République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERI, Tizi-Ouzou

Faculté de Génie Electrique et d'Informatique
Département d'Automatique

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

Master Académique en Automatique
Option : Commande des systèmes

Thème

Modélisation hybride du convertisseur buck et boost

Proposé et dirigé par **M^me^ : F.Louadj**

M^lle^ :KEZMANE DJEDJIGA
M^lle^ :BOUKHALFA DJAZIA

Promotion 2013

Remerciement

*A*Travers ce modeste travail, nous tenons à remercier vivement notre promotrice **Meme louadj** pour l'intéressante documentation qu'il mise à notre disposition, pour ses conseils précieux et pour toutes les commodités et aisances qu'elle nous a apportées durant notre étude et réalisation de ce projet.

Nos remerciements les plus vifs s'adressent aussi aux messieurs le président et les membres de jury d'avoir accepté d'examiner et d'évaluer notre travail.

Nous exprimons également notre gratitude à tous les professeurs et enseignants qui nous aident à réaliser ce travail, sans omettre bien sur de remercier profondément tous ceux qui ont contribué de près ou de loin à réaliser du présent travail.

Et enfin, que nos chers parents et familles, et bien avant tout, trouvent ici l'expression de nos remerciement les plus sincères et les plus profonds en reconnaissance de leurs sacrifices, aides, soutien et encouragement afin de nous assurer cette réussite.

Dédicaces

Je dédie ce modeste travail à :

Mes très chers parents qui m'ont toujours soutenue

Mes frères MALIK, SOFIANE, MAHMOUD et ANIS

Mes chères sœurs RAFIKA, SIHAM, FERIEL et KATIA avec son mari ZOUBIR et ma petite adorable nièce MALEK

Mes oncles et mes tentes et leurs familles

Mes cousins et cousines

Mes camarades en particulier MOUNIR, AREZKI, MOUKRANE

Tous mes amis(es) en particulier HASSINA, SAKINA CHERIFA HOURIA, MALIKA, NASIRA et leurs familles

A tous ceux qui ont su croire en moi

A tous ceux qui me sont chers

DJAZIA

Dédicaces

Je dédie ce modeste travail à ***mon père***

A ***ma chère mère*** *qui a tout l'honneur d'arriver à ce succès*

A mes frères ***SAID*** *et* ***AHMED*** *et* ***AHCENE*** *et* ***HOCINE***

A mes sœurs ***CHABHA*** *et* ***OUIZA*** *et* ***SALIHA*** *et* ***RACHIDA***

A mes neveux ***YACINE*** *et* ***AMAR*** *et* ***SAMIR*** *et* ***LOUNES*** *et* ***YANI*** *et* ***YAHIA*** *et* ***MENAD***

A toute personne qui m'a aidé du loin ou du près pendant ma vie scolaire et universitaire

Djedjiga Kezmane

Introduction Générale

Introduction générale 01

Chapitre I : Généralités sur les systèmes dynamiques hybrides.

