

M/621.861

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université 8 Mai 1945 – Guelma
Faculté des Sciences et de la Technologie
Département de Génie Electrotechnique et Automatique



Domaine : Sciences et Technologie
Filière : Automatique et Informatique Industrielle
Spécialité : Commande et Diagnostic
Des Système Industriels



Mémoire de fin d'études
Pour l'obtention du diplôme de Master Académique

Approche hybride pour le diagnostic
d'un système dynamique hybride

Présenté par :
Gnango Frederic Alain Didier
IKONGA-KOUMOU Krissy-Evrard
Otoubou Oyaba Julasvic

Sous la direction de :
Dr. Boucerredj Leila.

Juin 2015

Remerciements

Dieu tout puissant

Mr Gnango Désiré

Me Kaba Sarata

Me Kouadio Ahou Léonie

Mr Gnango Roger

Mr Aboulaye Adama

Me Doussou Kaba

Mr Fofana Sekou

Otoubou Oyaba Julasvic

Tous mes amis à Guelma et en Algérie



Sommaire

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

Chapitre 1: Etude des systèmes dynamiques Hybrides

1.1 Introduction.....	3
1.2 Notions des systèmes dynamiques hybrides (SDH).....	3
1.3 Définition des systèmes Dynamiques Hybrides(SDH).....	3
1.4 Structure générique d'un SDH.....	4
1.5 Différentes catégories des phénomènes hybrides.....	6
1.6 Différents exemples représentatifs des SDH.....	8
1.7 Différentes types d'automates hybrides.....	13
1.8 Conclusion.....	16

Chapitre 2: Diagnostic d'un Système Dynamique Hybride

2.1 Introduction.....	17
2.2 Terminologie.....	17
2.3 Classification des défauts.....	19
2.4 Performance d'une procédure de diagnostic.....	20
2.5 Méthodes de diagnostic des systèmes dynamiques hybrides.....	21
2.5.1 Méthodes à base de modèle.....	21
2.5.2 Contribution fondées sur des approches continues.....	22
2.5.2.1 Approches à base de techniques de génération de résidus.....	22
2.5.2.2 Approche à base de raisonnement causal.....	23
2.5.3 Contribution fondées sur des approches discrètes.....	24
2.5.3.1 Approche de diagnostic de Lunze.....	24
2.5.3.2 Approche à base de modèle hybride à temps discret.....	25
2.5.4 Contribution fondées sur des approches mixtes.....	26

2.6 Conclusion.....	27
----------------------------	-----------

Chapitre 3 : Modélisation des Systèmes Dynamiques Hybrides

3.1 Introduction.....	28
3.2 Structure générique des SDH.....	28
3.2.1 Méthode de fonctionnement.....	28
3.2.1.1 Approche discrète.....	28
3.2.1.1.1 Modification de dynamique continue causées par des sorties discrètes.....	29
3.2.1.1.2 Modification de dynamique continue causées par des entrées discrètes.....	29
3.2.1.1.3 Modification de dynamique continue causées par la structure du système.....	30
3.2.1.2 Approche continu.....	31
3.2.2 Spécification de fonctionnement.....	31
3.2.2.1 Spécification discrète.....	32
3.2.2.2 Spécification continue.....	32
3.3 Conception ou Elaboration des SDH.....	33
3.3.1 Approche événementielle.....	34
3.3.2 Approche continue.....	34
3.3.3 Approche mixte.....	34
3.4 Les outils de modélisation.....	35
3.4.1 Automates Hybrides.....	35
3.4.2 Réseaux de Petri.....	36
3.4.2.1 Réseau de Petri Hybride.....	36
3.4.2.2 Réseau Hybride élémentaire.....	38
3.5 Conclusion.....	39

Chapitre 4 : Modélisation par Stateflow

4.1 Introduction.....	40
4.2 Les objets de stateflow.....	41

4.2.1 Les états.....	41
4.2.2 Les transitions.....	42
4.2.2.1 Les transitions par défaut.....	43
4.2.2.2 Les labels des transitions.....	43
4.2.3 Les évènements.....	44
4.2.4 Les objets data (Données).....	44
4.3 Différents exemples d'applications.....	45
4.4 Conclusion.....	53

Chapitre 5 : Cas d'étude - Anti-lockBraking System (ABS)

5.1 Introduction.....	54
5.2 Principe de fonctionnement.....	54
5.3 Approche hybride et algorithme de simulation.....	55
5.4 Application : Intérêt de l'ABS.....	56
5.4.1 Modélisation dynamique.....	57
5.4.2 Modélisation physique.....	59
5.4.2.1 Simulation en mode ABS.....	60
5.4.2.2 Simulation en mode sans ABS	61
5.4.2.3 Simulation de l'ABS avec régulateur.....	62
5.5 Comparaison entre le freinage Sans ABS et Avec ABS.....	65
5.6 Conclusion.....	65

Conclusion Générale.....	66
---------------------------------	-----------

Liste Des Figures

Chapitre 1 : Etudes Des Systèmes Dynamiques Hybrides

Figure 1.1 : Structure du système dynamique hybride.....	4
Figure 1.2 : Commutation autonome.....	5
Figure 1.3 : Commutation Contrôlé.....	6
Figure 1.4 : Saut Autonome.....	7
Figure 1.5 : Trajectoire d'une boule de billard.....	8
Figure 1.6 : Système embrayage mécanique.....	9
Figure 1.7 : Modèle de thermostat.....	10
Figure 1.8 : Trajectoire de la température.....	11
Figure 1.9 : Système hydraulique.....	13
Figure 1.10 : Automate à états finis.....	14
Figure 1.11 : Automate hybride modélisant une machine simple.....	16

Chapitre 2 : Diagnostic d'un Système Dynamique Hybride

Figure 2.1 : Principe de diagnostic de SDH selon l'approche de Lunze.....	9
---	---

Chapitre 3 : Modélisation des Systèmes Dynamiques Hybrides

Figure 3.1 : Exemple d'une structure physique.....	3
Figure 3.2 : Partie d'un automate hybride.....	6
Figure 3.3 : Modèle de thermostat avec spécification de fonctionnement.....	7
Figure 3.4 : Automate hybride.....	10
Figure 3.5 : Exemple de Réseau Hybride élémentaire.....	13

Chapitre 4 : Modélisation par Stateflow

Figure 4.1 : Agencement des différentes plates-formes du logiciel Matlab	
--	--

Chapitre 5 : Cas d'étude - Anti-lock Braking System (ABS)

Figure 5.1 : Système de freinage antiblocage (ABS)	59
Figure 5.2 : Le mécanisme d'ABS sur une moto BMW K75.....	61
Figure 5.3 : Freinage avec et sans ABS	62
Figure 5.4 : Efforts agissant sur une roue freinée.....	63

Figure 5.5 : Modèle de freinage antiblocage.....	63
Figure 5.6 : Modèle de freinage antiblocage.....	64
Figure 5.7 : Résultat de la simulation en mode ABS.....	66
Figure 5.8 : Résultat de la simulation sans mode ABS	
Figure 5.9 : Modèle de freinage antiblocage avec PID	
Figure 5.10 : Schéma fonctionnel du régulateur PID	
Figure 5.11 : Résultat de la simulation d'un PID avec des valeurs expérimentales	
Figure 5.12 : programme du PID dans le bloc Chart	
Figure 5.13 : Résultat de la simulation en mode ABS avec PID	
Figure 5.14 : distance d'arrêt en freinant brutalement avec et sans ABS	

Summary

A hybrid system is a dynamic system which presents an at the same time discrete and continuous behavior. The hybrid systems find their application to the level of the industrial systems like the robots and the automobile world, in particular the brake (ABS).

When the driver slows down on a slipping road, the anti-locking brake prevents the wheels from blocking, thus preserving the maneuverability of the vehicle. When the wheels are blocked, they cannot transmit the forces of steering any more, which involves the loss of control of the vehicle. To avoid this possibility, the control unit ABS uses speed sensors of wheel to supervise the number of revolutions of each wheel. If it detects that a wheel is on the point of blocking, a valve solenoid, located in the element of central control of the anti-locking brake, reduced pressure of braking applied to the wheel in question until it starts again to turn freely. The pressure is then brought back to the threshold of blocking.

The objective in this memory is to develop a hybrid approach for the control and the monitoring of the studied dynamic system which is the brake of the automobile world. The example of application on the anti-locking one of the wheels (ABS) illustrates the study of the real behavior of the system on the Matlab/Simulink/Stateflow platform.

Key-words: hybrid dynamic systems, hybrid approach, diagnosis, Matlab, Simulink and Stateflow, regulation.

Liste Des Figures

Chapitre 1 : Etudes Des Systèmes Dynamiques Hybrides

Figure 1.1 : Structure du système dynamique hybride.....	4
Figure 1.2 : Commutation autonome.....	5
Figure 1.3 : Commutation Contrôlé.....	6
Figure 1.4 : Saut Autonome.....	7
Figure 1.5 : Trajectoire d'une boule de billard.....	8
Figure 1.6 : Système embrayage mécanique.....	9
Figure 1.7 : Modèle de thermostat.....	10
Figure 1.8 : Trajectoire de la température.....	11
Figure 1.9 : Système hydraulique.....	13
Figure 1.10 : Automate à états finis.....	14
Figure 1.11 : Automate hybride modélisant une machine simple.....	16

Chapitre 2 : Diagnostic d'un Système Dynamique Hybride

Figure 2.1 : Principe de diagnostic de SDH selon l'approche de Lunze.....	9
---	---

Chapitre 3 : Modélisation des Systèmes Dynamiques Hybrides

Figure 3.1 : Exemple d'une structure physique.....	3
Figure 3.2 : Partie d'un automate hybride.....	6
Figure 3.3 : Modèle de thermostat avec spécification de fonctionnement.....	7
Figure 3.4 : Automate hybride.....	10
Figure 3.5 : Exemple de Réseau Hybride élémentaire.....	13

Chapitre 4 : Modélisation par Stateflow

Figure 4.1 : Agencement des différentes plates-formes du logiciel Matlab	
--	--

Chapitre 5 : Cas d'étude - Anti-lock Braking System (ABS)

Figure 5.1 : <i>Système de freinage antiblocage (ABS)</i>	59
Figure 5.2 : Le mécanisme d'ABS sur une moto BMW K75.....	61
Figure 5.3 : Freinage avec et sans ABS	62
Figure 5.4 : Efforts agissant sur une roue freinée.....	63

Résumé

Un système hybride est un système dynamique qui présente un comportement à la fois discret et continu. Les systèmes hybrides trouvent leur application au niveau des systèmes industriels comme les robots et le monde automobile, notamment le système de freinage (ABS).

Lorsque le conducteur freine à fond sur une route glissante, le système de freinage antiblocage empêche les roues de bloquer, préservant ainsi la manœuvrabilité du véhicule. Quand les roues se bloquent, elles ne peuvent plus transmettre les forces de braquage, ce qui entraîne la perte de contrôle du véhicule. Pour éviter cette éventualité, l'unité de contrôle ABS utilise des détecteurs de vitesse de roue pour surveiller la vitesse de rotation de chaque roue. Si elle détecte qu'une roue s'apprête à bloquer, une valve solénoïde, située dans l'élément de contrôle central du système de freinage antiblocage, réduit la pression de freinage appliquée à la roue en question jusqu'à ce qu'elle recommence à tourner librement. La pression est ensuite ramenée au seuil de blocage.

L'objectif dans ce mémoire est de développer une approche hybride pour le contrôle et la surveillance du système dynamique étudié qui est le système de freinage du monde automobile. L'exemple d'application sur l'antiblocage des roues (ABS) illustre l'étude du comportement réel du système sur la plateforme Matlab/Simulink/Stateflow.

Mots clés : systèmes dynamiques hybrides, approche hybride, diagnostic, Matlab, Simulink et Stateflow, régulation.

Introduction générale

Introduction générale

L'automatisation des installations industrielles vise à augmenter la productivité des systèmes et à réduire les coûts de la maintenance des équipements de production. Cependant, la complexité due à l'automatisation des systèmes de production implique des besoins croissants en termes de disponibilité et de performance. Ainsi, il est nécessaire de disposer d'une fonction permettant le diagnostic des défaillances pouvant affecter le fonctionnement du système. Ceci permet d'envisager des actions correctives pour que ce dernier retourne à son fonctionnement nominal. Un module de diagnostic est nécessaire, non seulement pour améliorer les performances et la productivité des systèmes, mais également pour limiter les conséquences des pannes qui peuvent être catastrophiques sur le plan des biens et des vies humaines.

La fonction de diagnostic consiste à détecter une défaillance, de localiser son origine et de déterminer ses causes. 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 anormal. Plusieurs chercheurs ont abordé la thématique de la surveillance industrielle et du diagnostic mettant ainsi en évidence l'intérêt manifesté aussi bien par la communauté du contrôle des systèmes avec des approches issues de l'automatique. Les approches de diagnostic développées durant ces dernières décennies peuvent être classées en deux grandes catégories : les approches avec modèles qui se basent sur l'existence d'un modèle du système à surveiller et les approches sans modèles qui se basent sur l'analyse des variables de surveillance et sur l'expertise humaine.

Le principe des méthodes avec modèles repose sur la comparaison du comportement prévu par le modèle avec le comportement réellement observé du système. Tout écart entre ces deux comportements sera synonyme de défaillance. L'utilisation d'un modèle du système pour son diagnostic nécessite souvent sa conformité au critère de diagnosticabilité [32]; [39]. Il s'agit de vérifier si chaque défaillance peut être détectée et isolée dans un temps fini après l'occurrence du défaut source de la défaillance. En d'autres termes, ce critère consiste à déterminer si le modèle du système est suffisamment riche en information pour permettre l'identification de tous les défauts affectant son fonctionnement. De nombreuses approches de diagnostic à base de modèles ont été proposées dans la littérature. On retrouve ainsi les approches issues de la communauté FDI (Fault Detection and Isolation) [25]; [23]; [10], les approches issues de la communauté des Systèmes à Événements Discrets (SED) [31]; [39]; [42]; [44].

La notion de diagnostic des SED a été introduite au milieu des années 1990 dans les travaux de Sampath [39]; [40], qui ont inspiré de nombreuses extensions. Cette approche de référence repose sur la compilation d'un automate appelé

Introduction Générale

diagnostiqueur, à partir d'une machine à états finis modélisant le comportement du système. La dynamique du SED dans cette approche est décrite par une séquence d'événements qui caractérisent les transitions d'états du système. Ainsi, la prise en compte du temps est réalisée uniquement par l'ordre d'occurrence des événements. Cela entraîne une perte considérable d'information, indispensable pour l'identification des défauts. En effet, les dates d'occurrence des événements peuvent jouer un rôle considérable dans la discrimination des défauts.

Dans certains cas, les comportements défailants se manifestent uniquement par un changement des dates d'occurrence des événements observables, ce qui rend inappropriée, l'utilisation d'une démarche de diagnostic fondée sur une abstraction purement discrète de l'évolution du système. Dans certains systèmes, il faut aller au-delà de la prise en compte du temps, il est alors nécessaire d'introduire une modélisation mixte comprenant des variables continues et des variables discrètes. De tels systèmes, intégrant à la fois une dynamique continue et une dynamique discrète, sont appelés Systèmes Dynamiques Hybrides (SDH).

Les travaux de ce mémoire porteront sur la proposition d'une approche hybride de diagnostic pour les systèmes dynamiques. Il est composé de cinq principaux chapitres.

Le premier chapitre permet d'exposer les différents concepts liés à notre travail. D'abord, il permet de définir les systèmes concernés par notre approche de diagnostic à savoir, les systèmes dynamiques hybrides. Leur structure générale, leur interface de commande et les modes de commutations entre leurs parties discrètes et continues.

Dans le deuxième chapitre, nous présentons un rappel de la terminologie du diagnostic rencontrée dans la littérature et employée dans ce mémoire. Enfin, nous citons les principales méthodes de diagnostic existantes dans la littérature.

Le troisième chapitre concerne la modélisation des systèmes dynamiques hybrides, nous y dressons une présentation détaillée des méthodes de modélisation et des spécifications qui leur sont liées en vue de leur élaboration complète et comprenant toutes leurs performances.

Dans le quatrième chapitre nous modéliserons et simulerons des systèmes réels via la plate-forme de travail matlab et notamment grâce à la plate-forme simulink/stateflow qui nous permettra de concevoir des systèmes réels.

Et par la suite nous évoquerons notre cas d'étude au cours de ce dernier chapitre. Nous y tenterons de montrer le bien-fondé des approches hybrides concernant les systèmes complexes ainsi que les perspectives estimées pour ce travail.

Chapitre 1 :

Etude des systèmes dynamiques
hybrides

1.1 Introduction

Les méthodes de l'automatique n'ont longtemps reposé que sur deux types de technologies, fondées sur des méthodes qui leurs étaient propres et appliqués à des domaines différents : les systèmes continus et des systèmes événementiels. Les systèmes continus font appel pour leur description à des outils mathématiques tels que les équations différentielles, tandis que les systèmes à événements discrets sont classiquement représentés par des formalismes états- transitions ayant une représentation graphique (*Grafcet, réseau de Petri, automates à états finis etc....*).

Néanmoins, les systèmes réels sont souvent des systèmes complexes dont la dynamique est modélisée, d'un point de vue macroscopique, par des phénomènes discrets et continus, démontrant ainsi que la frontière entre systèmes continus et systèmes à événements discret n'était pas si étanche. Il apparaît alors évident que ces systèmes ne peuvent se contenter d'une représentation homogène à dynamiques purement continue ou purement événementiels. Depuis les années quatre-vingt-dix, une attention particulières de la part de la communauté scientifique s'est porté sur l'étude de ces systèmes dits hybrides [1] et de nombreuses approches de modélisation traitant à la fois les aspects continus et discrets ont été proposés.

1.2 Notions des systèmes dynamiques hybrides (SDH)

Les systèmes dynamiques hybrides sont des systèmes où la dynamique **discrète** interagit avec la dynamique **continue**. C'est cette interaction qui détermine le comportement qualitatif et quantitatif du SDH.

1.3 Définition des systèmes Dynamiques Hybrides (SDH)

Les systèmes dynamiques hybrides ou encore appelé SDH peuvent être définis de différentes manières selon le vocabulaire attribué. En outre avant que nous puissions proposer les différentes définitions des SDH, nous souhaitons dans les lignes qui suivent citer et par la suite définir deux notions de bases des SDH pour ainsi avoir une meilleure compréhension des définitions des SDH.

Les deux notions de bases du SDH sont :

- **La notion Continue** : dans le cadre des SDH une variable est dite continue si elle peut prendre ses valeurs dans un ensemble continu de valeurs (*donc non dénombrable*) et ses variations ne présentent pas de discontinuités.

En effet, la variables prend alors ses valeurs dans l'ensemble des réels (*température, concentration*).

- **La notion discrète** : dans le cadre des SDH une variable dite discrète si elle peut prendre ses valeurs dans un ensemble dénombrable de variables d'état discrètes, sous-ensemble des entiers naturels (*nombre de pièces*).

Ainsi, nous pouvons par la suite énoncer les différentes définitions des SDH.

➤ **Définition 1:**

Les SDH sont des systèmes dont le comportement dynamique est défini par des entités ou des processus de nature différente.

➤ **Définition 2 :**

Un système est dit hybride s'il possède deux parties de variables d'états : les variables d'états **continues** et les variables d'états **discrètes**.

➤ **Définition3 :**

Les SDH caractérisent les systèmes faisant intervenir explicitement et simultanément des phénomènes de types continu et évènementiel. Pour leur description, ils nécessitent l'utilisation de fonction du temps **continus** par morceaux et de fonction à valeurs **discrètes**.

➤ **Définition SDH 4 :**

Un système discret est tel que toutes ses variables d'état sont discrètes.

Un système **continu** est tel que toutes ces variables d'état sont continues.

Un système **hybride** comprend au moins : une variable d'état discrète et une variable d'état continue.

1.4 Structure générique d'un système dynamique hybride

D'une façon général, un système dynamique hybride est composé d'un système dynamique continu, d'un système à évènement discret et d'une interface qui gère les interactions entre les deux évolutions (*continue et discrète*) [34].

- **La partie discrète :**

Elle est associé à un système à évènement discret (*SED*) est représenter par un ensemble fini d'état discret à un autre état discret successeur est

réaliser grâce à l'occurrence d'évènements. Ces évènements sont de deux types : **évènements contrôlés** et **évènements autonomes**.

Les **évènements contrôlés** sont des évènements externes qui correspondent aux commandes discrètes automatiques ou manuelles, d'une part. Et d'autre part, les **évènements autonomes** ou spontanés sont des évènements internes produits lorsque le vecteur d'état continu traverse certaines frontières de l'espace d'état.

Ces évènements expriment donc bien l'influence de l'état continu sur l'état discret.

Les modèles les plus classiques permettant d'appréhender les SED sont les automates à états finis, les réseaux de Petri et les Statecharts.

- **La partie continue :**

L'évolution de la partie continue caractérise un espace de temps continu et peut être représenté de différentes manières (*équations différentielles ordinaires, algèbre-différentielles, les fonctions de transfert, les bonds graphs, etc....*).

Mais, l'évolution continue est classiquement modélisée par des équations d'état, en général de types non linéaires. Le modèle de comportement continu est exprimé sous la forme d'état :

$$y(x) = \begin{cases} \dot{x}(t) = f(x(t), z(t)) \\ h(x(t), z(t)) \end{cases} \quad (1.1)$$

Où $x(t)$ est le vecteur d'état, $y(t)$ est le vecteur des sorties et $z(t)$ est un vecteur regroupant les entrées connues ou inconnues, les perturbations, etc.....

F et h sont les fonctions vectorielles linéaires ou non linéaires.

De plus ces équations différentielles contraignent l'évolution continue de l'état continu. On parle alors de contraintes de type égalité.

- **l'interface :**

L'interface traduit l'interaction entre la partie continue et la partie discrète du SDH. L'état discret détermine la dynamique continue spécifique de la partie continue du SDH. La représentation formelle de l'interface est plus complexe et dépend des approches modélisation considérées.

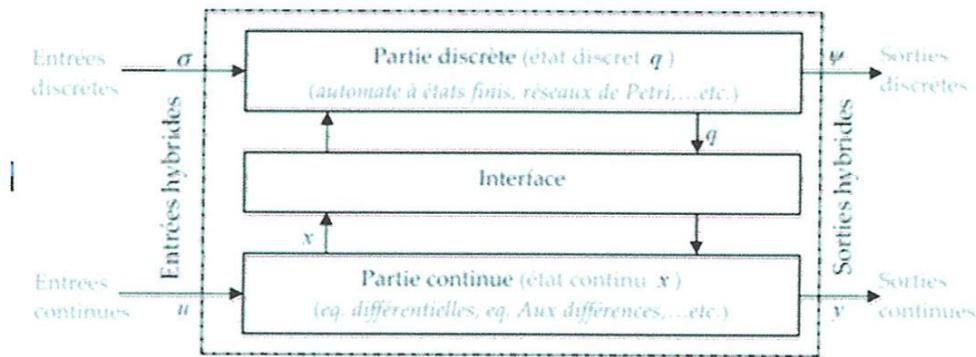


Figure 1.1 : Structure du système dynamique hybride

1.5 Différentes catégories des phénomènes hybrides

Dans cette partie, nous proposons une classification par catégories des différents phénomènes hybrides intervenant dans les SDH, Sachant qu'un SDH peut posséder plusieurs types de phénomènes. L'état hybride d'un système peut être nécessaire aux phénomènes physiques qui le dirigent.

