(Derbel, H et al, 2009) Derbel, H. Diagnostic à base de modèles des systèmes temporises et d'une sous-classe de systèmes dynamiques hybrides. Thèse de doctorat de l'université joseph fourrier, Grenoble, 2009.

(Derbel, H, 2009) Derbel, H. Diagnostic à base de modèles des systèmes temporisés et d'une sous-classe de systèmes dynamiques hybrides. Thèse de doctorat. Université Joseph fourrier, 2009.

(Dousson, C, 2007) Dousson, C. Contribution à l'application de la reconnaissance des formes et la théorie des possibilités au diagnostic adaptatif et prédictif des systèmes dynamiques. Thèse de doctorat, Université de Reims Champagne-Ardenne, 2007.

(Dubuisson, B, 1990) Dubuisson, B. Diagnostic et reconnaissance des formes, Hermès, 1990.

(Dubuisson, B, 2001) Dubuisson, B. *Diagnostic, intelligence artificielle et reconnaissance des formes*. Traité IC2 : Information - Commande - Communication. Hermes, 2001.

(Farreny, H, 1989) Farreny, H. Les systèmes experts - Principes et exemples. Cépadues, 1989.

(Frank, P. M, 1990) Frank, P. M. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – a survey and some new results. *Automatica*, 1990, vol. 26(3), p. 459 – 474.

(Frank, P. M, 1996) Frank, P. M. Analytical and qualitative model-based fault diagnosis - a survey and some new results. *European Journal of Control*, 1996, vol. 2(1), p. 6 – 28.

(Frehse, G, 2005) Frehse, G. Phaver: Algorithmic verification of hybrid systems past hytech. In Proceedings of the Fifth International Workshop on Hybrid Systems: Computation and Control, 2005, p. 258 – 273.

(Ghazel, M et al, 2005) Ghazel, M., Togueni, A., and Bigang, M. A monitoring approach for discrete events systems based on a timed perti net model. Proceedings of 16th IFAC World Congress, Prague 2005.

(Guéguen, H et al, 2004) Guéguen, H. et J. Zaytoon. «On the formal verification of hybrid systems», Control Engineering Practice, 2004, vol. 12, no 10, p. 1253 – 1267.

(Hamscher, W et al, 1992) Hamscher, W. Console, L. and Kleer. J. Readings in model-based diagnosis. *Morgan Kaufmann, San Meteo, CA, Etats-Unis*, 1992.

(Harel, D et al, 1987) Harel, D. et A. Pnueli. «Statecharts : A visual formalism for complex systems», 1987.

(Henzinger, T. A et al, 1996) Henzinger, T. A. «The theory of hybrid automata», Hybrid Systems II, LNCS, 1996, vol.999, p. 278 – 292.

(Henzinger, T. A et al, 1997) Henzinger, T. A., P. H. Ho et H. W. Toi. «Hytech : A model checker for hybrid systems», International Journal on Software Tools for Technology Transfer, 1997,vol. 1, no 1-2, p. 110 – 122.

(Huang, Z et al, 1996) Huang, Z., Chandra, V., Jiang, S., and Kumar, R. Modeling discrete event systems with faults using a rules based modeling formalism. Mathematical Modeling of Systems, 1996.

(Isermann, R, 1984) Isermann, R. Process fault detection based on modeling and estimation methods - a survey. *Automatica*, 1984, vol. 20(4), p. 387 – 404.

(Isermann, R et al, 1997) Isermann, R. and Ballé, P. Trends in the application of model-based fault detection and diagnosis of technical process. *Control Engineering Practice*, 1997, vol. 5(5), p. 709 – 719.

(Isidori, A, 1995) Isidori, A. *Nonlinear Control Systems*. Springer Verlag, 1995.

(Jones, H. L, 1973) Jones, H. L. *Failure Detection in Linear System*. Thèse de doctorat, MIT Cambridge, MA, 1973.

(Karoui, M. F et al, 2010a) Karoui, M. F., Alla H. and Chatti A. Monitoring of dynamic processes by rectangular hybrid automata, *Nonlinear Analysis: Hybrid Systems*, 2010. doi:10.1016/j.nahs.2010.05.004.

(Karoui, M. F et al, 2010b) Karoui, M. F., Alla H. and Chatti A. Surveillance des processus dynamiques par automates hybrides linéaires, Conferance Internationle Francophone d'Automatique (CIFA), 2010, papier n°187.

(Koutsoukos, X et al, 2001) Koutsoukos, X., F. Zhao, H. Haussecker, J. Reich et P. Cheung. «Fault Modeling for monitoring and diagnosis of sensor-rich hybrid systems», dans Proceedings of the 40th IEEE Conference on Decision and Control, 2001, p. 793 – 801.

(Kramer, M, 1987) Kramer, M. Malfunction diagnosis using quantitative models with non boolean reasoning in expert systems. *AIChE Journal*, 1987, vol. 33(1), p.130 – 140.

(Kurovszky, M, 2002) Kurovszky, M. *Etude des Systèmes Dynamiques Hybrides par représentation d'état discrete et automate hybride*. Thèse de doctorat, l'INPG, France, 2002.

(Manon, P, 2001) Manon, P. *Sur l'optimisation des séquences de fonctionnement des systèmes dynamiques hybrides*. Thèse de doctorat, LAGEP, Université Claude Bernard Lyon1, France, mars 2001.