I.1. Introduction 05

I.2. Définition des systèmes dynamiques hybrides. 05

I.3. Caractéristiques des systèmes 05

I.4. Les types de système hybride. 08

I.4.1. Système avec élément de commutation 08

I.4.2. Système continus pilotés par calculateur discret 09

I.4.3. Systèmes dynamiques impulsionnels. 10

I.5. Classification des systèmes hybrides proposée par M.S. Branicky. 11

I.5.1. Interactions entre les systèmes 12

I.5.2. Action des phénomènes hybrides. 12

I.5.3. Déclenchement des phénomènes hybrides. 13

I.5.4. Classifications des phénomènes. 14

I.5.4.1. Une commutation autonome 14

I.5.4.2. Sauts autonomes. 14

I.5.4.3. Commutations contrôlées. 15

I.5.4.4. Sauts contrôlées. 16

I.5.5. Exemple de phénomène hybride. 16

I.6. Représentation formelle des systèmes hybrides 17

I.6.1. Système dynamique continue 17

I.6.2. Les systèmes dynamiques discrets 18

I.6.3. Système à événement discret 18

I.7. Système hybride 18

I .7.1. Définition 18

I.7.2. Trajectoire d'état hybride. 20

I.8. Conclusion 21

Chapitre II : Modélisation et commande des systèmes dynamiques hybrides

II.1. Introduction 22

.II.2.1 Les automates à états finis 22

II.3 Les automates temporisés 24

II.4 Les automates hybrides 26

II.4.1 Définition 26

II.4.2 Exemples de modélisation par automate hybride 28

II.4.3 Types de représentation de l'automate hybride 30

II.4.3.1 Modèle analytique 30

II.4.3.2 Modèle graphique 32

II.4.3.3 Interconnexion d'automate 34

II.5 Modélisation des systèmes hybrides par les systèmes à interfac 35

II.5.1 Structure modulaire générale (modèle d'Antsaklis) 36

II.5.2 Structure adaptée 38

II.5.3 Le modèle de Brockett 40

II.5.4 Modèle à interface généralisé 43

II.6 modélisation par réseaux de Petri 46

II.6.1 Notion de base des réseaux de Petri 46

II.6.2 Les Réseaux de Petri discrets (RDPD) 47

II.6.3 Règle de fonctionnement 49

II.6.4 Les Réseaux de petri continus (RDPC) 49

II.6.5 Les réseaux de petri T-temporels 49

II.6.5.1 Les réseaux de perti temporisés 50

II.7 Méthodes de commande des systèmes hybrides 52

II. 7.1 Commande à base de modèles moyens 52

II.7.1.1 Quelques définitions 52

II.7.2 Problème de commande optimale des SDH 55

II.8 Formulation du problème de commande 55

II.9 Commande prédictive des SH sous forme MLD 56

II.9.1 La forme MLD 56

II.10 Conclusion 58

Chapitre III : Application

III-1.Introduction 59

III-2.Description de système étudié 59

III-2.1.Les convertisseurs continu-continu 59

A)- Le convertisseur Buck 59

B)-Le convertisseur Boost 60

III-3.Modes de fonctionnement 60

III-3.1.Le fonctionnement de convertisseur Buck 61

III-3.2.Le fonctionnement de convertisseur Boost 64
III-4.Modèle d'état 67
III-5.Modèle Hybride 68
III-6. Simulation 69
III-6.1 Schéma global 69
III-6.2 Le Statechart de convertisseur Back 70
III-6.2.1 Exécution et résultats graphiques de Back 72
III-6.3 Le Statechart de convertisseur Boost 73
III-6.3.1 Exécution et résultats graphiques de Boost 74
III-7 Conclusion 75

Conclusion générale

Conclusion générale et perspectives 76

Bibliographie

Bibliographie 78

Le développement massif de l'automatisation au XXe siècle s'est longtemps fait suivant deux types de technologies mises en œuvre selon des méthodologies, qui leur étaient propres et de nature différente : les systèmes à événements discrets et les systèmes continus.

Les systèmes à événements discrets sont des systèmes à éléments caractérisés par deux états: ouvert ou fermé, marche ou arrêt, sortie ou entrée... L'appareillage « tout ou rien » (TOR) correspondant est classiquement géré en s'appuyant sur l'algèbre de Boole, puis, en prenant en compte a chronologie des états, leur séquencement, par des méthodes états-transitions accompagnés d'une représentation graphique : graphes d'état, réseaux de Petri, Grafcet.

Quant aux systèmes continus, ils sont constitués d'éléments caractérisés par une mesure qui peut prendre une infinité de valeurs : température d'une pièce ou d'un objet, vitesse d'un mobile, niveau dans un réservoir...etc. La gestion de ces systèmes fait appel à des outils mathématiques aptes à la représentation de la dynamique continue : équations différentielles assorties de diverses transformations (Laplace, Fourier...etc.), méthodes d'état à forme matricielle. **[9]**

Certains systèmes présentent à la fois des aspects continus et discrets : il s'agit des systèmes dynamiques hybrides (SDH).Le suivi de ces systèmes est un enjeu majeur de la recherche scientifique. En effet la plupart des systèmes réels ont un comportement hybride provenant de la dynamique du système et d'action externe. Par exemple d'un opérateur humain ou artificiel.