En effet, un certains nombres de phénomènes physiques considérés comme hybrides ont été regroupés en deux catégories principales qui sont des **changements instantanés dans l'état** dits **saut** et des **changements instantanés dans la dynamique** (le champ de vecteur) dits **commutations**.

En effet, ces phénomènes traduisent leur influence sur les modèles mathématiques utilisés pour décrire les différentes classes des systèmes.

En outre, rappelons que l'évolution d'un système continu est définie par un ensemble d'équation différentielle de la forme :

$$\dot{x} = f(t, x) \quad (1.2)$$

Où $f(x, t)$ est appelé champ de vecteur d'état, \dot{x} l'état du système et t le temps.

Nous allons dans les lignes qui suivent énoncer les deux grandes catégories principales qui composent les phénomènes hybrides.

➤ 1^{ère} Catégorie de SDH : les commutations

- **Commutation autonome :**

Une commutation autonome est un phénomène qui est observé quand la fonction d'évolution change d'une façon discontinue lorsque l'une des variables d'état continu atteint certains seuils, comme schématisé sur la figure (a) pour une seule variable d'état. C'est le cas des systèmes à hystérésis figure (b).

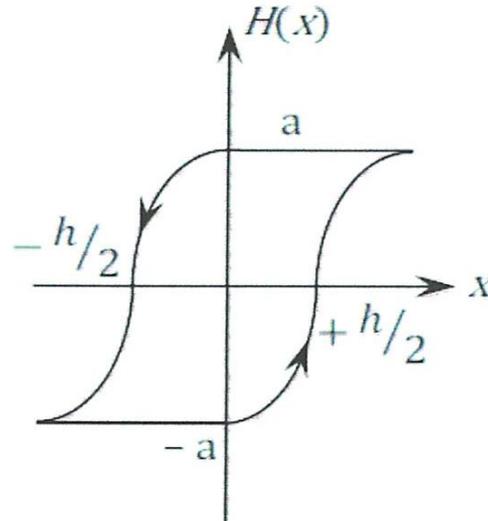
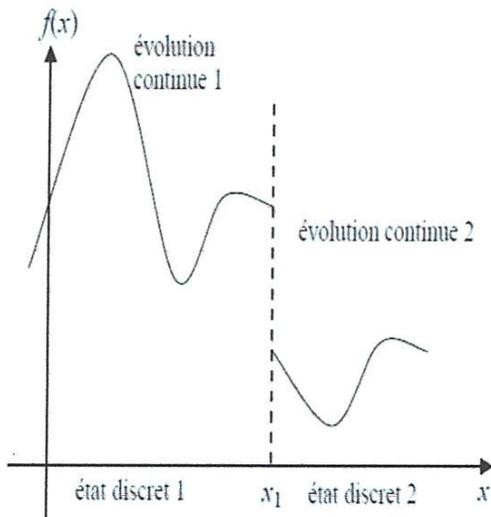


Figure a : Commutation à la valeur seuil x_1

Figure b : La fonction d'hystérésis

Figure 1.2 : Commutation autonome

- **Commutation contrôlée :**

Dans ce cas, le champ de vecteur commute en réponse à une loi de commande c'est-à-dire que le système traduit un phénomène ou le champ de vecteur $f(x, t)$ change de façon discontinue et instantanée en réponse à une entrée de commande (voir figure 1.3).

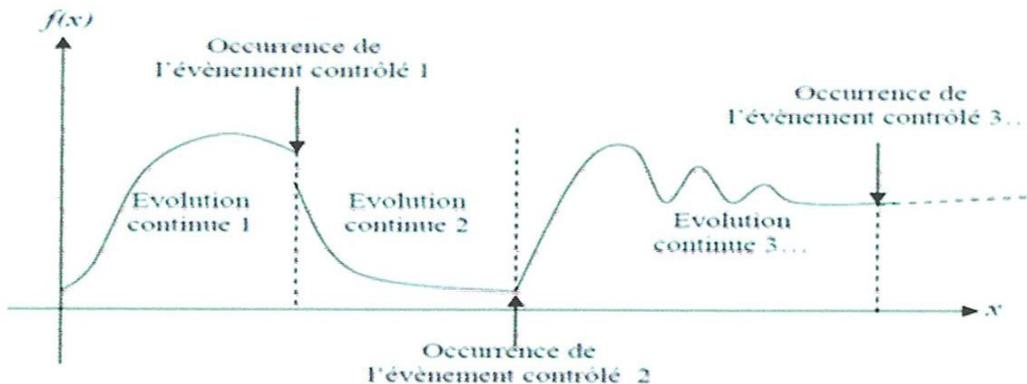


Figure 1.3 : Commutation Contrôlé

➤ 2^{ème} Catégorie de SDH : Les sauts

- **sauts autonomes :**

Le saut autonome est un phénomène rencontré dans les systèmes mécaniques, il s'agit d'un saut discontinu de l'état continu x lorsqu'il atteint une région donnée de l'espace d'état c'est-à-dire que l'état passe de façon discontinue de sa valeur courante à une autre.

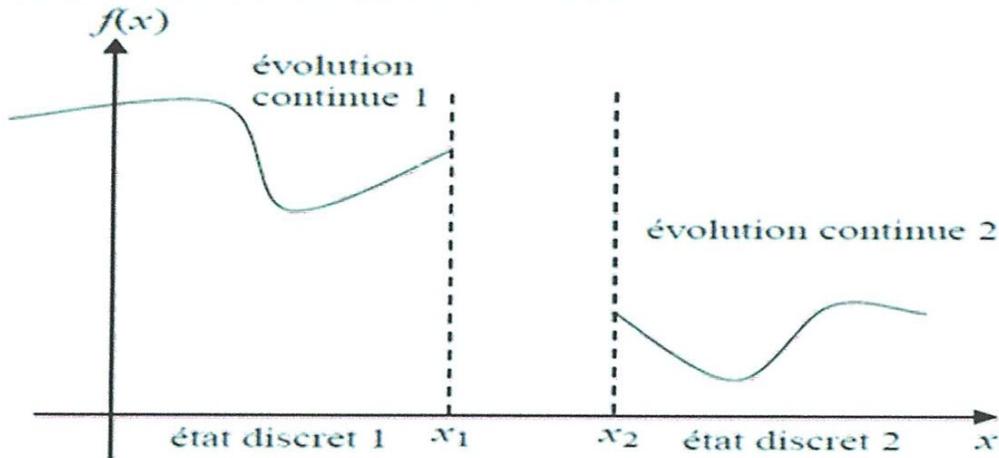


Figure 1.4 : Saut Autonome

- **Sauts commandés :**

L'état ou une partie change instantanément sous l'influence d'une action extérieure. Considérons par exemple un modèle simple d'un stock ou on dépose les quantités $(\alpha_1, \alpha_2, \dots)$ de matière aux instants $(t_1 < t_2 < \dots)$. L'évolution de la quantité de matière y est régie par l'équation suivante :

$$\dot{y} = -Q(t) + \sum_i \delta(t - t_i) \alpha_i \quad (1.3)$$

Avec Q la fonction d'utilisation de la matière et δ l'impulsion de Dirac.

1.6 Différents exemples représentatif des SDH

✓ 1^{ère} Exemple : Le jeu de billard

Un autre exemple classique c'est la table de billard de longueur l et de largeur h , avec une boule, comme l'illustre la figure 1.5.

La position initiale de boule est (x_0, y_0) et après avoir été frappée elle commence à se déplacer avec une vitesse v . Quand la boule arrive à un côté de la table

parallèle à l'axe y , elle rebondit et le signe de la composante de la vitesse v_x change. De même, le signe de la composante de la vitesse v_y change lorsque la boule arrive à un côté parallèle à l'axe x . La combinaison des signes des composantes de la vitesse donne quatre directions différentes du mouvement de la boule.

Le système de mouvement de la boule est un Système Dynamique Hybride (SDH) dont la dynamique continue est représentée par le mouvement de la boule, et la dynamique discrète par changement de sens de la boule lorsque celle-ci est heurtée aux côtés de la table.

Le changement d'état de la boule (*changement de sens du mouvement*) présente une commutation autonome.

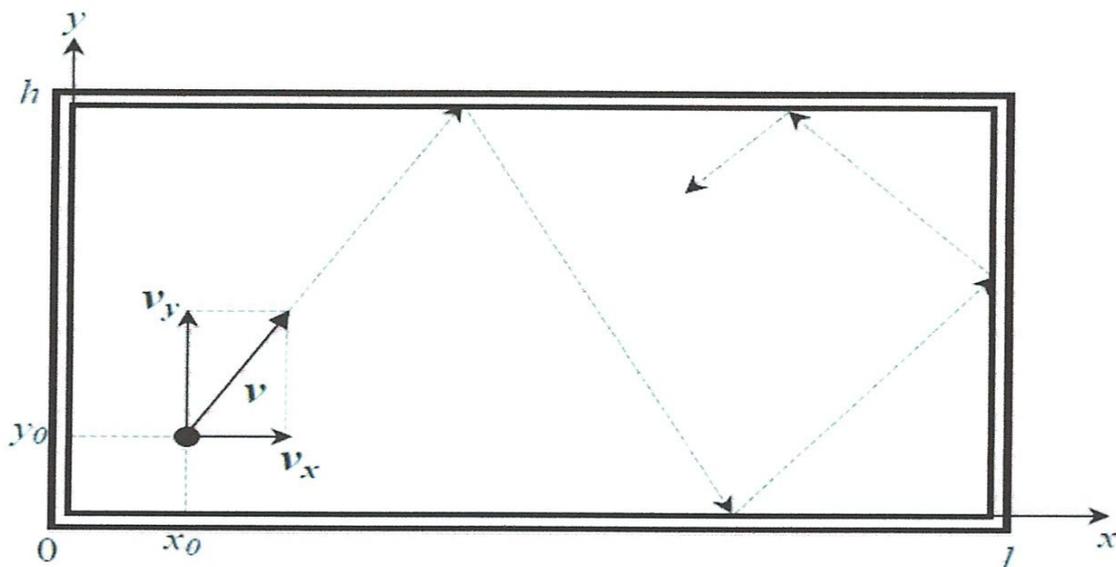


Figure 1.5 : Trajectoire d'une boule de billard

✓ 2^{ème} exemple : l'embrayage

Il s'agit d'un système composé de deux masses en rotation. Les masses sont couplées par un embrayage mécanique idéal. Chaque masse i , dont l'inertie est J_i , est entraînée par un couple à une vitesse de rotation. Quand les masses sont couplées, les valeurs des vitesses de rotation sont identiques. Ces vitesses sont indépendantes quand les masses sont découplées, cela est illustré par la figure 1.6.

Lorsque les axes de rotation sont indépendants le système est *débrayé*. Le modèle mathématique est donné par :

$$\begin{pmatrix} j_1 & 0 \\ 0 & j_2 \end{pmatrix} \begin{pmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \Omega_1 \\ \Omega_2 \end{pmatrix} \quad (1.4)$$

Quand les axes de rotation sont couplés le système est *embrayé*, les vitesses de rotation sont identiques et le système peut être décrit par l'équation :

$$\begin{pmatrix} j_1 & j_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix} + \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Omega_1 \\ \Omega_2 \end{pmatrix} \quad (1.5)$$

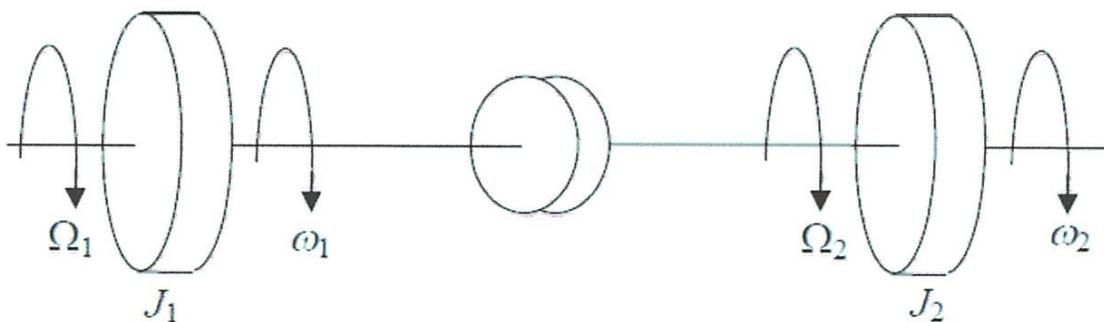


Figure 1.6 : Système embrayage mécanique

✓ 3^{ème} exemple : le thermostat

Considérons l'exemple classique d'un thermostat utilisé pour maintenir la température dans une chambre. Le système étudié est composé par un système de chauffage et capteur de température. Les seuils inférieur et supérieur du thermostat sont fixé à des valeurs θ_m et θ_M respectivement tel que :

$$\theta_m < \theta_M$$

Le système de chauffage est en **marche** tant que la température dans la chambre est inférieure au seuil θ_M .

Le chauffage est en arrêté lorsque le capteur détecte le seuil supérieur θ_M et il reste en **arrêt** jusqu'au moment où la température chute au dessous du seuil inférieur θ_m .

La température de la chambre et le thermostat peuvent être vus comme un système dynamique hybride (SDH) dont l'évolution continue est définie par la variation de la température dans la chambre et l'évolution discrète par le passage de l'état marche du système de chauffage à l'état arrêté.

Considérons que l'évolution de la température dans la chambre peut être modélisée par les équations différentielles suivantes :

$$\dot{x} = \begin{cases} f_1(x) = -x + \alpha & \text{si le chauffage est en marche} \\ f_2(x) = -x & \text{sinon} \end{cases} \quad (1.6)$$

Où $\alpha \in R_+$ est une constante réelle positive.

D'une manière graphique le système considéré peut être représenté par un graphe orienté présenté dans la figure suivante, les sommets du graphe correspondent aux dynamiques continues des états discrets du système. Notamment, la dynamique f_1 est associée au sommet modélisant l'état en marche du système de chauffage, et f_2 au sommet modélisant l'état d'arrêt. Le passage d'un état vers l'autre est modélisé par des arcs étiquetés.

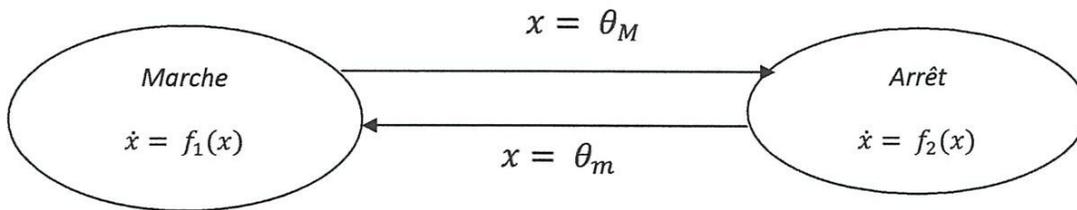


Figure 1.7 : Modèle de thermostat

Le problème de l'analyse consiste à vérifier que la température dans la chambre reste toujours dans l'intervalle désiré, notamment : $\theta_m \leq x \leq \theta_M$.

Pour cet exemple simple, les solutions analytiques des équations différentielles peuvent être facilement trouvées.

Ainsi, pour une valeur initiale de la température $x_0 = x(0) = \theta_0$, les solutions analytiques trouvées sont : $x(t) = \theta_0 e^{-t} + \alpha(1 - e^{-t})$ pour la dynamique correspondant à l'état de marche du système de chauffage, et $x(t) = \theta_0 e^{-t}$ pour l'autre état.

Initialement, supposons que le système est dans l'état en marche et la valeur initiale de la température vérifie la relation $\theta_0 \in [\theta_m, \theta_M]$. Dans cet état, l'évolution de la température respectera l'expression :

$$x(t) = \theta_0 e^{-t} + \alpha(1 - e^{-t}) \quad (1.7)$$

L'évolution croissante fait que, au bout de t_1 unité de temps, le seuil θ_M est atteint. Alors, le système de chauffage passera dans l'état arrêt. Suite au changement d'état du système, la dynamique de la température change et la nouvelle évolution est donnée par :

$$x(t) = \theta_0 e^{-(t_1+t)} \quad (1.8)$$

Dans cet état, la température aura une évolution décroissante jusqu'au moment où le seuil inférieur θ_m est atteint. A cet instant, le chauffage sera remis en marche et le système reviendra dans l'initial.

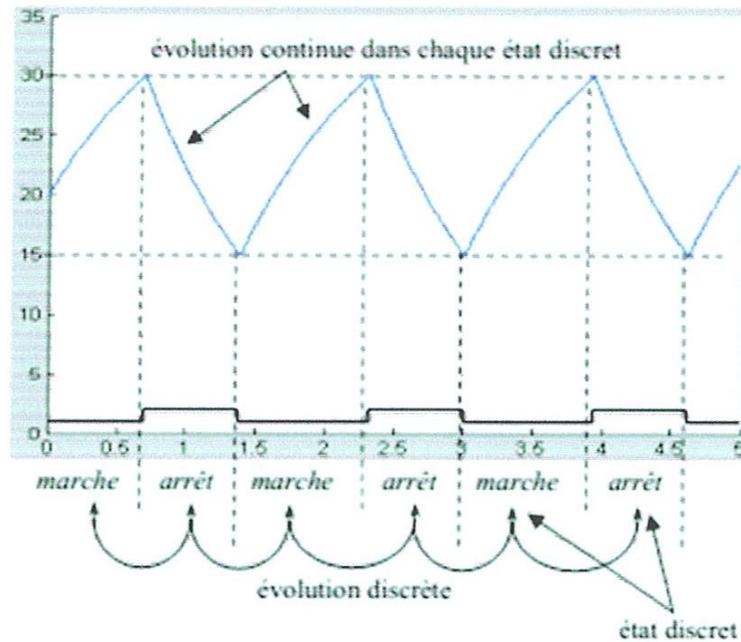


Figure 1.8 : Trajectoire de la température

D'après l'allure de la température (*figure précédente*), on constate clairement que ce système représente un système dynamique hybride, il comporte deux types d'évolution, une évolution continue et une évolution discrète (*changement d'état discret*) interagissent entre eux.

4^{ème} exemple : Système hydraulique

Le système de la figure (c) est constitué d'un réservoir de section S muni d'une conduite C , une vanne pneumatique voir figure (d) situé sur C et commandé en Tout ou Rien (*TOR*), permet de prélever du liquide pour l'utilisation. Une pompe P permet d'alimenter le réservoir et une commande automatique (*régulateur PI*) permet de maintenir un niveau de liquide constant dans le réservoir [22].

Le niveau de liquide $h(t)$ est une variable dont l'évolution est continue. Sa valeur dépend de s débits entrants (*sortie de la pompe*) et sortants (*évacuation au travers de la vanne V*). Ces deux débits ont eux aussi des évolutions continues : le débit d'entrée

est régulé (régulateur PI) et la valeur du débit de sortie est fonction de l'ouverture (restriction) de la vanne pneumatique.

L'ouverture de la vanne est en toute rigueur une fonction dans le temps. S on évolution est donné par la fonction θ_e , $\theta_e(x)$, où x est la position de la tige.

Lors d'une commande d'ouverture (*passage de $x = 0\%$ à $x = 100\%$*), l'évaluation continue de x , donc θ_e , est très rapide et la dynamique peut donc être négligée. Ainsi un état discret associé à la vanne peut être considéré. Cet état prend deux valeurs ou modalités correspondant respectivement à $x = 0\%$ (*vanne fermée*) et $x = 100\%$ (*vanne ouverte*). Ceci suffit en première approximation à caractériser le comportement de la vanne dans le système global. L'introduction de cet état discret met en évidence deux modes de fonctionnement du système et permet de simplifier la modélisation complète du système.

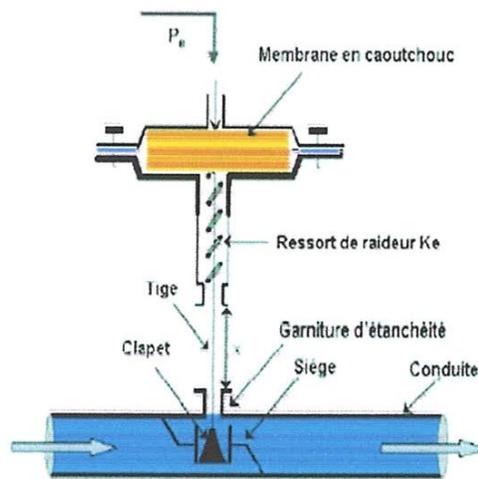
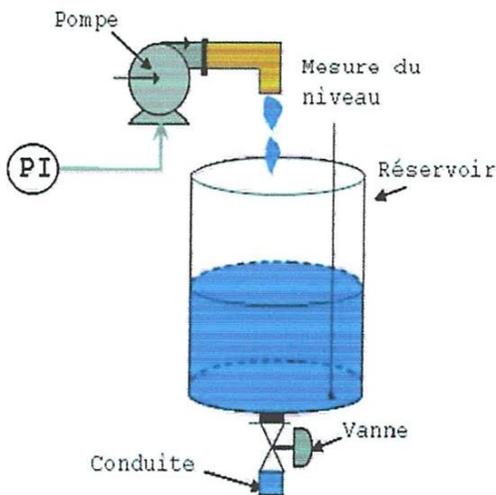


Figure c : Représentation du système hydraulique

Figure d : Représentation de la vanne

Figure 1.9 : Système hydraulique

1.7 Différents types d'automates hybrides

Un automate hybride est un automate à états finis qui a été étendu par des variables continues. Dans chaque sommet discret, la dynamique des variables continues est définie par des équations différentielles. Et les transitions entre les états discrets dépendent des valeurs des variables continues. Nous commençons dans la suite par définir les automates à états finis. C'est à partir de ce formalisme que les automates hybrides ont été définis.

➤ **Automates à états fini :**

Les automates sont des machines abstraites, très utilisées en informatique, qui permettent de modéliser le comportement d'un grand nombre de systèmes. De façon très informelle, un automate est un ensemble d'états du système, relié entre eux par transitions qui sont marquées par des symboles.

Définition formelle :

Un automate à états finis est un quadruplé $AEF = (Q, \Sigma, T, q_0)$

Où :

- Q est un ensemble fini de sommets ;
- Σ est un ensemble fini d'événements ;
- T est un ensemble de transitions entre sommets. Une transition est un triplet $t_i = (q, \sigma, q')$, où q est un sommet source ; σ est un événement de Σ et q' est un sommet but.
- q_0 est l'état initial de l'automate.

Dans un automate à états finis chaque sommet représente un état SED. La transition $t_i = (q, \sigma, q')$ traduit le passage du système du sommet q vers le sommet q' suite à l'occurrence de l'événement σ .

✓ **Exemple :**

La figure suivante modélise une machine simple qui peut prendre trois états : A (arrêt), M (marche) et P (panne) ainsi $Q = \{A, M, P\}$. La machine est initialement en arrêt ce qui est repéré par une flèche entrante. On considère l'état M (marche) comme l'état sortie, il est donc représenté avec un cercle double. La transition d'un état à un autre se fait par occurrence de l'un des quatre événements : début du travail (α), fin du travail (β), panne de la machine (γ) et réparation de la machine (λ).