(Mehra, R et al, 1971) Mehra, R. and Peschon, J. An innovation approach to fault detection and diagnosis in dynamic system. *Automatica*, 1971, vol. 7, p. 637 – 640.

(Mezyani, T, 2005) Mezyani, T. Méthodologie de surveillance des systèmes dynamiques hybrides, thèse de doctorat, Université des Sciences et Technologies de Lille, 2005.

(Mosterman, P. J, 1997) Mosterman, P. J. Hybrid Dynamic Systems : a Hybrid Bond Graph Modeling Paradigm and its Application in Diagnosis. Thèse de doctorat, Vanderbilt University, 1997.

(Montmain, J, 1992) Montmain, J. Interprétation qualitative de simulation pour le diagnostic en ligne de procédés continus. Thèse de doctorat, Institut National Polytechnique de Grenoble, 1992.

(Müller, O et al, 2000) Müller, O. et T. Stauner. «Modelling and verification using linear hybrid automata - a case study», Mathematical and Computer Modelling of Dynamical Systems, 2000, vol. 6, no 1, p. 71 – 89.

(Nourelfath, M, 1997) 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, L'Institut National des Sciences Appliquées de Lyon, INSA. 1997.

(Ondel, O, 2006) Ondel, O. Diagnostic par reconnaissance des formes : Application à un ensemble convertisseur-machine asynchrone. Thèse de doctorat, École Centrale de Lyon, 2006.

(Patton, R. J et al, 1991a) Patton, R. J. and Chen, J. A review of parity space approaches to fault diagnosis. In *SAFEPROCESS'91*, p. 239–255, Baden-Baden (Allemagne), 1991.

(Patton, R. J et al, 1991b) Patton, R. J. and Chen, J. A re-examination of the relationships between parity space and observer-based approaches in fault diagnosis. *Revue Européenne de Diagnostic et Sûreté de fonctionnement*, 1991, vol. 1(2), p. 183 – 200.

(Patton, R. J, 1994) Patton, R. Robust model based fault diagnosis: the state of the art. In *SAFEPROCESS'94*, volume 1, p. 1–23, Helsinki, Finland, 1994.

(Petri, C. A, 1962) Petri, C. A. Kommunication mit automaten. Thèse de doctorat, Univeristy of Bonn, 1962.

(Puri, A et al, 1996) Puri, A. Borkar, V. and Varaiya, P. e-approximation of differential inclusions. In *In Proceedings of Hybrid Systems III Workshop : Verification and Control*, vol. 1066 of *Lecture Notes in Computer Science*, 1996, pages 362–376.

(Rayhane, H, 2004) Rayhane, H. Surveillance des systèmes de production automatisés : Détection et Diagnostic. Thèse de doctorat, Institut National Polytechnique de Grenoble, INPG, 2004.

(Rouchon, G, 1992) Rouchon, G. Sûreté des automatismes: Comment assurer la sécurité, la disponibilité, la maintenanbilité des automatismes industriels ? : Les méthodes disponibles et les réglementations. *Publication CETIM. Mécanique et Productique*, 1992.

(Roux, O et al, 1996) Roux, O. et V. Rusu. «Uniformity for the decidability of hybrid automata», dans Proceedings of the 8th Conference on Computer Aided Veri CAV'96,1996, vol. 1145 of LNCS, p. 301 – 316.

(Sava, A, 2001) Sava, A. Sur la synthèse de la commande des systèmes à évènements discrets temporisés. Thèse de doctorat, Institut National Polytechnique de Grenoble, INPG. 2001.

(Sampath, M et al, 1995) Sampath, M. Sengupta, R. Lafortune, S. Sinnamohideen, K. and Teneketzis, D. Diagnosability of discrete event system. *IEEE Transactions on Automatic Control*, 1995, vol. 40(9), p. 1555 – 1575.

(Sampath, M et al, 1996) Sampath, M. Sengupta, R. Lafortune, S. Sinnamohideen, K. and Teneketzis, D. Failure diagnosis using discrete event models. *IEEE Transactions on Automatic Control*, 1996, vol. 4(2), p. 105 – 124.

(Shafer, G, 1976) Shafer, G. A mathematical theory of evidence. *Princeton University Press*, 1976, p. 22 – 28.

(Stiver, J et al, 1996) Stiver, J. Antsaklis, P. and Lemmon, M. A logical des approach to the design of hybrid control systems. *Math. and Computer Modeling, Special Issue on Discrete Event Systems*, 1996, vol. 23(11/12), p. 55 – 76.

(Tripakis, S, 2002) Tripakis, S. Fault diagnosis for timed automata. Proceeding 7th International Symposium on Formal Techniques in Real-Time and Fault Tolerant Systems (FTRTFT'02), 2791 of Lecture Notes in Computer Science, 2002, p. 205 – 224.

(Willsky, A. S, 1976) Willsky, A. S. A survey of several failure detection method. *Automatica*, November 1976, p. 601–611.

(Zad, S. H et al, 2005) Zad, S. H., Kwong, R. H., and Wonham, W. M. Fault diagnosis in discrete-events systems: Incorporating timing information. IEEE transaction on automatic control, 2005, vol. 50(7).

(Zaytoon, J, 2001) Zaytoon, J. *Modélisation, analyse et commande des systèmes dynamiques hybrides*. Hermès, 2001.

(Zwingelstein, G, 1995) Zwingelstein, G. Diagnostic des défaillances. Traité des nouvelles technologies, série Diagnostic et Maintenance, Hérmes, 1995.