Depuis plus de quinze ans et de plus en plus, les SDH prennent un grand intérêt dans le domaine de la recherche puisqu'on les retrouve dans de nombreuses applications : Systèmes mécaniques, électroniques et électroniques de puissance, robotique, industrie automobile, etc., qui sont des systèmes avec un couplage entre les dynamiques continues et les dynamiques événementielles. Ils présentent des progrès théoriques intéressants dus à leurs multidisciplinarités naturelles et jouent un rôle important dans de nombreux problèmes réels. Le caractère hybride du système peut provenir du système lui-même ou de sa commande.

De nombreux problèmes mal traités par les approches homogènes sont résolus par l'approche hybride :

– variation de structure liée aux différents modes de fonctionnement du système ;

– variation de paramètres au cours de fonctionnement du système ;

– commande discontinue (par exemple par relais ou impulsions) ;

– modélisation de phénomènes transitoires rapides par des commutations de modèles (ouverture ou fermeture des vannes dans les procédés hydrauliques, blocage et amorçage des semi-conducteurs dans les convertisseurs de puissances,...).

D'où l'intérêt de l'étude de ces systèmes par cette approche est de comprendre et maîtriser les phénomènes complexes engendrés par les systèmes hybrides qui a toujours été une tâche difficile mais fortement nécessaire pour pouvoir simuler, contrôler, estimer et anticiper leur comportement. Ceci a incité les chercheurs de disciplines différentes et surtout ceux d'automatique, informatique et mathématiques à développer des approches interdisciplinaires de modélisation, d'analyse et de commande de ces systèmes.

Les systèmes dynamiques à commutations (SDC) représentent une classe de systèmes dynamiques hybrides composée d'une famille de sous-systèmes à dynamique continue représentée par des équations différentielles et d'une loi logique qui donne le sous-système actif. La motivation pour l'étude de ces systèmes vient du fait :

– que certains systèmes, en pratique, sont fondamentalement multi-modèles, composés de plusieurs sous-systèmes, dans le sens où plusieurs dynamiques de systèmes apparaissent dans le comportement suite à des changements d'état d'actionneurs discrets par exemple.

– que certaines méthodes de contrôles intelligentes sont basées sur l'idée de commutation entre plusieurs contrôleurs.

– et qu'il faut bien orchestrer les commutations suivant le fonctionnement désiré du SDC. **[10]**

Les systèmes dynamiques hybrides sont prédominants dans le domaine de l'électronique de puissance et en particulier sur les convertisseurs de puissance DC-DC (courant continu-courant continu) qui sont caractérisés par une structure variable (en raison du changement de la topologie du circuit selon les états passant ou bloqué des transistors et diodes). En conséquence, les convertisseurs DC-DC à découpage sont considérés comme des systèmes hybrides pour lesquels la partie discrète est représentée par les commutations à haute-fréquence des transistors, alors que les courants et les tensions au sein du système sont décrits par des équations dynamiques à variables continues.

L'exploitation des techniques de modélisations hybrides a connu un développement important afin de prendre en compte le comportement hybride de convertisseurs continu-continu, assuré par les commutations du transistor et les dynamiques continues associées à ces commutations, et de reproduire le comportement dynamique du système réel. Grâce à ces

méthodes de modélisation hybride, il est possible de tenir compte à la fois de l'évolution en temps continu des variables d'état du système et de l'effet des éléments discrets.