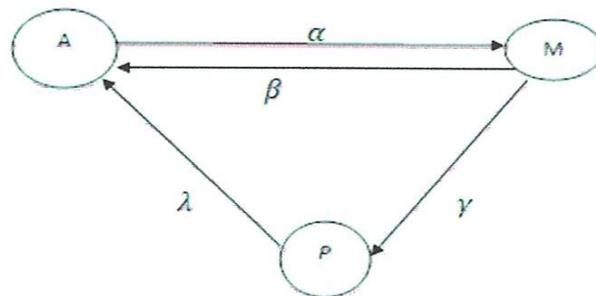


Figure 1.10 : Automate à états finis

➤ Automates Hybrides

Les automates hybrides ont été introduits par **Alur** [1], comme une extension des automates à état finis. Il s'agit d'étendre les automates à états par un ensemble de variables réelles $X = \{x_1, x_2, \dots, x_n\}$. A chaque sommet q d'un automate hybride associe pour chaque variable continue de la forme :

$$f = (\dot{x}(t), x(t)) = 0 \quad (1.9)$$

Une fonction d'évolution f_q encore dite activité, et un prédicat sur la valeur des variables appelée invariant du sommet.

Le système peut séjourner dans un sommet tant que les valeurs des horloges vérifient l'invariant associé. Le franchissement d'une transition est instantané, à chaque transition on associe une condition de franchissement, appelée garde, et une affectation. Une transition ne peut être franchie que si sa garde est vérifiée par les valeurs des variables. Les gardes modélisent les contraintes imposés au comportement du système, et l'affectation associée à une transition désigne les variables qui effectuent un saut suite au franchissement de la transition.

Définition formelle :

Un automate hybride est un sextuple $A = (Q, X, E', T, F, Inv)$ tel que :

- $Q = \{q_1, q_2, \dots\}$ est un ensemble fini de sommets ;
- $X \in R^n$ est un vecteur d'état comportant n variable réelles ;
- Σ est un ensemble fini d'événements ;
- T est un ensemble fini de transitions, chaque transition est un quintuple $t_i = (q, \sigma, g, \gamma, q')$ tel que :
 - ❖ $q \in Q$ est le sommet source ;
 - ❖ $\sigma \in \Sigma$ est un évènement associé à la transition t_i ;
 - ❖ g est la garde de la transition t_i , c'est un prédicat sur x ; la transition ne peut être franchie que si sa garde g est vérifié ;
 - ❖ $q' \in Q$ est le sommet du but ;
- F est une fonction qui associe à chaque sommet q une fonction continue f_q qui représente l'évolution dynamique du vecteur d'état dans le sommet ;
- Inv est une fonction qui associe à chaque sommet q , un prédicat $Inv(q)$ qui doit être vérifié par les valeurs des variables continues lors du séjour de l'automate dans le sommet q .

✓ **Exemple :**

L'automate hybride de la figure suivante modélise, d'une manière plus riche, la machine simple présentée précédemment dans la figure précédemment. Deux variables d'état continues sont utilisées, x pour spécifier la durée opératoire de la machine qui est 3 *u.t.* et Y pour le vieillissement de la machine, une panne ne peut surgir avant que la variable Y atteigne la valeur 7, depuis sa dernière action de réparation

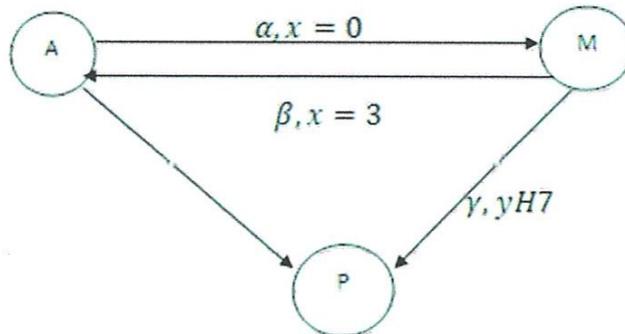


Figure 1.11 : Automate hybride modélisant une machine simple

1.8 Conclusion

Les systèmes dynamiques hybrides concernent de nombreux secteurs de l'industrie, mais aussi des transports, voire d'autres activités humaines. Un système dynamique hybride est un système comportant une partie à évolution continue dans le temps et une partie à évolution événementielle.

Les méthodes étudiées dans ce chapitre s'imprègnent des résultats issus de l'automatique des systèmes continus et de celle des systèmes à événements discrets mais aussi de l'informatique, pour étudier les solutions à des problèmes liés à la complexité des systèmes à commander ou à modéliser. L'application de la théorie des systèmes dynamiques hybrides aux outils et moyens de conduite opérationnels disponibles dans le monde industriel permettra d'améliorer le diagnostic, la sûreté des performances et les techniques de développement de systèmes réels. C'est le but du chapitre 2.

Chapitre 2 :

Diagnostic d'un système dynamique
hybride

2.1 Introduction

Le diagnostic est le raisonnement menant à l'identification de la cause (*l'origine*) d'une défaillance, d'un défaut ou d'une maladie à partir des caractères ou symptômes relevés par des observations, des contrôles ou des tests ; le mot diagnostic vient du mot grec « diagnosi » construit à partir de « dia » qui signifie à travers et de « gnosi » qui signifie connaissance, discernement.

Il s'agit d'acquérir la connaissance et de produire une décision à travers les signes observables. Pour nous le diagnostic consiste à détecter, localiser et éventuellement identifier le (*ou les*) défaut(s) affectant le système étudié.

2.2 Terminologie

Avant d'aller plus loin, il convient de définir certaine terminologie ou termes auxquels nous aurons souvent recours dans la suite.

- **Fonctionnement normal d'un système** : consiste à indiquer si au moins un défaut s'est produit [37].
- **Localisation de défauts** : Détermination des types, le (*les*) élément(s) en défaut et de l'instant d'occurrence d'un défaut, c'est-à-dire vise à préciser quel(s) défaut(s) s'est produits.
- **Dysfonctionnement** : Une irrégularité intermittente survenant au niveau d'une fonction remplie par le processus [36].
- **Défaillance** : Une défaillance définit une anomalie fonctionnelle au sein d'un système physique, c'est-à-dire caractérise son incapacité à accomplir certaines fonctions qui lui sont assignées.
- **Défaut** : Cette notion se rapporte à une anomalie de comportement au sein d'un système physique, c'est-à-dire l'écart entre la caractéristique observé sur le dispositif et la caractéristique de référence. Toutefois, un défaut n'implique pas nécessairement une défaillance donc n'altère pas le fonctionnement d'un système.
- **Panne** : Inaptitude d'un dispositif à accomplir une fonction requise. Une panne résulte toujours d'une défaillance et donc d'un défaut :

Défaut → **Défaillance** → **Panne**

- **Perturbation** : C'est une entrée non contrôlée autrement dit un phénomène considéré comme normal influant sur système physique, mais non ou mal pris en compte dans le modèle censé le représenter.
- **Symptôme analytique** : Changement dans les caractéristiques nominales d'une quantité observable provoqué par un défaut.
- **identification de défauts** : Détermination de la taille et du comportement temporel d'un défaut, autrement dit l'estimation des caractéristiques statiques et dynamiques de ce défaut : amplitude permanente, évolution dynamique du défaut.
- **Incertitudes (*uncertainty*)** : Ce sont les approximations numériques, les erreurs d'identification des paramètres, la réduction de modèle, la non prise en compte des paramètres variant dans le temps. Les incertitudes peuvent être structurelles ou paramétriques.
- **Effet de défaillance** : Conséquence d'un mode de délivrance sur l'opération, la fonction, ou le statu d'une variable.
- **Diagnostic**: Détermination du type, de la taille, de l'endroit et de l'instant d'occurrence d'un défaut ; il suit la détection de défauts et induit la localisation et identification.
- **Modélisation du défaut** : Détermination d'un modèle mathématique pour décrire un effet spécifique du défaut.
- **Résidu** : Signaux portant de l'information, soit un indicateur de défaut basé sur l'écart entre les mesures et les calculs basés sur le modèle.
- **Seuil** : Valeur limite de l'écart d'un résidu par rapport au zéro, ainsi s'il est dépassé, un défaut est déclaré comme détecté.

2.3 Classification des défauts

Les défauts sont des événements qui apparaissent à différents endroits du système. Ils peuvent être répartis en trois classes :

- **Défauts capteurs :**

Ce type de défaut est la cause d'une mauvaise image de l'état physique du système. Un défaut capteur partiel produit un signal avec plus ou moins d'adéquation avec la valeur vraie de la variable à mesurer.

Ceci peut se traduire par une réduction de la valeur affichée par rapport à la valeur vraie, ou de la présence d'un biais ou de bruit accru empêchant une bonne lecture. Un défaut capteur total produit une valeur qui n'est pas en rapport avec la grandeur à mesurer ($0, \infty$).

- **Défauts actionneurs :**

Modélisés en général comme des signaux additifs ou multiplicatifs aux signaux d'entrées. Ils agissent au niveau de la partie opérative et détériorent ainsi le signal d'entrée du système. Ils représentent une perte totale (*défaillance*) ou partielle d'un actionneur agissant sur le système. Un exemple de perte totale d'un actionneur est un actionneur qui est resté "collé" sur une position entraînant une incapacité à commander le système par le biais de cet actionneur.

Les défauts actionneurs partiels sont des actionneurs réagissant de manière similaire au régime nominal mais en partie seulement, c'est-à-dire avec une certaine dégradation dans leur action sur le système. (*Perte de puissance d'un moteur, fuite dans un vérin . . .*).

- **Défauts systèmes :**

Ce type de défaut provient du système lui-même, bien souvent les défauts n'appartenant pas à un défaut capteur ou actionneur sont classés de manière arbitraire dans cette catégorie. Néanmoins, un défaut composant résulte de la casse ou de l'altération d'un composant du système réduisant les capacités de celui-ci à effectuer une tâche.

En pratique, ceci revient à considérer une modification des caractéristiques du système proprement dit. (*La CTN : résistance à Coefficient de Température Négatif, d'une chaufferie est cassée, un roulement est altéré*).

Une autre classification de défauts selon leur comportement dans le temps peut être considérée :

- **Défaut d'abrupt :**

Ce défaut a un comportement temporel discontinu :

$$f(t - t_f) = \begin{cases} \delta & t \geq t_f \\ 0 & t < t_f \end{cases} \quad (2.1)$$

Où $f(t - t_f)$ représente le comportement temporel du défaut, t_f représente l'instant d'occurrence du défaut et δ un seuil constant. Ce type de défauts est caractérisé par une discontinuité dans l'évolution temporelle de la variable.

Cette évolution, si elle ne correspond pas aux évolutions dynamiques normales attendues pour la variable, est caractéristique d'une panne brutale de l'élément en question : arrêt total ou partiel, déconnexion.

- **Défauts intermittents :**

Ce défaut est un cas particulier de défaut abrupt avec la propriété que le signal revient de façon aléatoire à sa valeur normale. Ce type de défauts caractérise les faux contacts.

- **Défaut graduel :**

Ce défaut a un comportement temporel lent :

$$f(t - t_f) = \begin{cases} \delta(1 - e^{-\alpha(t-t_f)}) & t \geq t_f \\ 0 & t < t_f \end{cases} \quad (2.2)$$

Où α et δ sont des constantes positives. Il est très difficile à détecter car son évolution temporelle a la même signature que celle d'une modification paramétrique lente représentant une non-stationnarité du procédé. Ce type de défauts est caractéristique d'un encrassement ou d'une usure de pièces.

2.4 Performance d'une procédure de diagnostic

Pour que le système de diagnostic développé soit le plus efficace possible, il nous faut en premier lieu décrire quelques critères permettant d'évaluer un système. Nous allons dans les lignes qui suivent regrouper les différents critères de performance du système de détection de la manière :

- **Défectabilité** : C'est l'aptitude du système de diagnostic à pouvoir déceler la présence d'une défaillance sur le prédécédé.
- **Isolabilité** : C'est la capacité du système de diagnostic à remonter directement à l'origine du défaut.
- **Sensibilité** : Caractérise l'aptitude du système à détecter des défauts d'une certaine amplitude.
- **Robustesse** : elle détermine la capacité du système à détecter des défauts indépendamment des erreurs de modélisation.
- **Coût économique,**
- **Temps de développement.**

2.5 Méthodes de diagnostic des systèmes dynamiques hybrides

L'efficacité d'une méthode de diagnostic de SDH peut se mesurer par sa capacité à exploiter d'une manière optimale, les deux aspects présentés par ces systèmes : l'aspect continu à travers les variables d'état continues et l'aspect discret à travers les événements discrets. En effet, la détection d'une anomalie sur une variable d'état continue est réalisée lorsque celle-ci dépasse un certain seuil et sur une variable discrète lorsque l'occurrence d'un événement inattendu se produit ou lorsqu'un événement attendu ne se produit pas.

2.5.1 Méthodes à base de modèle

Le diagnostic à base de modèle est basé sur le test de cohérence qui consiste à vérifier les contradictions entre les observations et les prédictions effectuées par le modèle de référence, en ne se contentant pas seulement d'indiquer la présence de défauts (*détection*) mais également en renseignant sur la localisation de ces défauts [17] ; [8].

Dans le cadre des systèmes continus le test de cohérence dans le cas du diagnostic à base de modèle analytique est liée à l'étape de génération de résidus qui a pour but de générer, à partir d'un modèle de bon fonctionnement ou de fonctionnement défaillant du processus et des mesures disponibles des signaux révélateurs de la présence de défauts, appelés résidus [12].

Dans les conditions idéales (*pas de perturbation*), les résidus sont égaux à zéro et prennent des valeurs significatives quand le défaut se produit. Cependant à cause des perturbations des bruits qui agissent sur le système et des erreurs de modélisation un résidu n'est exactement égal à zéro. Le résidu devrait préserver une robustesse suffisante par rapport aux changements de modes de fonctionnement du système.

Dans le cadre des systèmes à événements discrets le test de cohérence consiste à vérifier les séquences d'événements prévues par le modèle avec les séquences d'événements observées grâce aux capteurs discrets. Toute discordance entre les événements observés et les événements prédits grâce au modèle indiquera la présence d'au moins un défaut [39] ; [40].

2.5.2 Contribution fondées sur des approches continues

Ces méthodes se basent généralement sur des approches de diagnostic issues de la communauté des systèmes continus, telles que la génération des résidus et le raisonnement causal [16] ; [28] ; [5] ; [26] ; [14]. L'exploitation de l'aspect discret des SDH est généralement faible dans ces approches.

2.5.2.1 Approches à base de techniques de génération de résidus

Il existe plusieurs approches de diagnostic basées sur la génération de résidus tel que nous pouvons citer à titre d'exemple l'approche de **Koutsoukos** [28]. Dans cette approche l'estimation en ligne du mode de fonctionnement des SDH sont réalisée grâce à l'emploi de RdP, un défaut est détecté en comparant les grandeurs mesurées à celles attendues en tenant compte des commandes envoyées au processus.

Le diagnostic de l'état du processus est ensuite réalisé à l'aide d'un arbre logique, cette approche adaptée aux SDH car elle présente l'avantage de détecter les anomalies dues aux variables continues et à l'occurrence d'événement perturbateur.

Une autre technique de diagnostic de SDH a été proposée par **Cocquemot** [11]. En effet, cette technique repose sur les méthodes de diagnostic à base de **redondance analytique**. Le système à diagnostiquer est modélisé par un automate hybride, la détection des défauts est réalisée grâce à la génération des résidus entre les variables d'entrée et de sortie mesurées et les relations de redondance analytique déterminées à partir des entrées et des sorties ainsi que leurs dérivées indépendamment du mode discret du système.

Le diagnostic des défaillances est réalisé à partir de résidus structurés spécifiques à chaque défaut. Dans l'approche de **Balluchi** [5], une solution basée sur l'utilisation d'un observateur hybride constitué d'un observateur continu et un observateur discret est proposée. L'observateur continu estime l'évolution des variables continues tandis que l'observateur discret permet d'identifier l'état discret courant du système.

2.5.2.2 Approche à base de raisonnement causal :

Gomaa propose dans [19] ; [20] une approche de diagnostic des SDH fondée sur une extension des **RdP** hybrides, dénommé les réseaux de Petri continus canaux hybrides (**RdPC²H**). Ce modèle intègre trois types de **RdPs** : un **RdP** continu temporisé qu'on a appelé Réseau de Petri Réseau de Petri continu causal hybride (**RdPC²H**) c'est un modèle approximatif modélisant le système de contrôle (**SDE**) ; et un **RdP** classique modélisant l'interaction entre la partie continues et le système de contrôle.

Les liens causaux (*transitions*) entre les variables continues sont représentés à travers des fonctions de transfert qualitatives basées sur les informations de gain, retard... L'auteur propose la conception d'un système de diagnostic de défauts basée sur un modèle. Le système de détection de défauts influençant les variables continues est réalisé de façon asynchrone, la localisation de défauts est effectuée par le chaînage arrière/avant des liens causaux entre les variables en utilisant leurs informations temporelles.

Une autre approche reposant sur un raisonnement causal a été proposé par **Karsai** [26], cette approche repose d'abord sur la modélisation du système par un modèle bond graph hybride de **Mosterman** [35], puis la génération d'un graphe de propagation des défauts qui permet de décrire les relations causales et temporelles entre les différents modes de défauts d'un côté et les observations associées d'un autre.

2.5.3 Contribution fondées sur des approches discrètes :

IL existe peu de contributions de diagnostic pour les SDH, issues de la méthode discrète. Nous pouvons citer deux grandes approches dans ce cadre, chacune détermine à sa manière l'acquisition du modèle hybride.

La première approche est fondée sur l'abstraction des trajectoires continues et sa représentation à travers des modèles discrets dynamiques, elle consiste à établir des

2.5.3 Contribution fondées sur des approches discrètes :

IL existe peu de contributions de diagnostic pour les SDH, issues de la méthode discrète. Nous pouvons citer deux grandes approches dans ce cadre, chacune détermine à sa manière l'acquisition du modèle hybride.

La première approche est fondée sur l'abstraction des trajectoires continues et sa représentation à travers des modèles discrets dynamiques, elle consiste à établir des partitions qualitatives des grandeurs continues et de les représenter sous la forme d'états ou d'étiquettes de transition d'état.

Cette approche représente une solution alternative aux approches fondées sur une représentation analytique précise. Parmi les approches développées dans ce cadre nous citons les travaux de **Bhowal** [8] qui se base sur un modèle automate hybride à temps discret et la deuxième approche se repose sur une acquisition expérimentale du modèle du système à travers l'abstraction des équations de l'espace d'état et les méthodes d'identification à base de réalisation expérimentales. Parmi les approches développées dans le cadre nous citons les travaux de **Lunze**.

2.5.3.1 Approche de diagnostic de Lunze

Lunze propose dans [29] ; [30] des algorithmes de diagnostic similaires au concept du diagnostiqueur de Sampath. Ainsi l'inférence et l'évaluation des hypothèses sur les défauts sont effectuées à travers un algorithme de diagnostic, comme il est indiqué dans la figure (2.1). Le problème de diagnostic est associé à un problème d'observation d'état qualitatif.

En effet une abstraction qualitative des variables continues du système est effectuée à travers l'utilisation de quantificateurs. L'identification de défaut consiste alors à trouver le modèle f_i dont la trajectoire prédite est cohérente avec celle observée. La localisation est une conséquence de l'identification qui associe à chaque f_i un composant ou sous-système. Si la discrimination entre les défauts est faible alors une réévaluation de l'abstraction qualitative faite ou bien une réévaluation du schéma d'instrumentation.

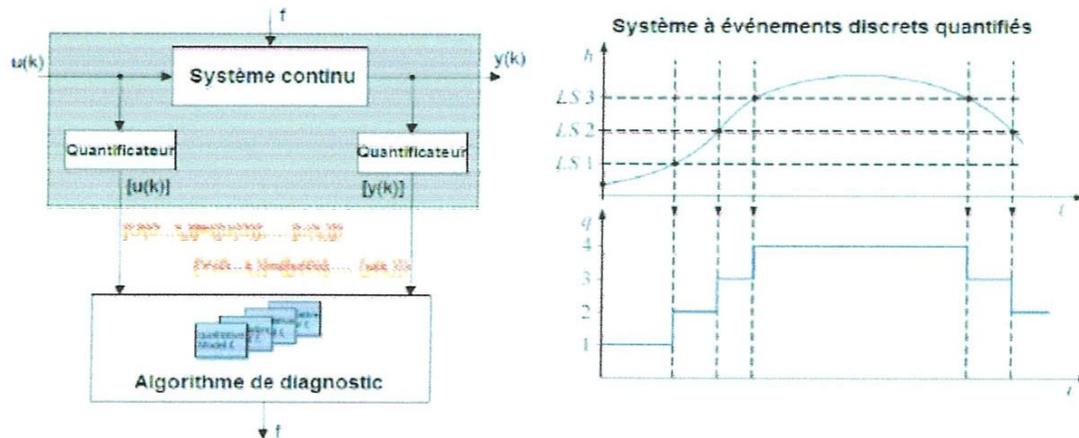


Figure 2.1 : Principe de diagnostic de SDH selon l'approche de Lunze

2.5.3.2 Approche à base de modèle hybride à temps discret

Une extension de l'approche de Sampath pour le diagnostic des systèmes hybrides à temps discret a été proposée dans les travaux de BHOWAL [8]. Le modèle considéré évolue comme un automate temporisé à n différents flux [2]. Il est représenté par un graphe de transition d'activité où chaque activité correspond à un sommet du graphe. L'évolution des variables continues dans chaque activité est synchronisée sur l'occurrence d'un événement spécial appelé **tick** ou qui se produit après chaque période constante. Sur l'occurrence d'un **tick** chaque variable évolue avec un pas constant défini en fonction de sa dynamique. Les variables sont réparties en des variables **discrètes** (ou événements) observables et non observables et des variables continues mesurables et non mesurables.

L'approche de diagnostic commence par modéliser les composants du système en utilisant les graphes de transition d'activités, ensuite obtenir le modèle global par la composition de ces modèles. Le diagnostiqueur est un graphe de transition d'activité compilé hors-ligne. Chaque sommet du diagnostiqueur contient une estimation de l'état du système et des étiquettes de l'ensemble $\{N, F_1, \dots, F_m\}$ pour indiquer la présence de défauts.

Cette approche est basée sur l'abstraction des dynamiques continues et la discrétisation du temps. En effet elle présente le même inconvénient que l'approche du diagnostic des SED à temps discret de Zad [45]. Cet inconvénient correspond au problème d'explosion combinatoire de l'espace d'état du modèle utilisé ainsi que celui du diagnostiqueur construit. Ce problème rend difficile la mise en œuvre de cette approche pour les systèmes complexes.

2.5.4 Contribution fondées sur des approches mixtes :

Comme nous l'avons vu, les méthodes issues des systèmes continus et celles issues des SED privilégient l'un des deux aspects du SDH au détriment de l'autre. Or, bien qu'un diagnostic "efficace" d'un SDH nécessite une prise en compte des deux dynamiques à la fois, il n'existe, à notre connaissance, que peu de travaux s'inscrivant dans ce cadre. A titre d'exemple, une approche reposant sur un raisonnement causal a été proposé dans Karsai [26]. Cette approche se base d'abord sur la modélisation du système par un modèle bond-graph hybride de Mosterman [35], puis la génération d'un graph de propagation des défauts, qui permet de décrire les relations causales et temporelles entre les différents modes de défauts d'un cote, et les observations associées d'un autre.

En outre, une méthodologie de diagnostic basée sur les **GPFN2** et **GPDF3** (*automate hybride*) a été proposée par **El Meznyani** [15]. L'idée consiste à surveiller les équations d'état (*contraintes égalité*), les conditions d'invariant (*contraintes inégalité*) et l'état discret du système. Les tâches de l'identification du mode de fonctionnement actif, de la détection et de la localisation des défaillances étaient basées sur les propriétés structurelles des résidus robustes, vis-à-vis des entrées inconnues, issus des **RRA4**.