Le principe consiste à exprimer les configurations du convertisseur en utilisant un modèle à temps continu ou discret et d'utiliser une expression ou une variable logique pour sélectionner le modèle de la configuration active à chaque instant. **[2]**

Parallèlement aux progrès de l'informatique, les ingénieurs se sont attaqués à des problèmes de plus en plus complexes, par le nombre de grandeurs qui interviennent et par leurs interactions. On rencontre ainsi de nombreux cas où des événements, volontaires ou subis, ou l'atteinte d'un seuil entraine des modifications du comportement de grandeurs continues : changement de valeur ou commutation d'un modèle continu à un autre.

Il devient ainsi nécessaire de mettre au point des méthodes capables de prendre en compte tous les types de grandeurs ainsi que leurs interactions. On appelle systèmes dynamiques hybrides (SDH), ou encore processus mixtes, les systèmes comportant des évolutions continues et des phénomènes discrets qui leur sont liés. Il n'existe pas pour l'instant de théorie globale pour l'étude de ces systèmes, mais plutôt des approches basées sur l'extension de méthodes classiques issues des systèmes continus ou discrets, en vue de couvrir une gamme plus étendue d'applications. Après avoir fixé la notion de système hybride par quelques exemples, qui serviront aussi à préciser la typologie, nous examinons les outils de description et de modélisation des systèmes dynamique hybride. Le fait de disposer d'un modèle cela permet de simuler le comportement du système dynamique hybride pour le prévoir, l'influencer et chercher à l'optimiser. Il permet aussi, sous certaines conditions, de l'analyser pour détecter a priori d'éventuels disfonctionnements et/ou d'en faire la commande pour obtenir les performances souhaitées.

Trois approches de modélisations des SDH ont été développées. La première est basée sur une extension des techniques de modélisation issues des systèmes dynamique continus comme les bonds graphs à commutation, le formalisme MLD (Mixed LogicalDynamical). La deuxième est basée sur une extension des techniques de modélisation issues des SED, comme les réseaux de petri hybrides. La troisième combine les deux techniques de base précédentes en les interfaçant pour prendre en compte l'aspect hybride du système, comme les automates hybrides, les réseaux de petri mixte…etc. **[6]**

Comme pour les modèles continus et les modèles discrets, la démarche classiquement utilisée pour valider un modèle hybride est la **simulation.**

Plusieurs logicielles et langages de programmation ont été développés pour modéliser et simuler les systèmes hybrides. Nous pouvons citer : Dymola, Simulink/Stateflow, HyBrSim, gPROMS (ce sont des logiciels destinés à simuler des systèmes dynamiques continus augmentés d'éléments à événements discrets (SED) (système à événement discret).

Ce mémoire est consacré à la modélisation et la simulation des systèmes hybrides à l'aide de MATLAB/Simulink/Stateflow. Elle est composée de trois chapitres :

- Dans le premier chapitre, nous citons les différents SDH les plus souvent rencontrés dans la littérature. Puis, nous proposons une classification des systèmes hybrides, dont les principes ont été donnés par M.S.Branicky. Ensuite, nous donnons une représentation formelle des SDH.
- Le deuxième chapitre est consacré à la modélisation et la commande des systèmes hybrides. Nous présentons dans un premier temps les automates hybrides et les réseaux de petri qui sont utilisés pour modéliser les SDH sous deux formalismes, et par la suite on a donné quelques commandes des SDH
- Enfin, dans le troisième chapitre nous allons mettre en œuvre Stateflow pour simuler le comportement d'un système hybride qui est un convertisseur DC-DC. Le processus (partie opérative) est formé de l'inductance et de capacité qui se charge et se décharge.
- Nous terminerons ce mémoire par une conclusion générale et quelques perspectives.

I.1 Introduction

Dans ce premier chapitre, nous présentons les notions fondamentales nécessaires à la compréhension des systèmes dynamiques hybrides (S.D.H). Leur définition, les caractéristiques des systèmes et les principaux types et phénomènes qui interviennent dans les systèmes dynamiques hybrides seront présentés, ainsi qu'une classification des systèmes proposée par M.S.Branicky et nous terminons par une représentation formelle des SDH.