Dans la même optique, **Derbel** [13] propose une démarche de diagnostic pour une classe de SDH, représentée par une sous-classe d'automate hybride : les automates rectangulaires. Ce formalisme permet d'approximer, sous des conditions très restreintes, les comportements de la dynamique continue du SDH à travers l'utilisation des conditions de flux rectangulaires. L'idée de cette approche repose sur la conception d'un diagnostiqueur en ligne, qui estime l'état hybride courant du système ainsi que les occurrences de défauts.

A partir de ces estimations, le diagnostiqueur décide si un défaut s'est produit ou non. De même, **Bayoudh** et **Trave-Massuyes** [6] ont reformulé les problèmes de diagnostic des SDH, comme étant l'identification non ambiguë, tout instant, du mode du SDH (*nominal ou défaillant*). Pour cela, la partie continue du SDH a été diagnostiquée par l'extension de l'approche espace de parité [11]. Les résidus qui en résultent sont utilisés par le diagnostiqueur pour l'identification du mode actif.

Par ailleurs, une autre approche basée sur les Réseaux de Pétri Différentiels Objet (**RdPDO**) et les filtres de **Kalman** étendus a été proposée par Olivier-Maget, où les **RdPDO** ont été utilisés pour la modélisation et le diagnostic d'un procédé chimique ayant un comportement hybride. Le filtre de Kalman étendu permet l'estimation de l'état continu dans le mode de fonctionnement actif, en éliminant l'effet des perturbations (*bruit, précision de modèle...*).

En revanche, la convergence du filtre de Kalman n'est aucunement assurée car il s'agit d'une convergence locale.

En résumé, bien que les travaux présentés dans cette catégorie de méthodes considèrent conjointement les parties continue et discrète, les problèmes de la robustesse vis-à-vis des entrées inconnues et de la sensibilité aux défauts des résidus méritent d'être encore approfondis.

2.6 Conclusion

Dans ce chapitre, nous avons proposé un tour d'horizon sur les méthodes de diagnostic à base de modèles des systèmes à événements discrets, continu et des systèmes dynamiques hybrides. Nous avons commencé par préciser un certain nombre de notions fondamentales ainsi que les principes d'un système de diagnostic à base de modèles.

Nous avons par la suite classifié ces méthodes selon différents critères, parmi lesquels, le modèle utilisé pour représenter le comportement du système, la représentation du temps : modèles à approche continue, discrète et mixte.

Chapitre 3 :

Modélisation des systèmes dynamiques
hybrides

3.1 Introduction :

Afin d'étudier les SDH, les différentes approches développées pour le système continu et le système discret ont pu être combinées pour définir une modélisation hybride. D'une part les équations différentielles (*linéaires ou non linéaires*), liées au processus physique, permettent de représenter l'état continu... Et d'autre part un automate ou un réseau de pétri modélise l'état discret du système.

Généralement afin de concevoir une loi de commande un observateur ou un algorithme de diagnostic pour un SDH une modélisation par un automate hybride peut être utilisée. Un automate hybride se définit (*d'une manière non formelle*) par un ensemble fini d'états discrets appelés modes. Chaque mode est associé aux équations différentielles propres au type de fonctionnement du système. Un ensemble fini de transitions entre les modes est défini pour traduire les possibles passages d'un mode à un autre. Le franchissement d'une transition est géré par loi de commutations.

3.2 Structure générique des SDH :

D'une manière générale, en vue de la commande, un système dynamique peut être décrit en termes de système à commander, représentant la **méthode de fonctionnement** ou **procédé**, et par ses **spécifications de fonctionnement**, décrivant le fonctionnement désiré du système en boucle fermée.

3.2.1 Méthode de fonctionnement :

Les systèmes dynamiques hybrides sont des systèmes dont le comportement dynamique est défini par l'interaction entre ses dynamiques continues et discrètes. Par conséquent, la méthode de fonctionnement lui aussi possèdera les deux approches :

- approche continu,
- approche discret.

3.2.1.1 Approche discret :

D'une manière générale, l'évolution discrète d'un système dynamique hybride est déterminée par l'occurrence des évènements d'origine externes ou internes dont la

nature peut être contrôlable ou non. L'occurrence des événements implique des changements de la dynamique continue du système.

Dans les systèmes dynamiques hybrides, ces modifications peuvent être soit par une structure particulière du procédé physique, soit par des entrées/sorties discrètes générées par différentes composantes du système.

3.2.1.1.1 Modification de dynamique continue causées par des sorties discrètes :

Dans bon nombre de système réel, les capteurs sont utilisés pour signaler aux opérateurs les dépassements de certains seuils qui peuvent générer un fonctionnement non désiré du système.

Ainsi, leur utilisation est étroitement liée à la manière dont les spécifications de fonctionnement du système sont formulées. Les informations qu'ils fournissent sont explicitement prises en compte lors de la synthèse d'un modèle de commande.

La remarque concernant la nature incontrôlable des événements générés par des capteurs en cas de détection d'un seuil, par exemple, reste toujours vraie.

3.2.1.1.2 Modification de dynamique continue causées par des entrées discrètes :

Les éléments fréquemment utilisés dans les systèmes réels qui peuvent introduire des discontinuités dans leur fonctionnement sont les actionneurs discrets (*les vannes avec leurs états correspondants : ouvert ou fermé*). En général, ce sont des éléments auxquels sont associées des fonctions de contrôle système (*la commande manuelle/automatique de fermeture/ouverture de vannes*).

Pour pouvoir intégrer des actionneurs discrets dans le modèle global d'un processus, il faut d'abord définir les concepts de modélisation qui leur correspondent. Les événements modélisant un changement d'état du système seront forcément des événements dont la nature est contrôlable pour pouvoir agir sur leur date d'occurrence.

De plus, dans ce cas s'y les événements sont utilisés pour modéliser ou concevoir les états des actionneurs. Il s'agit donc de modéliser les états discrets du système global et de définir les transitions entre ces états. Ceci permettra de construire un modèle global de fonctionnement du système.

3.2.1.1.3 Modification de dynamique continue causées par la structure du système :

Nous allons présenter dans les lignes qui suivent une utilisation des capteurs qui permet la modélisation de discontinuités dans l'évolution du système.

Dans le cas des systèmes réels, les variables d'état $x(t)$ peuvent modéliser des variations de volume, de température ou bien de concentration. Le champ de vecteur $f(x(t))$ est en général, une fonction continue mais parfois il peut présenter des discontinuités. Celles-ci reflètent des changements de la dynamique continue dus aux caractéristiques du procédé physique. Pour illustrer ce phénomène, considérons un réservoir représenté par la figure 3.1.

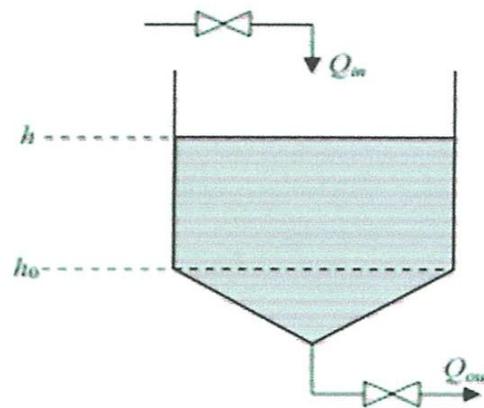


Figure 3.1 : Exemple d'une structure physique

La forme du réservoir implique un changement de modèle mathématique, qui modélise la variation du niveau de fluide dans le réservoir, lorsque le niveau est au-dessous du seuil h_0 . Ainsi, si le niveau dans le réservoir respecte la relation $h > h_0$, l'équation modélisant la vitesse de variation du niveau dans le réservoir est donnée par :

$$S\dot{h}(t) = Q_{in} - Q_{out} \quad (3.1)$$

Où :

Q_{in} et Q_{out} : Représentent les débits d'entrée et de sortie respectivement.

h et h_0 : Les niveaux de fluide.

S : La section du réservoir.

Si $h < h_0$ alors le niveau dans le réservoir varie en conformité avec la dynamique décrite par :

$$S\left(\frac{h(t)}{h_0}\right)^2 \dot{h}(t) = Q_{in} - Q_{out} \quad (3.2)$$

Dans ce cas, les équations différentielles modélisant la variation de niveau dans le réservoir ne présentent pas de discontinuités, cependant dans la dynamique du système on distingue deux comportements différents. Pour différencier ces deux comportements, la solution repose sur l'utilisation d'une variable discrète associée à chaque dynamique continue.

Dans les procédés réels, cette distinction peut se faire en utilisant un capteur qui détecte le seuil où le changement de modèle intervient. Dans l'exemple considéré, pour distinguer les deux comportements du système, l'utilisation d'un capteur qui détecte le seuil h_0 peut fournir cette information.

Les changements de comportement dont l'origine se trouve dans la structure physique du système correspondent au phénomène de commutation autonome. L'utilisation des capteurs, pour modéliser d'une manière explicite les discontinuités introduites par la structure physique du système, implique naturellement l'occurrence d'un événement généré au moment où le changement du comportement continu du système intervient. De tels événements sont des événements incontrôlables dans le sens où leur occurrence ne peut pas être empêchée.

3.2.1.2 Approche continu :

L'évolution dynamique d'un système est déterminée souvent par les processus physiques qui ont lieu. Ainsi, le modèle mathématique est obtenu à partir des propriétés physiques du système permettant de trouver une description de celui-ci sous la forme de représentation d'état. Le modèle mathématique général est du type :

$$\dot{x}(t) = f(x(t), u(t), t) \quad (3.3)$$

Où $u(t)$ représente le vecteur d'entrées et $x(t)$ est le vecteur des variables d'état.

Le modèle présenté par la relation (3.3) est assez général et l'analyse de ses propriétés doit être réalisée en temps continu.

3.2.2 Spécification de fonctionnement :

Globalement, les spécifications introduisent des restrictions dans l'évolution du procédé. Ainsi, pour formuler le problème de synthèse de la commande, il est nécessaire qu'après la description du procédé (*système à commander*), une description de ses spécifications de fonctionnement soient faite.

L'objectif de la synthèse consistera à restreindre l'évolution du procédé telle que le fonctionnement en boucle fermée du procédé couplé avec son système de commande respecte toujours les spécifications imposées. Dans le cas des systèmes hybrides, les restrictions imposées par des spécifications du système peuvent être décrites en les divisant en deux groupes :

- les spécifications correspondant à la partie **continue** du système.
- les spécifications correspondant à la partie **discrète** du système.

3.2.2.1 Spécification discrète :

La partie discrète d'un système hybride peut être vue comme une machine à états finis. En général, les spécifications 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**.

3.2.2.2 Spécification continue :

Les spécifications continues correspondent à des restrictions imposées sur les valeurs des variables d'état continues \dot{x} , exprimées par des conditions logiques qui limitent leur évolution à une certaine région de l'espace d'état, appelée région désirée. Toute évolution sortant de cette région est non désirable. Si nous utilisons des automates pour modéliser le système étudié, l'**évolution non désirée** sera représentée par un sommet appelé **sommet interdit**.

✓ Exemple :

Considérons la partie d'un automate hybride présentée dans la figure

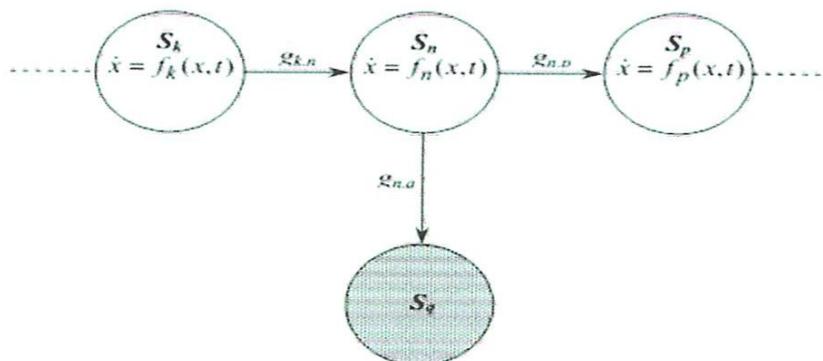


Figure 3.2 : Partie d'un automate hybride

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 que l'évolution discrète est réalisée par le franchissement des transitions (*arcs*) du graphe. Le sommet interdit S_q est atteint depuis le sommet S_n par la validation de la transition étiquetée par $g_{n,q}$.

En effet, un **sommet interdit** modélise la situation où les spécifications continues du système ne sont plus vérifiées.

✓ **Exemple :**

Considérons thermostat utilisé pour maintenir la température du procédé dans un intervalle imposé.

Le système de chauffage sera arrêté ou redémarré en fonction des informations envoyées par le capteur de température du procédé. L'intervalle de température imposé, représente la spécification continue. Le dépassement des seuils de cet intervalle peut impliquer des défaillances dans le système, provoquées par exemple par le sur chauffage des différentes composantes. Les états interdits permettent la modélisation d'une telle situation. Ainsi, le modèle automate où les évolutions non désirées sont représentées est illustré dans la figure III.3

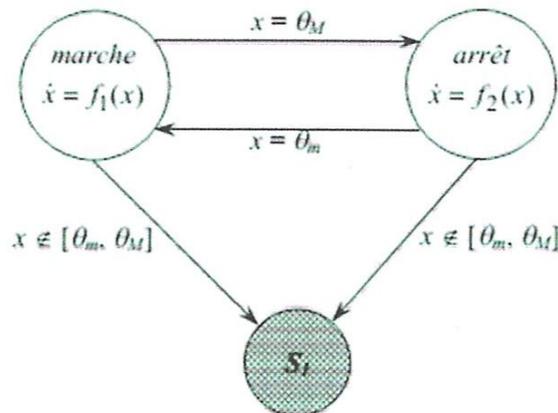


Figure 3.3 : Modèle de thermostat avec spécification de fonctionnement

3.3 Conception ou Elaboration des SDH :

De manière générale, un SDH sera élaboré ou conçu par un ensemble de système à dynamique continue interagissant avec un ou plusieurs systèmes à événements discrets. Il y a en général trois classes principales des approches de conception des systèmes dynamiques hybrides (SDH) :

- approche continue,
- approche événementielle,
- approche mixte.

3.3.1 Approche événementielle :

Cette approche est purement discrète, elle consiste à concevoir ou modéliser les systèmes hybrides en supprimant les dynamiques continues ou à faire une approximation de l'évolution continue de façon à ce que le système hybride soit représenté uniquement par les événements qui le caractérisent. La modélisation événementielle d'un SDH permettra ainsi de faire appel à la théorie classique de superviseur des SED (*Systèmes à Evènement Discret*) pour la synthèse d'un modèle de commande.

3.3.2 Approche continue :

Contrairement à l'approche événementielle dans cette approche, il s'agit avant tout d'étudier le comportement des modèles continus en présence des discontinuités et éventuellement de définir un modèle étendu, c'est-à-dire qu'elle consiste à définir une approximation des dynamiques discrètes du système hybride par des équations différentielles (*aux différences*) pour modéliser l'occurrence des événements discrets. L'idée est qu'en utilisant une approche unifiée dans le domaine des systèmes continus, où les théories sont bien établies (*les questions de stabilité, de commandabilité et d'observabilité...*) pourront être étudiées selon les théories classiques.

3.3.3 Approche mixte :

L'approche mixte repose sur la supposition que le fonctionnement d'un système hybride est une séquence de deux phases. La première étape correspond à une transformation de l'état continu décrite en termes de temps écoulé durant cette phase. Dans la seconde étape, l'état est soumis à un changement discret instantané. Ainsi, les modèles développés dans le cadre de cette approche reposent sur l'interaction de deux sous-modèles, l'un pour les aspects événementiels, basé sur les automates à états finis, les réseaux de Pétri ou des extensions de ces formalismes, et l'autre, formalisé par des équations d'état (*souvent par des équations différentielles*) pour les aspects continus. Chacun des aspects, continu ou événementiel, est ainsi décrit sous une forme classique et l'aspect hybride est clairement pris en compte dans l'interface entre les deux sous-modèles.

L'aspect événementiel influe sur le modèle continu en validant certain des équations continues en fonction de l'état discret actif et l'aspect continu agit sur le

modèle évènementiel en validant ou en forçant le franchissement de certaines transitions.

Parmi les outils de modélisation résultant de cette approche mixte, on retrouve : les automates hybrides [2] représentant le modèle formel fondamental de cette approche, les statecharts hybrides pour apporter des solutions aux problèmes posés par la spécification des modèles, en particulier de la structuration hiérarchisée [43], et enfin les différents extensions des réseaux de Pétri [24], [38].

3.4 Les outils de modélisation :

3.4.1 Automates Hybrides :

Un automate hybride est un outil de représentation qui permet de tenir compte explicitement des deux évolutions continue et discrète du SDH. Il apparaît comme l'association d'un automate à états finis pilotant un ensemble d'équations dynamiques continues. Les équations modélisant le comportement continu à un instant donné dépendent de l'état de l'automate mais ce dernier peut évoluer en fonction de la valeur des grandeurs continues [1].

Un automate hybride est un graphe composé de sommets (*places*) et d'arcs orientés modélisant les transitions discrètes qui relient les sommets. Tout arc orienté doit avoir un sommet destination.

✓ **Exemple :**

$$\text{Mode 1: } \dot{x}(t) = a_1x(t) + b_1$$

$$\text{Mode 2: } \dot{x}(t) = a_2x(t) + b_2$$

$$a_1, a_2, b_1, b_2 \in \mathbb{R}$$

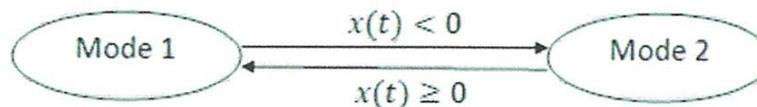


Figure 3.4 : Automate hybride

En effet, cette modélisation permet de représenter à la fois l'évolution de l'état discret (*entre les modes 1 et 2*) et la dynamique de l'état continu $x(t)$ associé à chaque mode.

3.4.2 Réseaux de Petri

Les réseaux de Petri représentent un formalisme puissant pour la modélisation et l'analyse des systèmes à événements discrets comme les systèmes de télécommunication, les réseaux de transports, les systèmes automatisés de production, etc. Leur représentation graphique permet de visualiser d'une manière naturelle le parallélisme, la synchronisation, le partage de ressources et le non déterminisme, leur représentation mathématique permet d'établir les équations d'état à partir desquelles il est possible d'apprécier les propriétés du modèle et les comparer au comportement du système modélisé [7].

Depuis leur apparition, les RdP (*réseau de Petri*) ont connu nombreuses extensions notamment pour la prise en compte des aspects stochastiques, temporel...

3.4.2.1 Réseau de Petri Hybride

⚡ Définition formel :

Un RdP hybride (*autrement dit en anglais Hybride Petri Nets PN_H*) est une structure $PN_H = (\mathbf{P}, \mathbf{T}, \mathbf{b}, \mathbf{E}, \mathbf{Pré}, \mathbf{Post}, \Sigma, \mathbf{Tempo}, \mathbf{V}, \mathbf{M}_0)$ tel que :

- $\mathbf{P} = \{P_1, P_2, \dots, P_n\}$ est un ensemble de n places, $\mathbf{P} = P^C \cup P^D$ avec :
 - $P^C = \{P_1, P_2, \dots, P_{n_C}\}$ est l'ensemble fini de places continues (*C-places*) ;
 - $P^D = \{P_{n_C+1}, \dots, P_n\}$ est l'ensemble fini de places discrètes (*D-places*) ;
- $\mathbf{T} = \{T_1, T_2, \dots, T_m\}$ est un ensemble de m transitions, $\mathbf{T} = T^C \cup T^D$ avec :
 - ❖ $T^C = \{T_1, T_2, \dots, T_{m_C}\}$ est l'ensemble fini de transitions continues (ou C-transitions) ;
 - ❖ $T^D = \{T_{m_C+1}, \dots, T_m\}$ est l'ensemble fini de transitions continues (ou C-transitions) ;
- $\mathbf{b}: \mathbf{P} \rightarrow \{D, C\}$ est une application qui désigne les nœuds discrets, $\mathbf{h}(x) = D$, et les nœuds continus, $\mathbf{h}(x) = C$;
- \mathbf{E} est un ensemble fini d'événements ;
- $\Sigma: T^D \rightarrow E$ est une fonction qui associe à chaque transition discrète un événement de E ;
- **Pré** et **Post** désignent respectivement les applications d'incidence avant et arrière, ces applications doivent satisfaire la condition suivante :

$$\forall (P_i, T_j) \in P^D \times P^C: \text{Pré}(P_i, T_j) = \text{Post}(P_i, T_j);$$

- ❖ **Tempo** : $T^D \rightarrow \mathbb{Q}^+$ est une application qui associe à chaque D-transition la durée de sa temporisation.
- ❖ **V** : $T^C \rightarrow \mathbb{R}^+$ est une application qui associe à chaque C-transition sa vitesse maximale de franchissement.
- ❖ **M₀** est le marquage initial, des D-places contiennent un marquage entier positif et les C-places contiennent un marquage réel positif ;

La condition sur les applications d'incidence avant et arrière est repérée sur le RdP par des boucles reliant les D-places aux C-transitions, elle signifie qu'une marque discrète ne peut pas être fluidifiée par une transition continue. Le modèle RdP hybride ainsi défini permet donc la modélisation des conditions logiques influant sur le comportement du système, mais il permet aussi la modélisation de transformation de marques continues en marques discrètes et vice-versa (*formation et éclatement de lots*).

Nous numérotons les nœuds du RdP de telle sorte que les nœuds continus aient les indices les plus petits ; cela fait que la matrice d'incidence a la forme suivant :

$$W = \begin{pmatrix} W_C & W_{cD} \\ \mathbf{0} & W_D \end{pmatrix}$$

Les RdP élémentaires constituent une classe particulière de RdP hybrides ou il n'y a pas de transformation de marquage, du discret vers le continu ou du continu vers le discret. Dans ce modèle le RdP T-temporisé contrôle le comportement du RdP continu (C) via des boucles connectant certaines D-places à certaines C-transitions, ce qui signifie que ces dernières ne sont validées et par conséquent ne peuvent être franchies que si les D-places sont marquées.

Le RdP continu (C) à son tour peut influencer le comportement du RdP T-temporisé une D-transition T_j peut avoir comme condition de franchissement le marquage d'une C-place P_i qui atteint un seuil S . Graphiquement ceci est représenté de deux manières soit par une boucle (*un arc de P_i vers T_j et un arc de T_j vers P_i*) dont le poids est S si ce seuil est un seuil supérieur c'est-à-dire si le marquage de P_i ne peut être supérieur à S .

Dans le cas contraire si le marquage de P_i ne doit pas être inférieur à S , un arc inhibiteur est utilisé pour relier T_j à P_i . Et dans les deux cas le franchissement de T_j ne modifie pas le marquage de P_i .

3.4.2.2 Réseau Hybride élémentaire :

✚ Définition formel

Un élémentaire est un couple (PN_H, I) tel que :

- PN_H est un RdP dont les applications Pré et Post satisfont la condition suivante :
- $\forall (P_i, T_j) \in (P^D \times T^C) \cup (P^D \times T^C)$, alors $\mathbf{Pré}(P_i, T_j) = \mathbf{Post}(P_i, T_j)$
- $I: (P_i, T_j) \rightarrow \mathbb{R}$, est une application d'inhibition, si un arc inhibiteur de poids S relie la place P_i à la transition T_j , le franchissement de T_j n'est possible que si le marquage de P_i est inférieur à S .

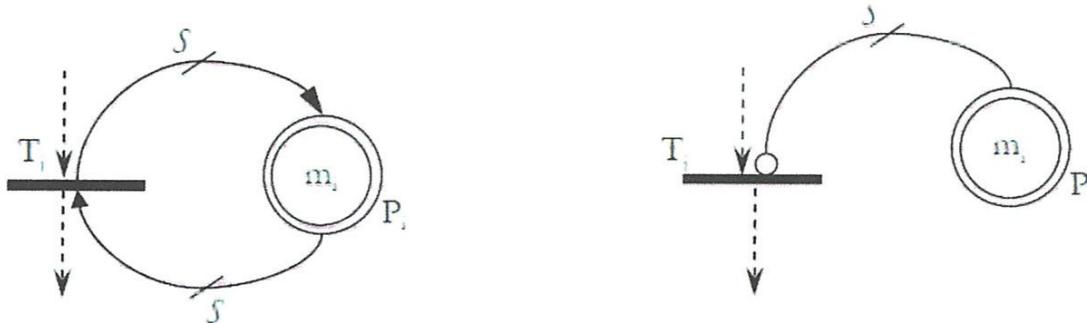


Figure a. validation de T_j si $m_i \geq S$;

Figure b. validation de T_j si $m_i < S$

Figure 3.5 : Exemple de Réseau Hybride élémentaire

✓ Exemple :

Considérons un système constitué de deux réservoirs et de trois vannes, comme illustré dans la figure a. Les vannes sont caractérisées par leur débit V_1, V_2 et V_3 (litres/ seconde), et sont supposées être ouvertes en permanence.

Le système décrit dans les exemples est modélisé par le RdPCC de la figure b. Pour les valeurs numériques suivantes : $V_1 = 3, V_2 = 4$ et $V_3 = 6$. Les **marquages des places P_1 et P_2** représentent les **quantités de liquide** dans les réservoirs. Les vitesses associées aux transitions du RdP continu modélisent les débits des vannes.

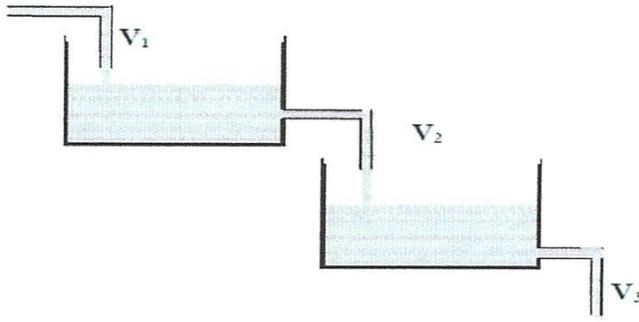


Figure a. système hydraulique

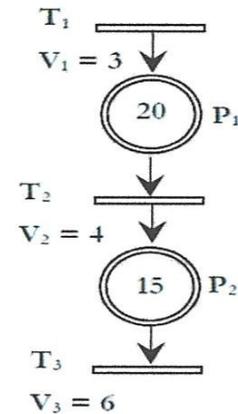


Figure b. RdPCC modèle

Figure 3.5 : Exemple de Réseau Hybride élémentaire

A l'instant initial les trois transitions sont fortement validées, une transition est dite fortement validée si toutes ses places d'entrée sont marquées, et elles sont franchies à leurs vitesses maximales.

3.5 Conclusion

En conclusion, les SDH sont des ensembles d'éléments qui associent une partie discrète et une partie continue. Dernièrement, ces ensembles ont reçu nombreuse attention et plusieurs formalisme ont été proposés afin d'établir un modèle homogène permettant la modélisation l'interaction entre les parties discrètes et continues.

Et par la suite dans ce chapitre, nous avons eu à énoncer les principales approches de modélisations des SDH, les outils de modélisation et la structure générique des SDH.

Et enfin dans notre travail, nous nous sommes intéressés à l'approche consistant à modéliser le système par un automate hybride.

Chapitre 4 :

Modélisation par stateflow

4.1 Introduction

Stateflow est un module développé par la société américaine MathWorks qui permet la simulation de machines d'état. Une machine d'état comporte un nombre fini d'états. Elle modélise le comportement de systèmes qui passent d'un état à un autre en réponse à des événements. On parle alors de systèmes à événements discrets.

Comme son nom l'indique, ce module permet de tracer des diagrammes d'état (*State Chart*) et des diagrammes de flux (*Flow Chart*). Stateflow est intégré à Matlab et Simulink. Les modèles construits pourront par conséquent comporter des blocs des différentes toolboxes de Simulink et/ou appeler des fonctions Matlab et/ou des fonctions Simulink comme nous le verrons. Ainsi le modèle global d'un système complexe pourra comporter des modèles linéaires continus construits avec Simulink sous la forme de schéma-blocs, des machines à état construites avec Stateflow ou encore des modèles acausaux réalisés en utilisant Simscape.

De plus, Stateflow permet aussi de simuler le comportement de systèmes hybrides c'est-à-dire à événements discrets et continus. C'est par exemple le cas d'une balle qui rebondit sur le sol. En effet, son déplacement dans l'air est continu alors qu'à chaque rebond considéré comme un événement, sa trajectoire est modifiée. C'est encore le cas si un robot doit éviter un obstacle présent sur sa trajectoire.

Une connaissance approfondie de Matlab et Simulink n'est pas indispensable pour commencer à travaillé avec Stateflow.

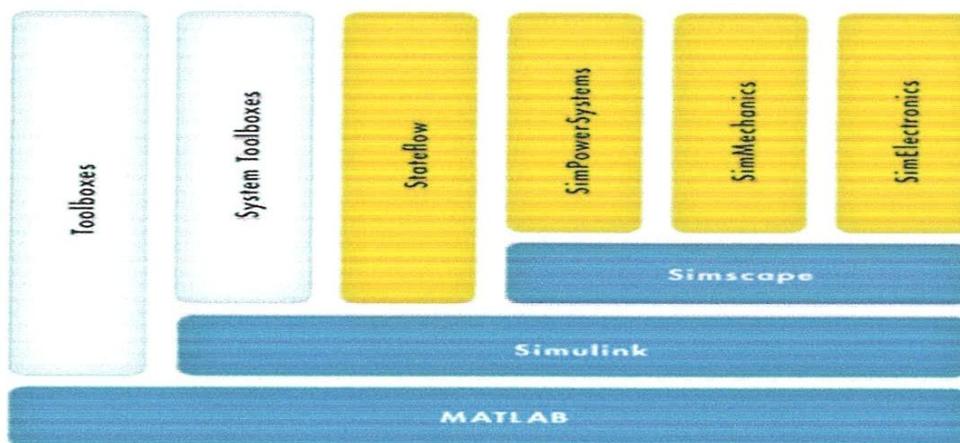


Figure 4.1 : Agencement des différentes plates-formes du logiciel matlab

4.2 Les objets de stateflow

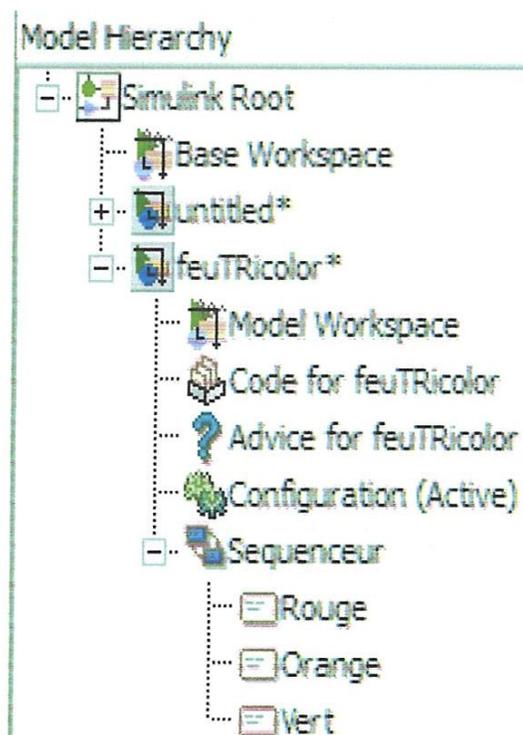
Un graphe Stateflow est composé des différents objets suivants :

- états
- événements
- transition
- les données
- les conditions
- jonction
- jonction de l'historique

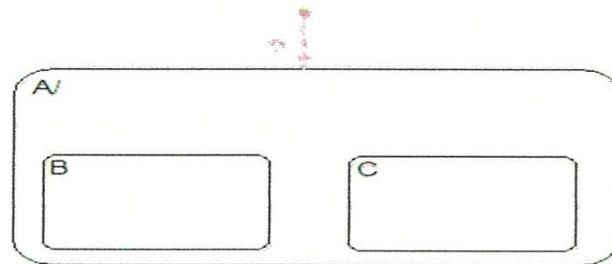
4.2.1 Les états

Dans les états on décrit les actions à réaliser dans le cadre du graphe qui décrit la machine à états finis. Un état peut être actif ou inactif selon des événements qui peuvent intervenir ou des conditions de validation des transitions. En outre un état est appelé super état, ou état parent vis à vis de ceux qu'il contient et ceux-là deviennent des sous-états (*ou enfants*).

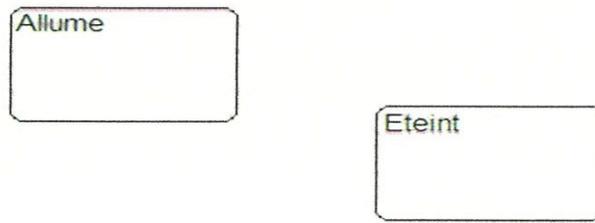
Cette hiérarchie apparaît dans la fenêtre de gauche de l'explorateur (*Tools* → *Explorer*) les états finit, Rouge, Orange, et vert sont des sous-états de l'état Séquenceur, lui-même un enfant de l'état racine Feu tricolore. Notons que feu tricolore est un enfant pour Simulink.



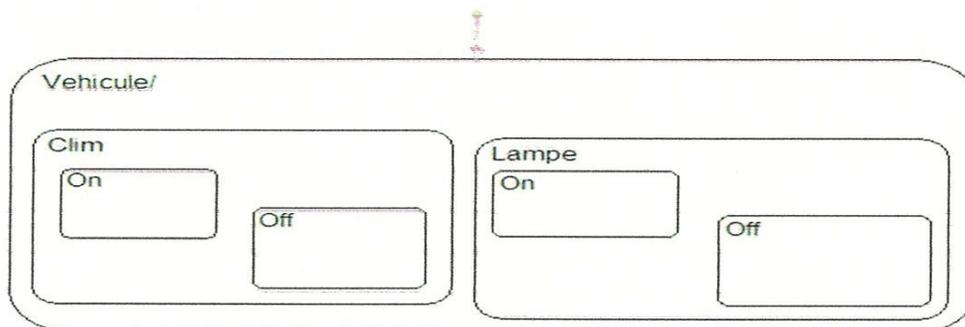
Chaque état possède un parent. Dans le cas suivant, les états **B** et **C** ont l'état **A** comme parent.



Le cas suivant contient des états exclusifs. C'est aussi le cas précédent pour les états **B** et **C** vis-à-vis de l'état parent **A**. Ils ne peuvent pas être actifs en même temps. On dit qu'ils sont en décomposition **OU**. L'exclusivité est réalisée par les transitions qui contiennent des conditions qui valident le passage d'un état vers un autre.



Un graphe Stateflow peut posséder plusieurs états qui peuvent être actifs simultanément. Ces états sont dits parallèles, soit en décomposition **ET**. Les états parallèles sont encadrés par des bords en pointillés. Chaque état possède un nom (*label*) en haut à gauche du rectangle délimitant cet état.

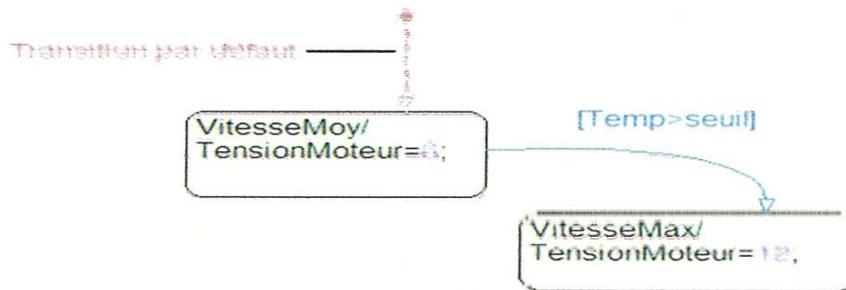


4.2.2 Les transitions

Une transition est un objet graphique qui lie un état à un autre. Un label décrit les circonstances ou les conditions du passage entre ces états.

4.2.2.1 Les transitions par défaut

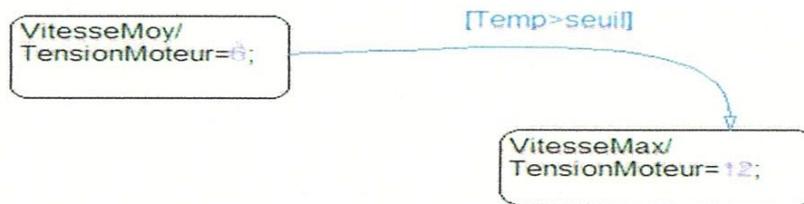
La transition par défaut (*sans aucune condition, ni label*) détermine l'état qui doit être actif lorsqu'il y a ambiguïté entre plusieurs états en décomposition OU et qui ont le même niveau de hiérarchie. Chaque état qui contient des sous-états doit avoir sur l'un d'eux une transition par défaut qui détermine cet état actif dès l'entrée de ce super état (*ou état parent*).



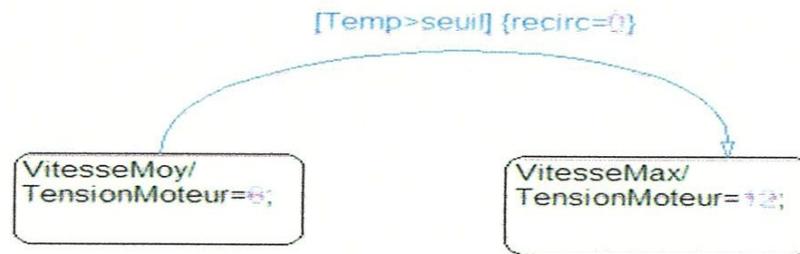
4.2.2.2 Les labels des transitions

Une condition est une expression booléenne qui valide la transition du passage d'un état vers un autre. La condition est mise entre crochets. Les conditions peuvent faire intervenir des variables locales ou des entrées SIMULINK.

L'exemple suivant montre l'augmentation de la vitesse du ventilateur lorsque la température ambiante dépasse un certain seuil.

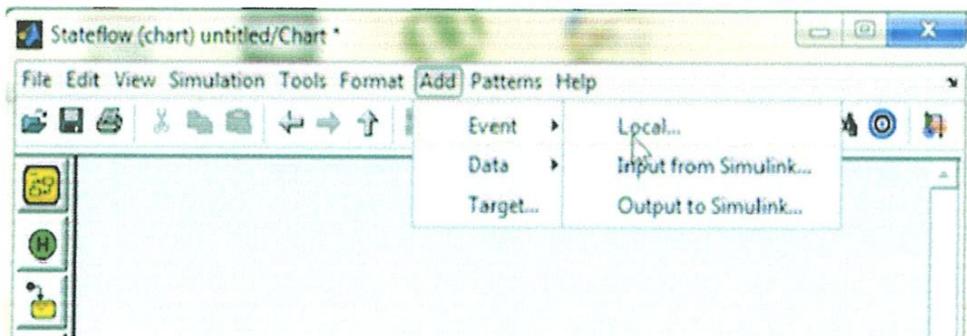


On peut mettre un nom de variable logique locale auparavant mise à 1 lorsque la condition est réalisée. Dans une transition, on peut réaliser des actions comme celles qu'on fait à l'intérieur d'un état. Les actions réalisées dans une transition sont mises entre accolades. Dans le cas suivant, la transition se fait lorsque la température dépasse le seuil et dans ce cas on arrête le moteur de recirculation de l'air en mettant la variable recirc à 0.

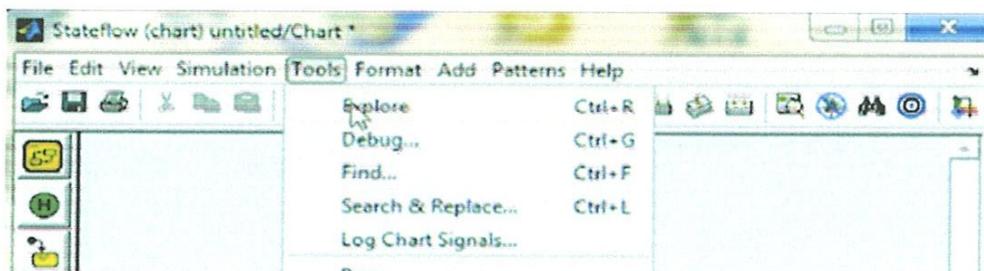


4.2.3 Les évènements

Les événements, objets non graphiques qui agissent sur l'exécution du graphe Stateflow. Les événements peuvent être visualisés, pour ce faire il nous faut cliquer sur le menu **Add** de l'éditeur Stateflow (*Tous les événements doivent être définis par le menu Add*).



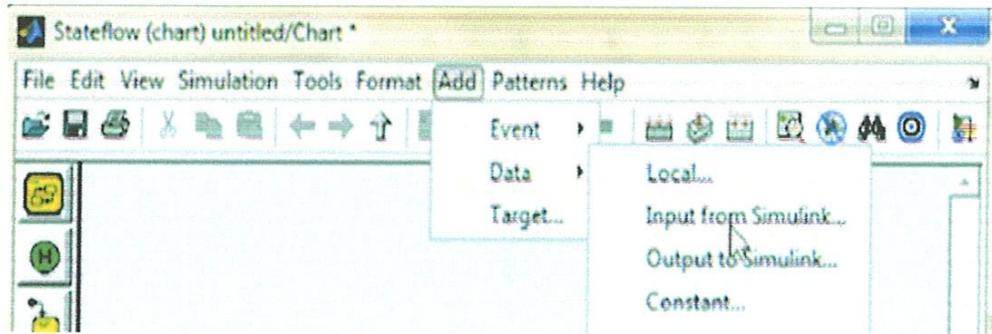
Par la suite, les événements seront alors visibles via le menu **Tools**→**Explorer**.



L'occurrence d'un événement peut servir à valider une transition de passage d'un état à un autre ou une action à exécuter. Enfin, les événements représentant l'environnement (*le monde externe*) doivent être obligatoirement un front montant, descendant ou les deux, ceci afin de simuler le comportement par interruption du système d'ordonnancement.

4.2.4 Les objets data (*Données*)

Les objets Data peuvent être des variables locales, entrées de ou sorties vers SIMULINK, une constante, etc.



Les objets data stockent des valeurs numériques pour référence dans le diagramme Stateflow® mais sont des objets non graphiques.

4.3 Différents exemples d'applications

✓ Exemple 1 : Représentation d'un signal de sortie

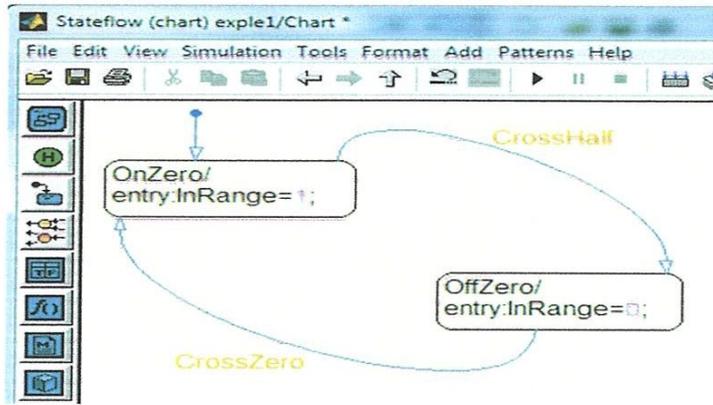
Construction d'un modèle simulink de représentation de signal de signal de sortie basé sur un évènement agissons sous sumulink en utilisant un bloc stateflow. Sachant que le système doit produire une sortie de 1.0 lorsque l'entrée est dans l'intervalle $0 \leq u \leq 0.5$ et la valeur u est croissante et produit une sortie égale à 0.004.

Pour construire le modèle sumilink, il nous faut suivre les étapes suivantes :

- Ouvrir matlab puis simulink ensuite créer un nouveau modèle.
- Ouvrir simulink library browser.

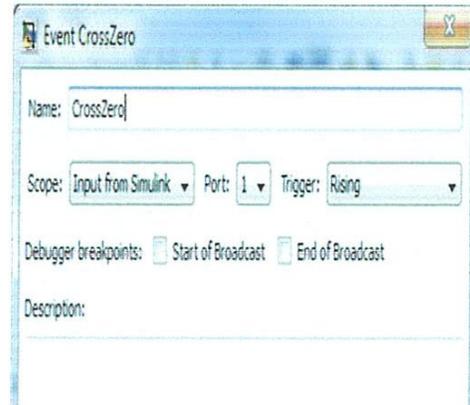
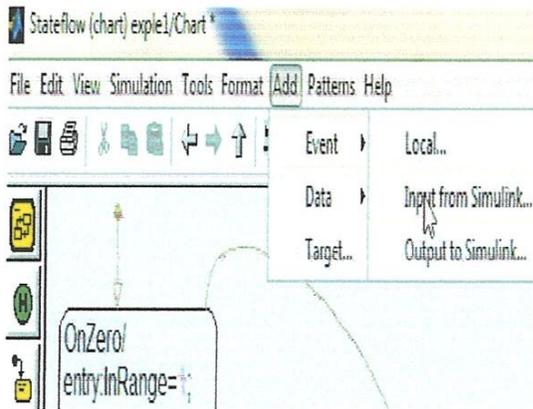
Puis dessiner dans ce modèle en utilisant les blocs :

- Simulink → Sources → Sine Wave
- Simulink → Signal Routing → Mux
- Simulink → Discontinuities → Hit Crossing pour la valeur offset 0 de direction to rising et le nommée Cross zero rising.
- Simulink → Discontinuities → Hit Crossing pour la valeur offset 0.5 de direction to either et le nommée Cross ½.
- Sumilink → Stateflow → Chart
- Simulink → Sinks → Scoope
- Ouvrir le bloc chart afin de le modéliser :
 - Réaliser la saisie suivant.



Configurer Stateflow afin de définir les deux événements :

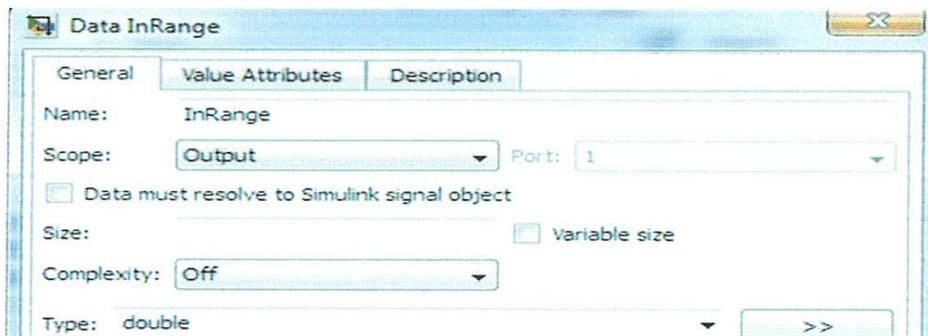
- **Menu stateflow** → **Add Event** → **Input from simulink** et configurer comme indiqué ci contre (*Name et Trigger*).



- Faire la même opération pour CrossHalf : l'index devient 2 et Trigger à Either.

Définir la sortie du bloc chart :

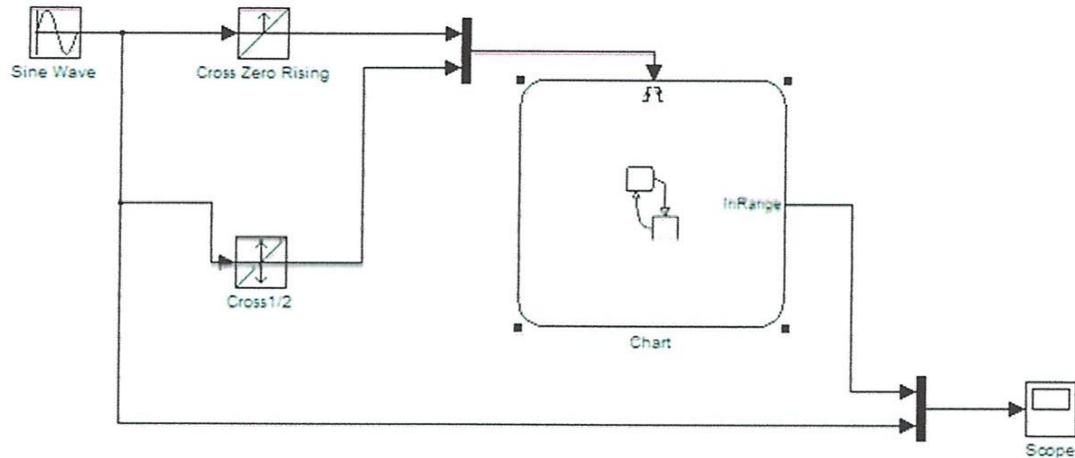
- **Menu stateflow** → **Add Data** → **Output from simulink** puis configurer comme indiqué ci-dessus (*Name*).



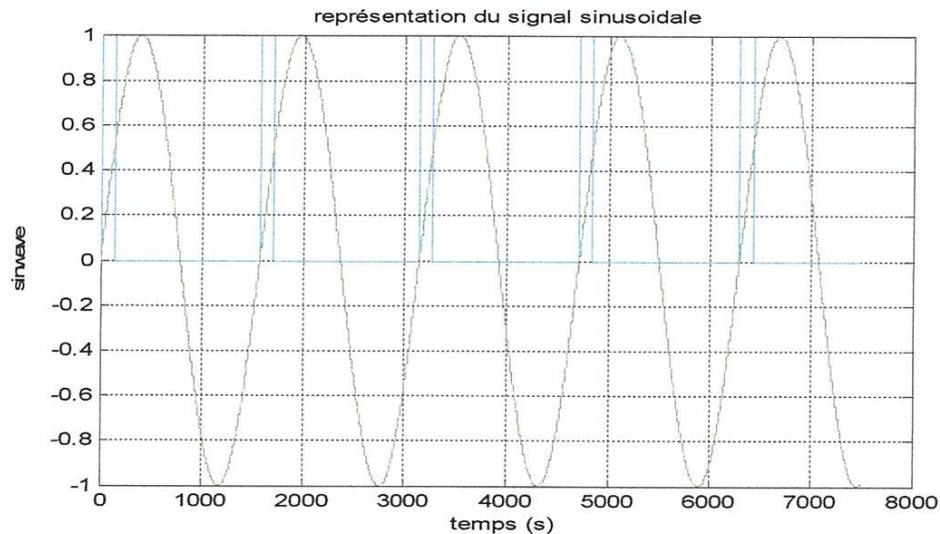
➤ Définir les paramètres de la simulation simulink :

- **Modèle Simulink** → **Simulation** → **Configuration Parameters** puis configurer l'instruction type à fixed step and Discrete, donner à Fixed step size la valeur 0.004 et le stop time la valeur 30.

➤ Réaliser la simulation du modèle suivant :



➤ Résultat de la simulation :



Nous constatons que :

- La sortie passe à 1 à chaque front montant au passage par zéro du signal sinusoidal.
- La sortie passe à 0 à chaque front montant au passage par $\frac{1}{2}$ du signal sinusoidal.

✓ Exemple 2 : Représentation d'un Clignotant

Le système consiste au passage d'un état vers un autre à la fréquence d'un signal d'horloge. L'ordre de clignotement doit être validé avant tout fonctionnement, indépendamment de la présence de ce signal.

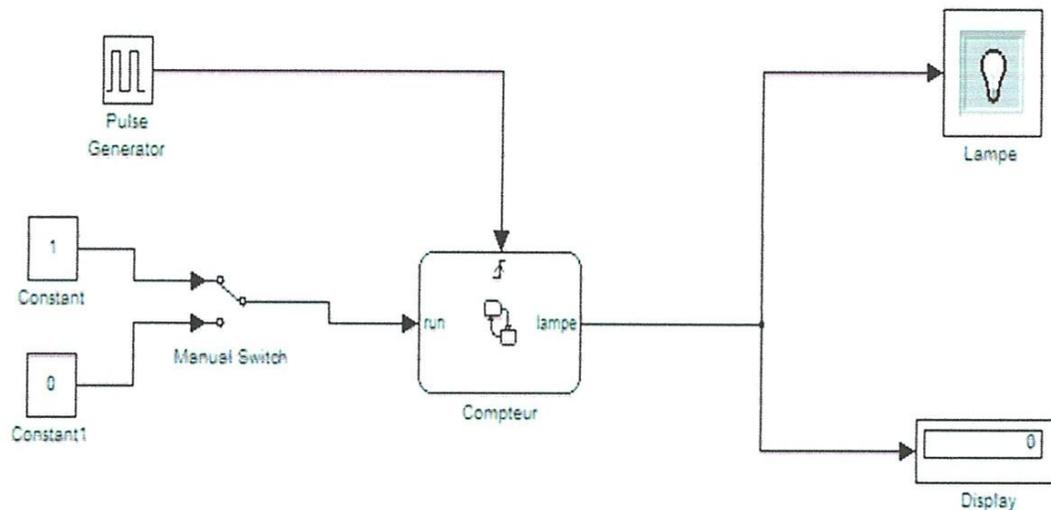
Le modèle est constitué de :

- Deux blocs constants
- Un Pulse Generator
- Un manuel switch
- Un bloc chart
- Un display
- Light bulb

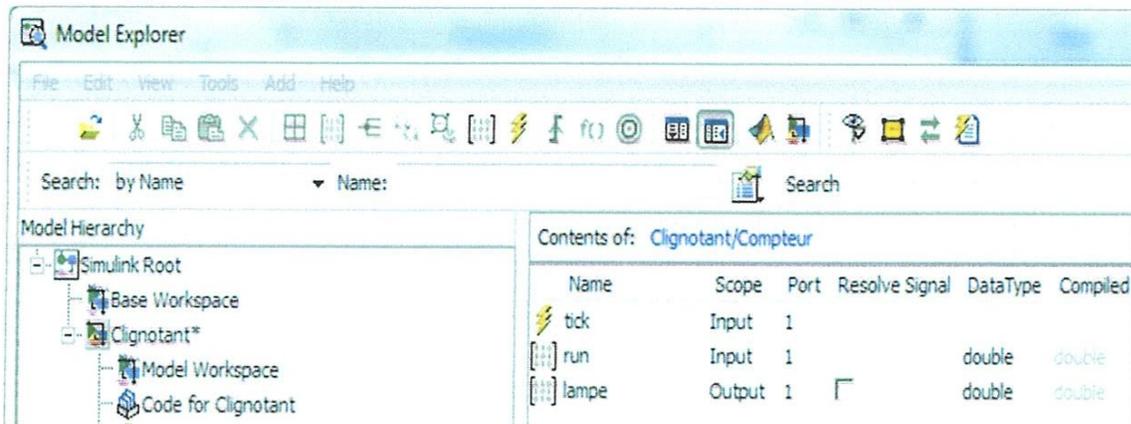
Dans l'arborescence des bibliothèques de Simulink les LEDs sont disponibles à l'emplacement suivant :

➤ **Gauges Blockset** → **Global Majic ActiveX Library** → **On Off Gauges** → **Light Bulb**

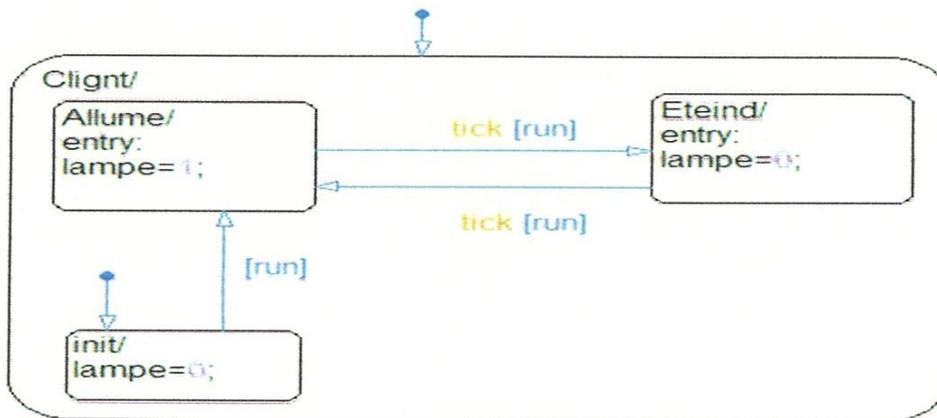
Le schéma bloc du modèle est représenté comme suit :



- Le modèle stateflow possède une entrée **run** qui valide le fonctionnement de clignotement et une entrée **tick** de déclenchement (trigger) sous forme d'un signal carré. Le signal de sortie **lampe** attaque un afficheur et une ampoule (Light Bulb).



➤ Modélisation du bloc chart (*compteur*) :



L'état **Clignot** peut être défini comme un sous état du bloc Stateflow puisqu'il est le seul.

De plus l'état **init** est un sous état de l'état **Clignot**, Ils ont tous deux comme tout état interne ou sous état une transition par défaut (*sans condition*). Dès l'entrée dans l'état **Clignot**, on aboutit à l'état **init** où on simule l'extinction de la lampe.

Si l'entrée **run** est à au niveau **1** on entre dans l'état Allume. Dès l'entrée dans cet état (*entry :*), on met le signal logique lampe à **1**.

L'extinction de la lampe se fait au front montant du signal de déclenchement si le signal **run** est à **1**, d'où la condition `tick [run]`. Cette même condition sur les signaux **run** et **tick** permet le retour à l'état **init** où la lampe s'éteint. Tant que cette condition est valide, on passe sans discontinuer de Eteint à Allume et inversement.

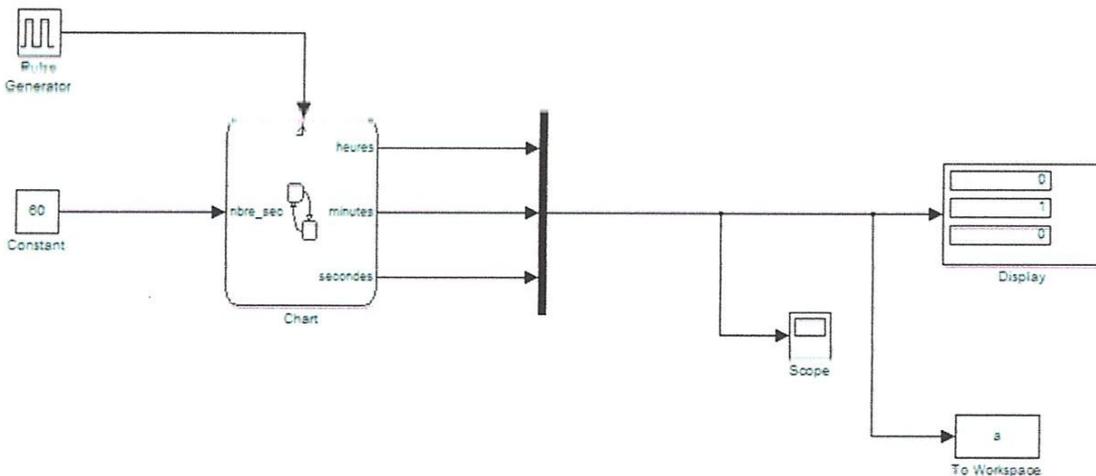
On observe dans SIMULINK le clignotement (*jaune/blanc*) de la lampe de l'outil Gauges Blocksets ou l'affichage 0/1 sur le bloc display.

✓ **Exemple 3 : Horloge**

Le simulink suivant permet à partir d'un nombre de secondes, de donner le nombre d'heures, de minutes et de secondes.

Le modèle simulink est composé par :

- Un Pulse Generator
- Un bloc constant
- Un bloc chart
- Un Mux
- Un display



Le bloc Stateflow est déclenché par le signal tick sur le front montant (Rising) d'un signal carré.

Il reçoit le chiffre en secondes sur son entrée nbre_sec, entrée de simulink sous forme d'un entier non signé, codé sur 8 bits, unit8.

Ses 3 Sorties (heures, minutes, secondes) sont affichés sur le bloc display, après passage dans un multiplexeur.

Name	Scope	Port	Trigger	Resolve Signal	DataType	Compiled Type
⚡ tick	Input	1	Rising			
run	Local			<input type="checkbox"/>	double	double
nbre_sec	Input	1			double	double
stop	Local			<input type="checkbox"/>	double	double
heures	Output	1		<input type="checkbox"/>	double	double
minutes	Output	2		<input type="checkbox"/>	double	double
secondes	Output	3		<input type="checkbox"/>	double	double

Les signaux logiques **run** et **stop** servent aux transitions pour valider le calcul et son arrêt.

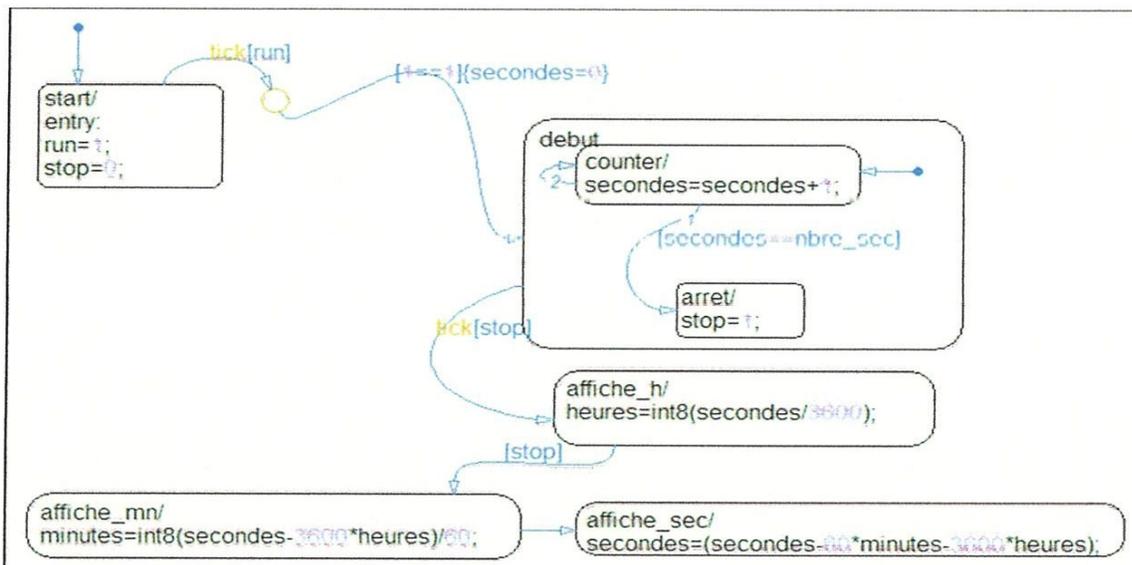
La programmation n'est pas optimale mais uniquement à but pédagogique afin d'étudier d'autres fonctionnalités de stateflow.

Le test $[1==1]$ est toujours valide, condition toujours réalisée.

Lorsqu'on doit se déplacer vers plusieurs états selon des conditions différentes, ces dernières valideront autant de transitions qui partent d'un même état.

On ne peut pas avoir plusieurs transitions par défaut partant d'un même état.

Une manière d'éviter cette erreur est de mettre cette condition qui rend cette transition prioritaire à une autre par défaut.



Dès l'entrée dans l'état **start**, on initialise les variables locales **run** et **stop** respectivement à 1 et 0.

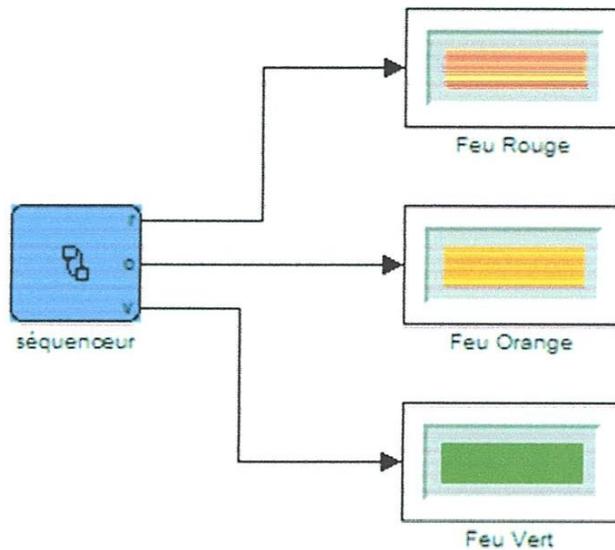
L'affiche des sorties ne se fait que lorsque la variable **stop** est à 1 et au front montant du signal de déclenchement.

✓ Exemple 3 : Séquenceur

Dans cet exemple il s'agira de réaliser la conception d'un séquenceur permettant de commander automatiquement des feux tricolores.

Le schéma bloc dans Simulink utilise :

- Un bloc chart
- Trois LEDs



➤ Modéliser les blocs LEDs afin de pouvoir affecter les couleurs allumé et éteinte des LEDs.

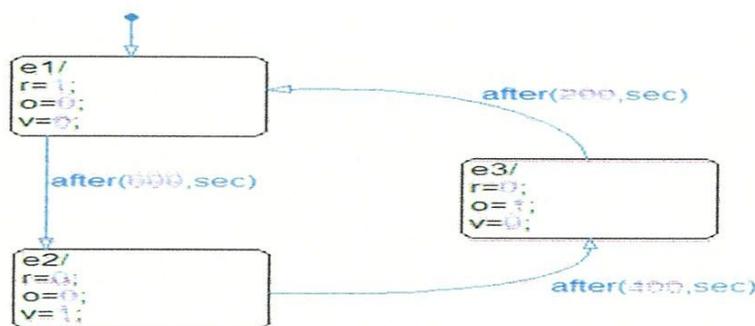
- **Clic droit sur LEDs → Control Display Properties → Styles**

➤ Modéliser le bloc chart.

Le modèle stateflow possède 3 sorties qui valident l'allumage ou le faite d'être éteint d'une LEDs.

Name	Scope	Port	Resolve Signal	DataType	Compiled Type	Size	Compiled Size	UpdateM
r	Output 1	1	<input type="checkbox"/>	double	unknown			Discrete
o	Output 2	2	<input type="checkbox"/>	double	unknown			Discrete
v	Output 3	3	<input type="checkbox"/>	double	unknown			Discrete

Le bloc chart contient 3 états (nommés ici e1, e2 et e3) allumant 1 feu et éteignant les 2 autres. Pour passer automatiquement d'un état à l'autre au bout d'un certain temps on utilise la transition after.



4.4 Conclusion

Ce chapitre a passé en revue un ensemble d'outils de modélisations en présentant les aspects forts de chacun vis-à-vis des systèmes dynamiques hybrides. Il est à rappeler que les systèmes dynamiques hybrides ont été l'objet de plusieurs études et plusieurs formalismes ont été proposés afin d'établir un modèle permettant la modélisation de l'interaction entre les parties discrètes et continues.

Nous avons présenté dans ce chapitre des conditions concernant la modélisation sous Matlab, Simulink et Stateflow. En effet, Simulink/Stateflow et en particulier Stateflow s'avère être un outil essentiel sinon vital pour la modélisation et la simulation des systèmes réels.

Les méthodes de conversion simulink/stateflow qui ont été énumérées dans ce chapitre permettent de représenter, sans difficulté et sans complexité des systèmes dynamiques.

Nous allons développer dans le chapitre suivant une approche hybride développée dans la plate-forme Matlab/Simulink/Stateflow pour le diagnostic et la surveillance du comportement réel du système étudié.

Chapitre 5 :

Cas d'étude

Systeme de freinage anti-lock braking
system(ABS)

5.1 Introduction

Le système de freins antiblocage (ABS) est un dispositif électronique servant à mesurer et à contrôler la vitesse de chaque roue pendant la période de freinage. Le dispositif fonctionne avec les systèmes de freinage pneumatique standard.

Le système ABS mesure en tout temps la vitesse de chaque roue et contrôle le niveau de freinage dans les conditions de blocage de roues. Le système permet d'améliorer la stabilité et la directibilité du véhicule en réduisant le niveau de blocage des roues pendant le freinage.

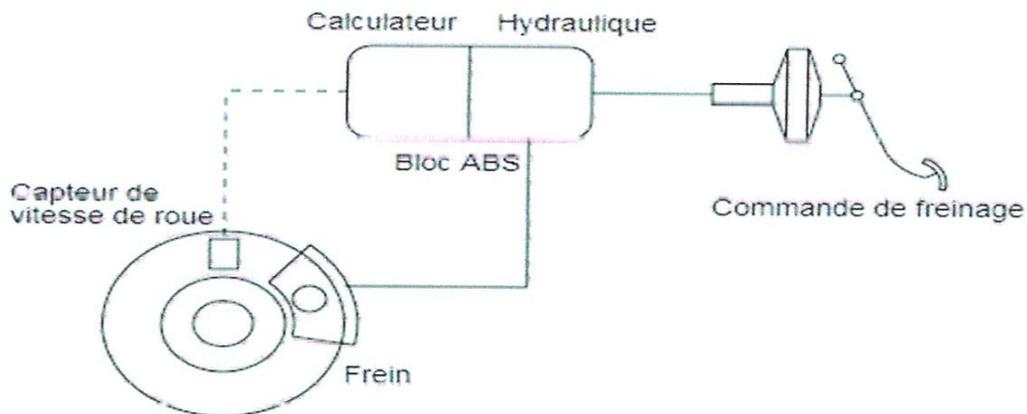


Figure 5.1 : Système de freinage antiblocage (ABS) : Schéma synoptique d'un système électro-hydraulique d'ABS additionnel automobile.

Dans ce chapitre nous allons par la suite montrer comment modéliser un modèle simple pour un système de freinage antiblocage (ABS). Et ensuite simuler le comportement dynamique d'un véhicule dans des conditions de freinage des disques. Le modèle représente une seule roue qui peut reproduire un certain nombre de fois pour créer un modèle pour véhicule multi-roue.

5.2 Principe de fonctionnement

Le système anti-blocage des roues, plus connu sous l'abréviation ABS (de l'allemand Antilockiersystem), est un système d'assistance au freinage utilisé sur les véhicules roulants, limitant le blocage des roues pendant les périodes de freinage intense. Fonction secondaire dans le système de contrôle de traction, elle est utilisée dans les avions (lors de l'atterrissage) et dans les véhicules automobiles ou motocyclettes, où elle fait de plus en plus partie de l'équipement standard.

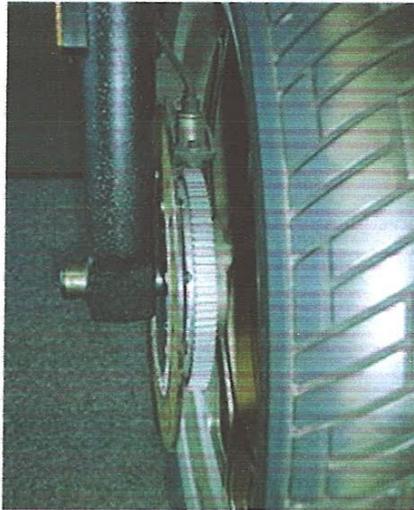


Figure 5.2 : Le mécanisme d'ABS sur une moto BMW K75. Notez la roue dentée et le capteur à côté du disque.

L'ABS a été conçu à l'origine pour aider le conducteur à conserver la maîtrise de son véhicule, dans le cas d'un freinage dans des conditions d'adhérence précaire (pluie, neige, verglas, gravier...). Celui-ci permet de conserver la directivité du véhicule afin, d'effectuer une manœuvre d'évitement éventuelle, tout en optimisant la distance de freinage, celle-ci augmentant considérablement lorsque les roues se bloquent et que les pneumatiques glissent sur la chaussée.

Le système ABS est constitué d'un capteur de vitesse pour chaque roue, d'un calculateur électronique et d'un système de régulation hydraulique de la pression de freinage. Lors d'un freinage d'urgence, si le système détecte le blocage d'une roue, cela signifie que la pression de freinage est trop forte compte tenu de l'adhérence disponible. Le système va alors pomper le liquide de freins du frein vers le maître-cylindre de façon à baisser la pression jusqu'à ce que la roue soit débloquée. Dès lors que la roue retrouve de l'adhérence, la pression de freinage est à nouveau augmentée de façon à optimiser la performance du freinage. Par conséquent, le système régule la pression de freinage autour du point de blocage de la roue par une succession rapide de blocages/déblocages. Le système fonctionne par impulsions, chose que l'on ressent en général à la pédale de frein qui se met à vibrer au rythme des variations de pression du circuit de freinage.

5.3 Approche hybride et Algorithme de simulation

L'objectif de cette partie est de développer une approche hybride pour la surveillance et le contrôle du cas d'application(ABS). Comme le système ABS est un système dynamique hybride, on peut résumer l'algorithme de simulation par la structure générale suivante :

L'évolution dynamique d'un système hybride peut être décrite par une succession de modes. Chaque mode I ($I \in M$; $M = \{1, 2, \dots, m\}$, où m est le nombre de modes) correspond à une configuration physique possible.

- Pas 1 : initialisation des variables;
- Pas 2 : simulation de la partie discrète;
- Pas 3 : **si** aucune transition ne puisse être franchie **alors** allez au pas 4
Sinon retour au pas 2;
- Pas 4 : intégration du mode I ;
- Pas 5 : mise en place et initialisation du modèle continue global
- ❖ Pas 6 : simulation de la partie continue
- ❖ Pas 7 : **Si** l'occurrence d'un événement est atteinte **alors** fin de cycle, alors aller au pas 8
Sinon retour au pas 6 ;
- ❖ Pas 8 : Obtention du nouveau mode $I=I+1$;
- ❖ Pas 9 : **si** le temps de simulation total est atteint **alors** :
 - concaténation des résultats d'intégration de mode,
 - schématisation des résultats,**Sinon** retour au pas 2;
- ❖ Pas 10 : fin de simulation.

5.4 Application : Intérêt de l'ABS

Le système de freins antiblocage (ABS) est un dispositif électronique servant à mesurer et à contrôler la vitesse de chaque roue pendant la période de freinage. Le dispositif fonctionne avec les systèmes de freinage pneumatique standard.

Le système ABS mesure en tout temps la vitesse de chaque roue et contrôle le niveau de freinage dans les conditions de blocage de roues. Le système permet d'améliorer la stabilité et la directibilité du véhicule en réduisant le niveau de blocage des roues pendant le freinage comme représenté dans la figure 5.3.

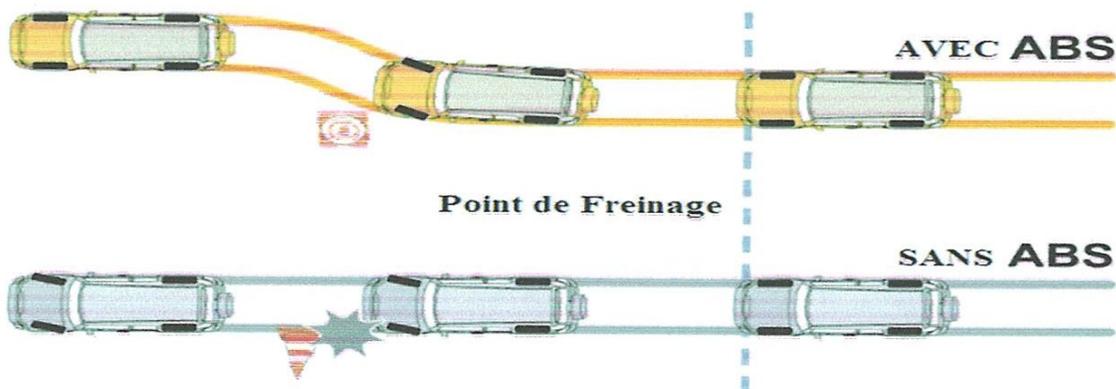


Figure 5.3 : Freinage avec et sans ABS

5.4.1 Modélisation Dynamique

La modélisation dynamique est une étape de la modélisation consistant à déterminer à partir du profil de mission les variables x_i .

En effet, la dynamique du système ABS est décrite par des équations différentielles qui permettent de réaliser un modèle physique de ce système ABS.

❖ Détermination des équations différentielles

Afin de parvenir à avoir les équations différentielles, illustrant tout d'abord une roue tournant avec une vitesse angulaire initiale qui correspond à la vitesse du véhicule avant que les freins ne soient appliqués avec tous les efforts qui s'appliquent sur cette dernière.

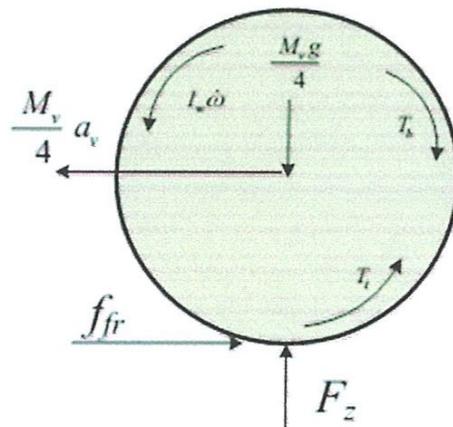


Figure 5.4 : Efforts agissant sur une roue freinée

Où :

- w représente la vitesse angulaire
- T_t est le moment de frottement
- T_b est le moment de freinage
- I est le moment d'inertie
- a_v représente l'accélération du véhicule
- F_{fr} est la force de frottement
- M_v est la masse du véhicule
- μ est le coefficient de frottement
- f_s est l'effort normal agissant sur la roue
- V_v représente la vitesse du véhicule
- R représente le rayon de la roue

A partir des efforts appliqués sur une des quatre roues du véhicule comme le montre la figure nous pouvons donc écrire les équations suivantes :

La somme des forces suivant l'axe y nous donne l'expression :

$$F_z = \frac{M_v}{4} * g \quad 5.1$$

La somme des forces suivant l'axe x nous permet d'avoir l'expression de la force de frottement :

$$-F_{fr} = \frac{M_v}{4} * a_v \quad 5.2$$

L'expression de la vitesse s'écrira alors comme suit :

$$V_v = \int a_v . dt + V_{v0} = \int -\frac{4F_{fr}}{M_v} . dt + V_{v0} \quad 5.3$$

La force de frottement est égale à :

$$F_{fr} = \mu * f_z \quad 5.4$$

En remplaçant 5.1 dans 5.4, nous obtenons l'expression suivante :

$$F_{fr} = \mu * \frac{M_v}{4} * g \quad 5.5$$

La somme des moments agissant sur la roue s'écrivent de la manière suivante :

$$T_t - T_b = I \dot{w} \quad 5.6$$

Le moment généré par la force de frottement s'écrit comme montré dans l'équation :

$$T_t = F_{fr} * R = \mu * \frac{M_v}{4} * g * R \quad 5.7$$

En remplaçant 5.8 dans 5.7, nous obtenons l'équation :

$$\mu * \frac{M_v}{4} * g * R - T_b = I \dot{w} \quad 5.8$$

L'expression de la vitesse s'écrit de la manière suivante :

$$w = \int \dot{w} . dt + w_0 = \int \frac{1}{I} \left(\mu * \frac{M_v}{4} * g * R - T_b \right) . dt + w_0 \quad 5.9$$

Le taux de glissement est défini comme suit :

$$S = \frac{V_v - R w}{V_v} \quad 5.10$$

5.4.2 Modélisation physique

Après la détermination des équations différentielles propres au système ABS, nous allons construire le modèle physique de ce système en utilisant Simulink.

Dans ce modèle, nous avons utilisé un contrôleur de freinage anti-blocage idéal, qui utilise le contrôle «bang-bang» sur la base de l'erreur entre glissement réel et le glissement désiré. Nous avons fixé le glissement souhaité à la valeur de glissement à laquelle la courbe mu-dérapant atteint une valeur de crête, ce qui est la valeur optimale pour la distance minimale de freinage.

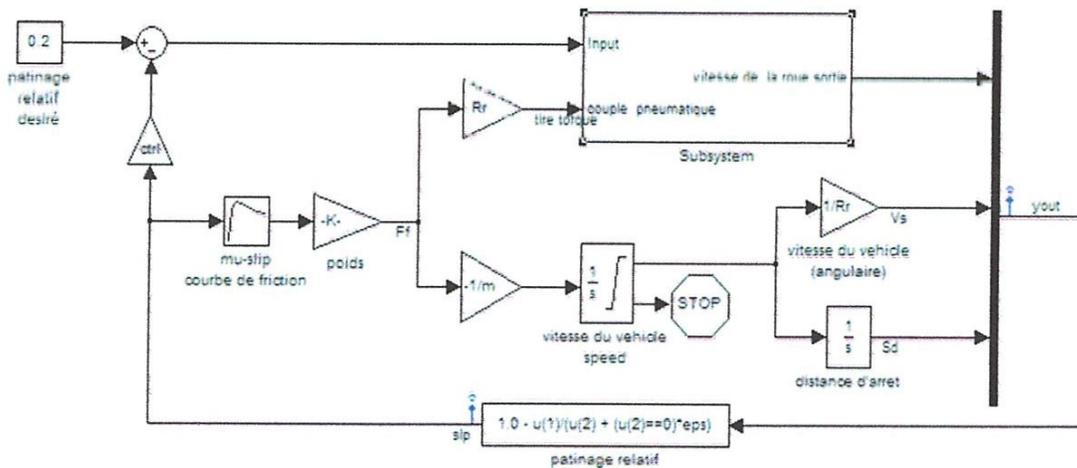


Figure 5.5 : Modèle de freinage antiblocage

Le modèle de freinage antiblocage actuel est constitué par sous-système 'Wheel speed'. Ainsi pour l'ouvrir, il nous faut réaliser un double clic sur celui. Et compte tenu du patinage des roues, le patinage des roues désiré et le couple de pneus ce sous système nos permet par ailleurs de calculer la vitesse angulaire de la roue.

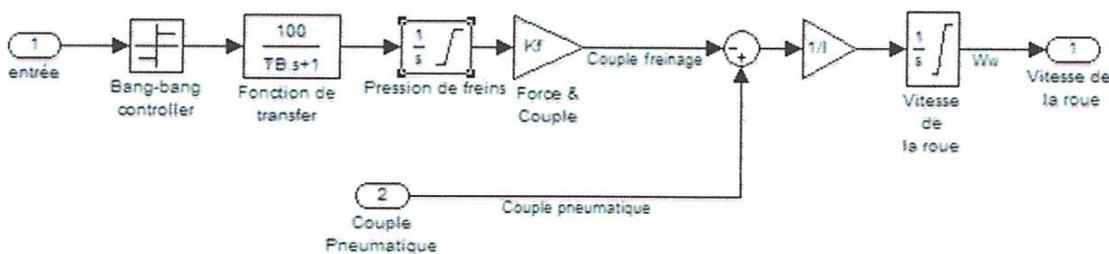


Figure 5.6: Modèle de freinage antiblocage

Le modèle soustrait le glissement réel du glissement souhaité et alimente ce signal dans un contrôle bang-bang (1 ou -1, selon le signe de l'erreur, voir la figure 5.6), ainsi pour pouvoir contrôler le taux de variation de la pression de freinage. Ce taux on / off traverse un retard du premier ordre qui représente le retard associé à des conduites hydrauliques du système de freinage. Le modèle intègre alors le taux

filtré pour donner la pression de freinage réelle. Le signal résultant, multipliée par la surface du piston et de rayon par rapport à la roue (K_f), est le couple de freinage appliqué à la roue.

Ainsi, pour donner le couple d'accélération de la surface de la route sur la roue, le modèle multiplie la force de frottement sur la roue par le rayon de la roue (R_r). Le couple de freinage est soustrait pour donner le couple net sur le volant. Divisant le couple net par la roue d'inertie de rotation, pour obtenir rendements de l'accélération de la roue qui est ensuite intégré à fournir vitesse de roue. Et afin de maintenir la vitesse de la roue et la vitesse du véhicule positif, intégrateurs limités sont utilisées dans ce modèle.

5.4.2.1 Simulation en mode ABS

Pour réaliser la simulation, il nous faut juste appuyez sur le bouton "Play" sur la barre d'outils du modèle. Nous obtenons par la suite les graphes suivants :

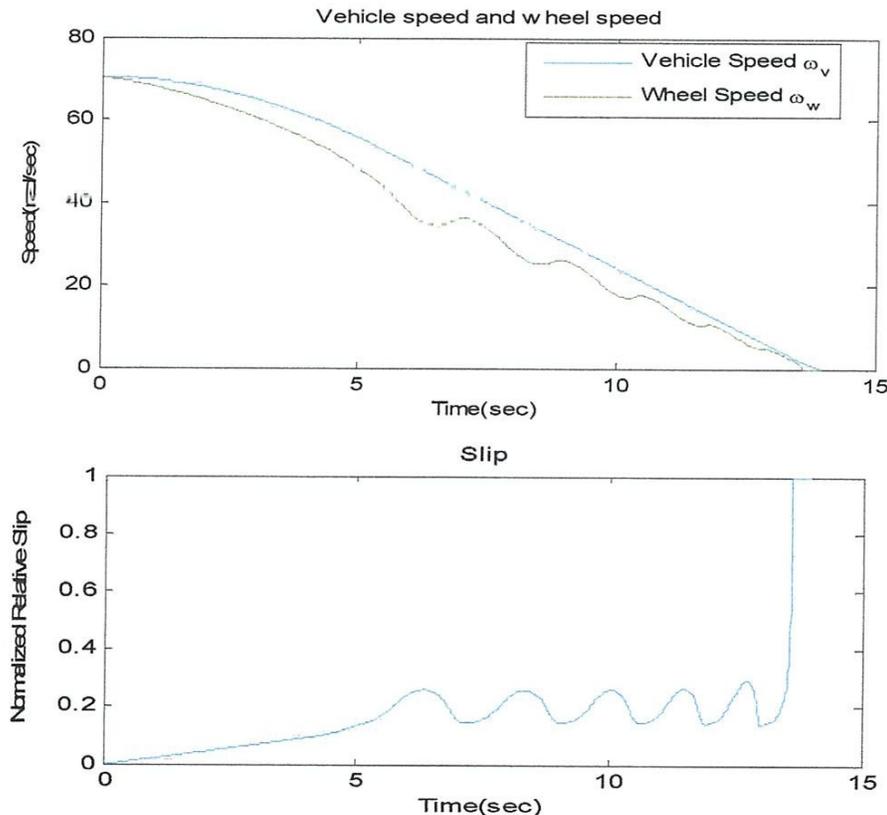


Figure 5.7 : Résultat de la simulation en mode ABS

Nous constatons que le premier graphique montre la vitesse angulaire de la roue et le véhicule correspondant vitesse angulaire. De plus, ce graphique montre que la vitesse de la roue reste en dessous de la vitesse du véhicule sans bloquer, la vitesse du véhicule passe à zéro en moins de 15 secondes, d'une part.

Et d'autre part, le second graphe modélise la durée de freinage sous mode ABS.

5.4.2.2 Simulation sans mode ABS

Afin d'obtenir des résultats plus significatifs, et de considérer le comportement du véhicule sans ABS. En effet, sur la ligne de commande MATLAB définissons la variable de modèle **ctrl = 0**. Ceci déconnecte les évaluations de glissement du contrôleur (voir Figure 5.5) ce qui entraîne un freinage maximal. Et nous obtenons les résultats comme suit :

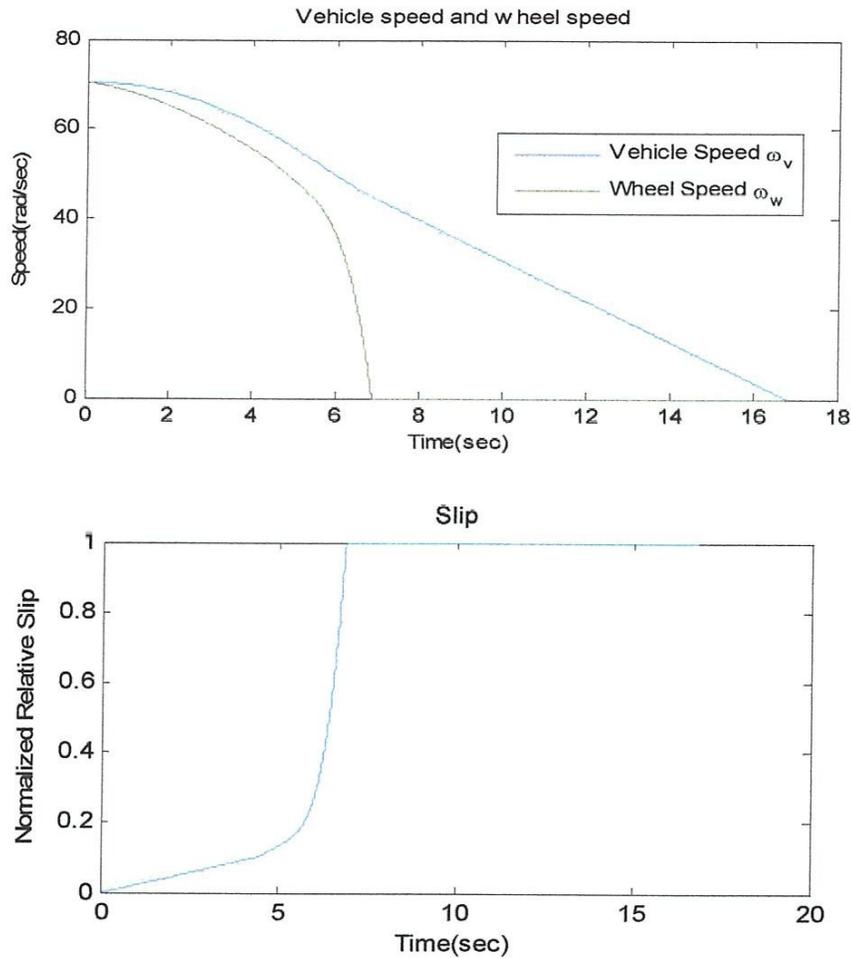


Figure 5.8 : Résultat de la simulation sans mode ABS

Dans la parcelle supérieure de la figure 5.8, nous observons que le verrouillage de la roue se réalise à environ sept secondes. Et le freinage à partir de là est appliqué dans une partie moins que optimal de la courbe de glissement.

Autrement dit, lorsque le glissement = 1 comme on le voit dans le graphique inférieur de la figure 5.8, le pneu dérape tant sur le trottoir que la force de frottement a fléchi.

5.4.2.3 Simulation de l'ABS avec régulateur

On ajoute un régulateur PID au système ABS pour améliorer son fonctionnement comme représenté dans la figure suivante :

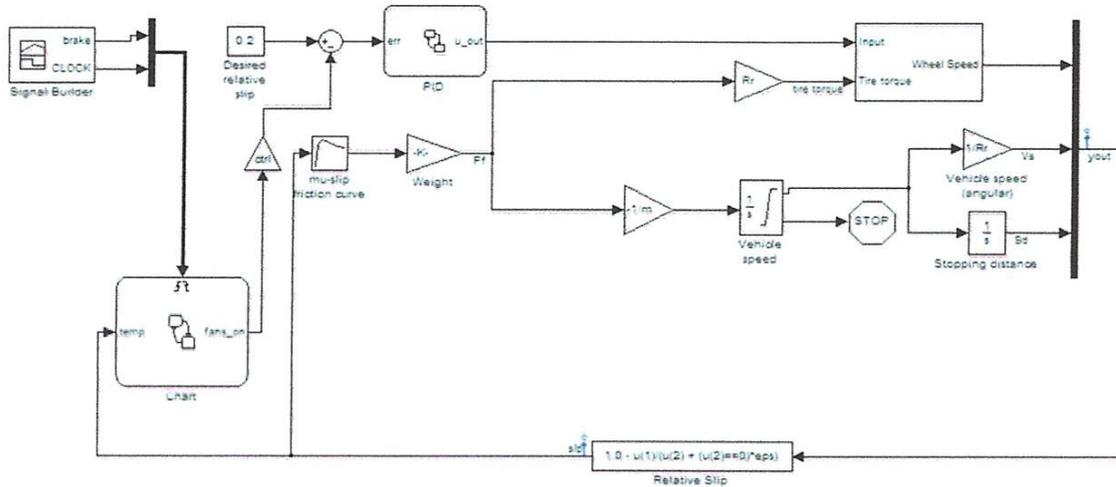


Figure 5.9 : Modèle de freinage antiblocage avec PID

Le régulateur PID, i.e. Proportionnel-Intégral-Dérivée, est la combinaison des trois actions de base P, I et D. Grâce au terme I, il permet l'annulation d'une erreur statique tout en autorisant grâce à l'action D des performances de rapidité supérieures à celles d'un régulateur PI.

- Loi de commande du régulateur PID :

$$u(t) = k_p * (e(t) + \frac{1}{T_i} * \int_{-\infty}^t e(\tau) * d\tau + T_d * \frac{d_e}{dt})$$

- Fonction de transfert du régulateur PID :

$$G_c(s) = \frac{U(s)}{E(s)} = K_p * \frac{1 + s * T_i + s^2 * T_i * T_d}{s * T_i}$$

- Schéma fonctionnel du régulateur PID :

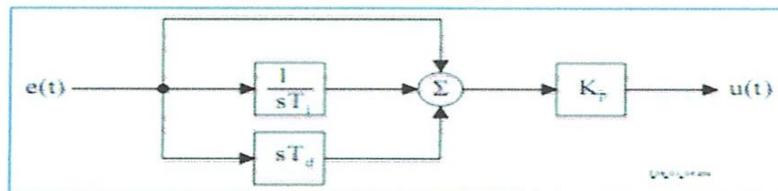


Figure 5.10 : Schéma fonctionnel du régulateur PID

❖ Simulation en mode ABS régulé via PID

Dans ce mode de simulation nous allons concevoir un régulateur PID via la plate-forme Stateflow en vue de stabiliser notre système de freinage et d'éliminer tout risque de patinage de la roue pendant le freinage.

La fonction de transfert de notre régulateur est la suivante :

$$\frac{e(p)}{u(p)} = Kp. \left(1 + \frac{1}{T_i \cdot P} + T_d \cdot P \right)$$

Nous avons tenté dans un 1^{er} temps de réaliser la régulation d'un système ABS à l'aide des paramètres expérimentaux via un PID. Sachant que les valeurs données aux paramètres du régulateur sont :

- Proportional =1
- Integral =1.9
- Dérivative = 3

Nous obtenons par la suite les courbes suivantes :

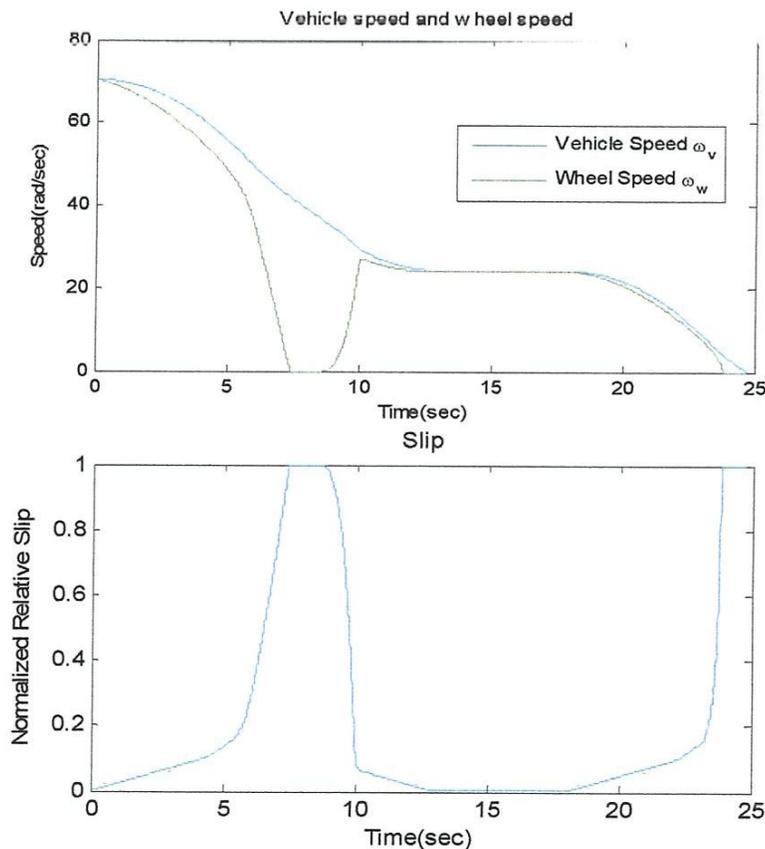


Figure 5.11 : Résultat de la simulation d'un PID avec des valeurs expérimentales (non idéal)

Afin d'obtenir une régulation idéale, nous avons réalisé un programme représentant un régulateur PID sous la plate-forme Stateflow contenu dans un bloc Chart ayant pour entrée le rapport d'erreur entre la sortie et l'entrée du système, et comme sortie le rapport régulé.

```

(Kp=1;Ti=0;Td=1.8;)}
PID_analogique/
during:
u_dot=err*(Kp*(1+(1/Ti)+Td));
u_out=u_dot;

```

Figure 5.12: programme du PID dans le bloc Chart

Les résultats de la simulation sont les suivants :

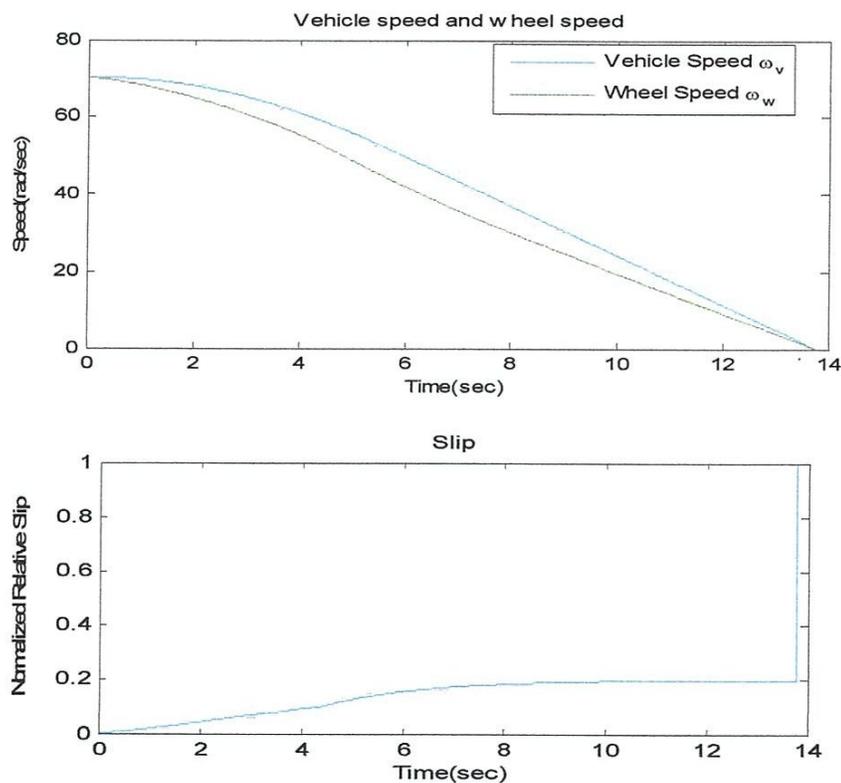
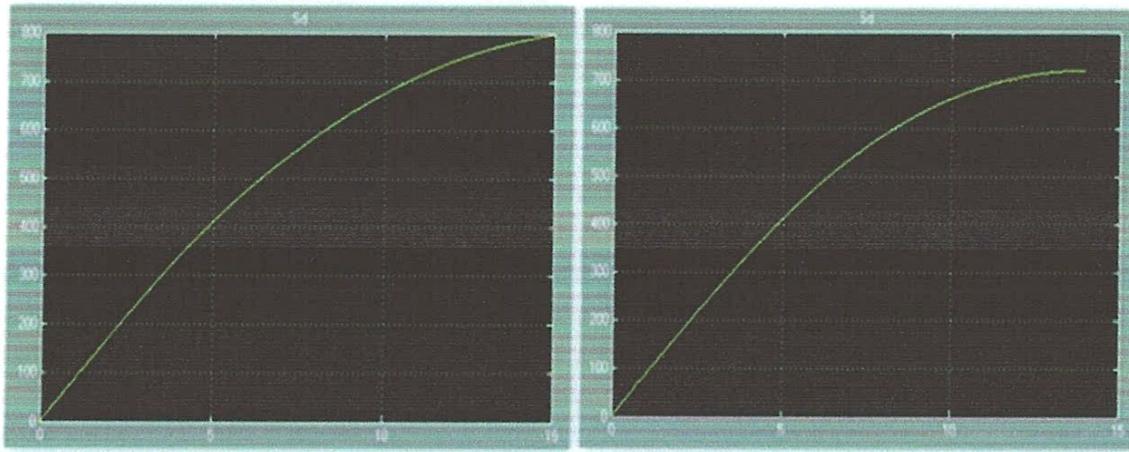


Figure 5.13 : Résultat de la simulation en mode ABS avec PID

Comme le montrent les résultats, l'ajout d'un régulateur PID permet de s'améliorer grandement la réponse du système et de synchroniser la vitesse de la roue et celle exigée par le dispositif pendant la phase de freinage.

5.5 Comparaison entre le freinage Sans ABS et Avec ABS



Sans ABS

Avec ABS

Figure 5.14 : distance d'arrêt en freinant brutalement avec et sans ABS

En termes de comparaison la figure 5.14 nous montre la distance parcourue par le véhicule lors pour les deux cas. Sans ABS, le véhicule dérape sur un supplément de 100 pieds, en prenant environ trois secondes de plus à venir à un arrêt.

5.6 Conclusion

En conclusion l'ABS permet d'améliorer la stabilité et la directibilité du véhicule en réduisant le niveau de blocage des roues pendant le freinage.

Nous pouvons affirmer à la suite de notre étude que le système ABS est un outil d'assistance au freinage fiable et performant, mais comme tout système dynamique il comporte certaines insuffisances. Notre travail tout au long de ce chapitre a consisté à les mettre en évidence et à proposer des solutions optimales à l'aide de régulateurs en vue d'obtenir des systèmes plus stables et plus sécuritaire pour les conducteurs.

Conclusion générale

Conclusion Générale

En conclusion, la majorité des systèmes technologiques de notre époque ont une nature hybride dans le sens où ils possèdent à la fois une dynamique discrète et une dynamique continue. Ces systèmes peuvent être de différents types, comme les systèmes hydrauliques et les systèmes manufacturiers traitant une quantité importante de produits. En effet l'étude des SDH a retenu l'attention des chercheurs automaticiens que ce soit ceux qui s'intéressent aux systèmes continus ou ceux qui étudient les systèmes à événement discrets (*SED*).

L'objectif de l'étude des Systèmes dynamiques hybrides consiste à apporter une solution en termes de modélisation, de méthode d'analyse et des performances à des problèmes que l'on ne peut pas traiter avec les méthodes classiques de l'automatique des systèmes continus ou des systèmes à événement discrets. Ainsi devant la complexité de ces systèmes, l'automaticien est souvent amené à développer de nouvelles méthodes qui prennent en compte les deux dynamiques du SDH, continue et discrète, et de leurs Interactions.

Au cours de ce mémoire, nous avons exposé dans le premier chapitre l'étude détaillée des Systèmes Dynamiques Hybrides. En effet, ces systèmes sont composés essentiellement d'un mélange de deux composantes (continue et discrète) interagissent entre eux, et où leur interaction détermine le comportement qualitatif et quantitatif de ces systèmes. Puis dans un deuxième temps nous avons cité les deux principales classes des systèmes hybrides présentées dans les SDH à commutation autonome et les SDH à commutation contrôlée. Et par la suite pour éclaircir ces notions, nous avons cité quelques exemples illustratifs classiques.

Le deuxième chapitre a été destiné à la présentation du contexte général dans lequel s'inscrit l'approche de diagnostic que nous proposons. Après quelques définitions sur les terminologies employées dans le domaine du diagnostic nous décrivons les concepts fondamentaux des approches de diagnostic à base de modèle issues des systèmes continus, système à événement discret et des systèmes dynamiques hybrides.

Le troisième chapitre compte à lui introduit des notions fondamentales relatives à la modélisation des SDH. En utilisant les méthodes et les spécifications de fonctionnement des systèmes continus et discrets ensuite les

Conclusion Générale

outils de modélisation des SDH dont nous avons besoin à savoir les réseaux de Petri et les automates.

Ensuite, nous avons le quatrième chapitre qui a été consacrée à l'explication détaillé et explicite de la modélisation grâce l'outil Stateflow. Afin de permettre une meilleure compréhension, nous avons aussi énoncé des exemples illustratifs démontrant étape par étape de la modélisation d'automate hybride.

Et enfin, le cinquième chapitre de notre mémoire est consacré à notre cas d'étude, cas d'étude qui est basé sur la modélisation de l'Anti-lock Braking System (ABS). Ainsi suite à la modélisation de l'ABS grâce à l'outil Stateflow, nous avons par la suite réalisé de la simulation en mode ABS et sans ABS.

En définitive, un système hybride est un système dynamique qui présente un comportement à la fois discret et continu. Les systèmes hybrides trouvent leur application au niveau des systèmes industriels comme les robots et le monde automobile les systèmes micro-électroniques et les équipements médicaux.

Bibliographie

- [1]: Alur et al, 95 R Alur, C Courcoubetis, N Halbwachs, TA Henzinger, <<The algorithmic analysis of hybrid systems>>, Theoretical Computer Science pp. 3-34, 6 Février, 1995.
- [2]: A. Alur, C. Courcoubetis, T.A. Henzinger, and P.H. Ho, <<Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems>>, In Hybrid Systems, LNCS, 736: 209-229, 1993.
- [3]: [And 96] D. Andreu, J.C.Pascal and R. Valette, Events as a key of a process control system, in: Proc. CESA'96, Symp. On Discrete Events and Manufacturing Systems, Lille, France, 1996.
- [4]: Antsaklis et al, 97. P.Antsaklis, X. Koutsoukos, J. Zaytoon, <<On hybrid control of complex system: a survey>>, Technical Report of the ISIS Group at the University of Notre Dame ISIS-97-017, Notre Dame, janvier 1997.
- [5]: Balluchi, A., L. Benvenuti, M.D.D. Benedetto et A. L. Sangiovanni-vincentelli. 2002, <<Design of observers for hybrid systems>>, dans Proceedings of Hybrid Systems: Computation and Control, volume 2289 of LNCS, Springer-Verlag, p. 76-89.
- [6] : Bayouhd, M., & Travé- Massuyès, L 2009. An algorithm for active diagnosis of hybrid systems casted in the DES Framework. 2nd IFAC workshop on Dependable Control of Discrete Systems, Bari, Italy.
- [7] : Bon 01, P.Bonhomme. Réseaux de Petri P-Temporels : Contribution a la Commande Robuste.
- [8]: Bhowal, P., Sarkar, D., Mukhopadhyay, S., & Basu, A. (2007). Fault diagnosis in discrete time hybrid systems – a case study. Information Sciences, vol. 177, n° 5, p. 1290-1308.
- [9]: [Bra 95] M. S. Branicky. Studies in hybrid systems: Modling, Analysis, and control. Phd thesis, Massachusetts Institute of Tech., 1995.
- [10]: Chow, E. et A. Wilsky. 1984, <<Analytical redundancy and the design of robust failure detection system>>, IEEE transactions on Automatic and Control, vol. 29, n° 7, p. 603- 614.
- [11]: Cocquempot, V., T. E. Mezyani et M. Staroswieckiy. 2004, <<Fault detection and isolation for hybrid systems using structured parity residuals>>, dans Asian Control Conference, ASCC'04, vol. 2, New Mexico, p. 1204-1212.

Bibliographie

- [12]: De Flaugergues, V., Cocquempot, V., Bayart, M., & Lefebvre, N. 2009. Exhaustive search of residuals computation schemes using macro-graphs and invertibility constraints. IFAC Safeprocess, Barcelona, Spain.
- [13] : Derbel, H 2009. Diagnostic à base de modèle des systèmes temporisés et d'une sous- classe de systèmes dynamiques hybrides. Thèse de Doctorat, Université Joseph Fourier- Grenoble I et l'école nationale des sciences de l'informatique, France et Tunisie.
- [14]: Domlan, E., D. Maquin et J. Ragot. 2004, <<Diagnostic des systèmes à commutation, approche par la méthode de l'espace de parité>>, dans Conférence Internationale Francophone d'Automatique, Tunisie.
- [15] : El Mezayani, T 2005. Méthodologie de surveillance des systèmes dynamiques hybrides. Thèse de doctorat, Ecole Polytechnique Universitaire de Lille, Lille, France.
- [16]: Feng, S.N., F. Zhao, G. Biswas et E. Hung. 2000, <<Fault isolation in hybrid systems combining model based diagnosis and signal processing>>, dans Proceedings of the 4th IFAC Symp. SAFEPROCESS, p. 1074-1079.
- [17]: Furlan, K., K. Kyriakopoulos et N. Krikoris. 2002, "Diagnosability of hybrid systems", dans Proceedings of MCCA'02, p. 3994-3999.
- [18] : Guéguen, H., & Lefebvre, M.A. 2001. Acomparision of mixed specification formalisms. Journal Européen des systèmes Automatisés. JESA, vol.35, pp.381-394.
- [19]: Gomaa, M. et S. Gentil. 1996, <<Hybrid industrial dynamical system supervision via hybrid continuous causal petri nets>>, dans CESA'96 IEEE SMC IMACS Symposium on Discrete Events and Manufacturing Systems, Lille, France, Springer-Verlag, p. 380-384.
- [20]: Gomaa, M. 1997, Représentation et Supervision des Systèmes Hybrides par Réseaux de Petri, thèse de doctorat, Institut national polytechnique de Grenoble, Grenoble.
- [21] : Haithem DERBEL, "Diagnostic à base de modèle des systèmes temporisés et d'une sous- classe de systèmes dynamiques hybrides" Université Joseph Fourier- Grenoble I.
- [22]: [Has 99] S. Hashtrud. Fault Diagnosis in discrete-event and hybrid systems. Thesis, Toronto University, Canada, Septembre 1999.

Bibliographie

- [23]: Iserman, R. 1984, <<Process fault detection based on modeling and estimation methods – a survey>>, Automatica, vol. 20, n° 4, P. 387-404.
- [24]: J. Le Bail, H. Alla and R. David, "Hybrid petri nets", In Proceedings of the European Control Conference, Grenoble, France, pages: 1472-1477, 1991.
- [25]: Jones, H. L. 1973, Failure detection in linear systems; these de doctorat, Massachusetts Institute of Technology.
- [26]: Karsai, G., S. Abdelwahed et G. Biswas. 2003, <<Integrated diagnosis and control for hybrid dynamic systems>>.
- [27] : KHAOUNI Sofiane, "Modélisation et Surveillance des Systèmes Dynamiques hybrides par Automates Hybrides" Université Ferhat Abbas- Sétif.
- [28]: Koutsoukos, X., F. Zhao, H. Haussecker, J. Reich et P. Cheung. 2001, <<Fault modeling for monitoring and diagnosis of sensor-rich hybrid systems>>, dans Proceedings of the 40th IEEE Conference on Decision and Control, p. 793-801.
- [29]: Lunze, J. 2000, <<Diagnosis of quantized systems>>, dans 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, SAFEPROCES'00, p.28-39.
- [30]: Lunze, J. 2006, <<Diagnosis of discretely controlled continuous systems>>, Automatisierungstechnik, vol. 54, n°8, p. 385-395.
- [31]: Lin, F et W. M. Wonham. 1994, <<Diagnosability of discrete event systems and its applications>>, Discrete Event Dynamic Systems, vol. 4, n° 2, p. 197- 212.
- [32]: Lin, F. et W. M. Wonham. 1988, <<On observability of discrete- event systems>>, Information sciences, vol. 44, n° 3, p. 173- 198.
- [33]: MERAHI Kamel, "Estimation d'état et Diagnostic de Fonctionnement des systèmes non Linéaires" Université Badji Mokhtar Annaba.
- [34]: Mez 05 T.EL Mezyani. Méthodologie de surveillance des systèmes dynamiques hybrides. Thèse de doctorat, spécialité : automatique et informatique industrielle, préparé au laboratoire d'automatique Génie informatique et Signal UMR CNRS 8146 de l'université des sciences et technologies de Lille, 2005.
- [35]: Mosterman, P.J. 1997, Hybrid Dynamic Systems: a Hybrid Bond Graph Modeling paradigm and its Application in Diagnosis, these de doctorat, Vanderbilt University.

Bibliographie

- [36]: R. Isermann and P. Ballé, Terminology in the _eld of supervision, fault detection and diagnosis, Technical Committee of Safeprocess' 97 (August, 1997).
- [37] : R. Toscano, Commande et diagnostic des systèmes dynamiques, Edition Ellipses Paris, 2005.
- [38]: Sait. Belkacem, "Modélisation, Simulation et Surveillance des Systèmes Dynamiques Hybrides utilisant les résidus de parité et réseaux de pétri", UFAS, Université Ferhat Abbas, Sétif, 2007.
- [39]: Sampath, M., R. Sengupta, S. Lafortune, k. Sinnamohideen et D. Teneketzis. 1995, <<Diagnosability of discrete event systems>>, IEEE Transactions on Automatic Control, vol 40, n° 9, p 1555-1575
- [40]: Sampath, M., R. Sengupta, S. Lafortune et K. Sinnamohideen. 1996, <<Failure diagnosis using discrete event models>>, IEEE Transactions on Control Systems Technology, vol. 4 n° 2, p. 105-124.
- [41]: Travé- Massuyès, L., Dague, P., & Guerrin, F 1997. Le raisonnement qualitatif pour les sciences de l'ingénieur. Edition Hermès, Paris, France.
- [42]: Tripakis, S. 2002, <<Fault diagnosis for timed automata>>, dans Proceedings of the 7th International Symposium on Formal Techniques in Real- Time and Fault-Tolerant Systems, FTRTFT'02, Springer- Verlag, London, UK, ISBN 3- 540- 44165- 4, p. 205- 224.
- [43]: Y. Kesten and A. Pnueli, "Timed and hybrid statecharts and their textual representation", formal techniques in real- time and fault- tolerant systems, LNCS, 571: 591-620, 1992.
- [44]: Zad, S. H. R. H. Kwong et W. M. Wonham. 2003, <<Fault diagnosis in discrete-event systems: Framework and model reduction>>, IEEE Transactions On Automatic Control, vol. 48, n° 7, p. 1199- 1212.
- [45]: Zad, S., R. Kwong et W. M. Wonham. 2005, <<Fault diagnosis in discrete-events systems: Incorporating timing information>>, IEEE Transactions On Automatic Control, vol. 50, n° 7, p. 1010-1015.

Remerciements

Ce mémoire a été réalisé à l'Université 08 Mai 1945 Guelma, au niveau du domaine Sciences et Technologie via la filière Automatique et Informatique industrielle.

Je souhaite exprimer d'abord ma reconnaissance à mon encadreur de mémoire, pour tout ce qu'elle m'a apporté pendant ces six mois de rédaction et de travail de recherche. Merci de m'avoir permis d'effectuer cette thèse, de m'avoir soutenu et conseillé tout au long de ce travail. Je vous remercie également pour vos nombreuses remarques et corrections qui ont pu améliorer la de mon travail. Mercl Dr. Boucerrej Leila.

Je tiens ensuite à remercier Gnango Frederic et Otoubou Julasvic de mes binômes au cours de ce travail de mémoire, d'avoir de très bon collègue de travail et pour tout l'aide apporter lors de la réalisation de ce travail. Un grand merci à eux pour les bons et mauvais moments partagés lors de la rédaction...

Je remercie par la suite mes amis proches sur lesquels j'ai pu compter sur le soutien moral et non financier. Une spéciale dédicace : Ibara Pitterson A.K.A Fibre optique, Bokayaka Dulcy A.K.A Yayos récupérateur ya témbé, Mafoula Mechac A.K.A l'emblématique, Banzoulou Chris A.K.A Djo loboko général Toufik, Semi laurna, ampoukélé Galia, Matalic Confiance A.k.A l'élément rare..... Je n'oublie pas les autres dont les noms n'ont pas été mentionné, paniquez pas.

Enfin j'adresse mes sincères remerciements à ma famille. Spécialement à :

- A ma source d'inspiration, de motivation et bénédiction ma mère Mme IKONGA née ETSEKENDZOTO Joséphine.
- A mon Père IKONGA rémy, cette source sérénité et de patience.
- A mes frères et sœurs, Pour votre soutient moral et non financier je précise car même à distance avez été présent. Et donc Eurydia M. , Laila I. , Tancrede I. et Uriel I. je vous remercie du plus profond de moi-même.
- A mes neuves et nieces un petit clin d'œil à vous, Owen-Shalaan-Yael-Kate...

Evarad IKONGA