I.2 Définition des systèmes dynamiques hybrides

Avant de donner une définition formelle des SDHs, il convient de mieux situer les systèmes par-rapport aux systèmes purement discrets ou purement continus. En effet, pour de nombreux systèmes appelés hybrides, l'aspect continu ou discret est tellement dominant que l'étude sous une approche hybride n'apporte rien de significatif. C'est pourquoi il nous semble nécessaire de déterminer les SDH pour lesquels cette étude sous une approche hybride est intéressante.

Pour cela, nous rappelons rapidement les caractéristiques fondamentales des systèmes, ce qui nous amènera à donner une classification dans laquelle on pourra isoler les systèmes que nous allons étudier.

Définition

Une première définition des systèmes dynamiques hybrides est basée sur la notion de variable d'état. Un système est dit hybride s'il possède deux types de variables d'état : des Variables d'état continues et des variables d'état discrètes. **[3]**

Dans cette partie, Nous allons voir qu'à partir de ces caractéristiques et notamment à partir de la notion de variable d'état, on peut classer ces systèmes en différents types

I.3 Caractéristiques des systèmes

- **Systèmes dynamiques et systèmes statiques**

Les systèmes statiques ont une sortie qui est toujours indépendante des valeurs passées de son entrée.

Les systèmes dynamiques ont une sortie qui dépend des valeurs passées de son entrée. Les équations différentielles sont requises pour décrire le comportement des systèmes dynamiques.

- **Systèmes stationnaires et systèmes non stationnaires**

Le comportement des systèmes stationnaires ne change pas avec le temps. Cela implique qu'à une entrée spécifique appliquée au système correspondra toujours à la même sortie.

- **Systèmes linéaires et systèmes non linéaires.**

Un système linéaire satisfait la condition suivante:

$$f(\lambda * x + \mu * y) = \lambda * f(x) + \mu * f(y) \quad \textbf{(I.1)}$$

Où :

x et y sont des vecteurs d'entrée, et μ et λ des scalaires et f(x) la sortie du système.

- **Systèmes avec des variables d'état continues et systèmes avec des variables d'état discrètes.**

Les variables continues prennent leurs valeurs sur l'ensemble des réels R, alors que les variables discrètes prennent leurs valeurs dans un ensemble fini de valeurs distinctes, par exemple un sous-ensemble des entiers naturels N. Le temps est une variable particulière qui peut également être soit continue, soit discrète. De plus, dans le temps discret, on distingue deux autres cas : les systèmes échantillonnés et les systèmes à événements discrets.

- **Systèmes à temps continu, systèmes échantillonnés, systèmes à événements discrets.**

Les systèmes à temps continu changent d'état continument lorsque le temps change. Les systèmes à temps discret ne changent pas d'état qu'à des instants précis. Lorsque ces instants sont synchrones, réglés par une horloge, on dit que ces systèmes sont échantillonnés. Lorsque ces instants sont asynchrones, on les appelle des événements. On définit alors les systèmes à événements discrets comme étant des systèmes à états discrets pilotés par des événements.

La Figure (I.1) représente une classification des systèmes d'après les caractéristiques décrites ci-dessus. Dans cette classification, on identifie cinq types de systèmes :

- Les systèmes linéaires continus notés **SLC**. Ces systèmes peuvent être modélisés par des équations différentielles.

- Les systèmes échantillonnés à états continus sont notés SE continus et ceux à états discrets sont notés **SE** discrets. Ces systèmes peuvent être modélisés par des équations aux différences.

- Les systèmes discrets notés **SD**. L'évolution des variables d'état peut se faire par exemple à partir de fonctions booléennes paramétrées dans le temps.

- Les systèmes non linéaires continus notés **SNLC**.

- Les systèmes à événements discrets notés **SED.** Ces systèmes peuvent être modélisés par des automates, des réseaux de Petri. **[B.CEBRON, 2000]**

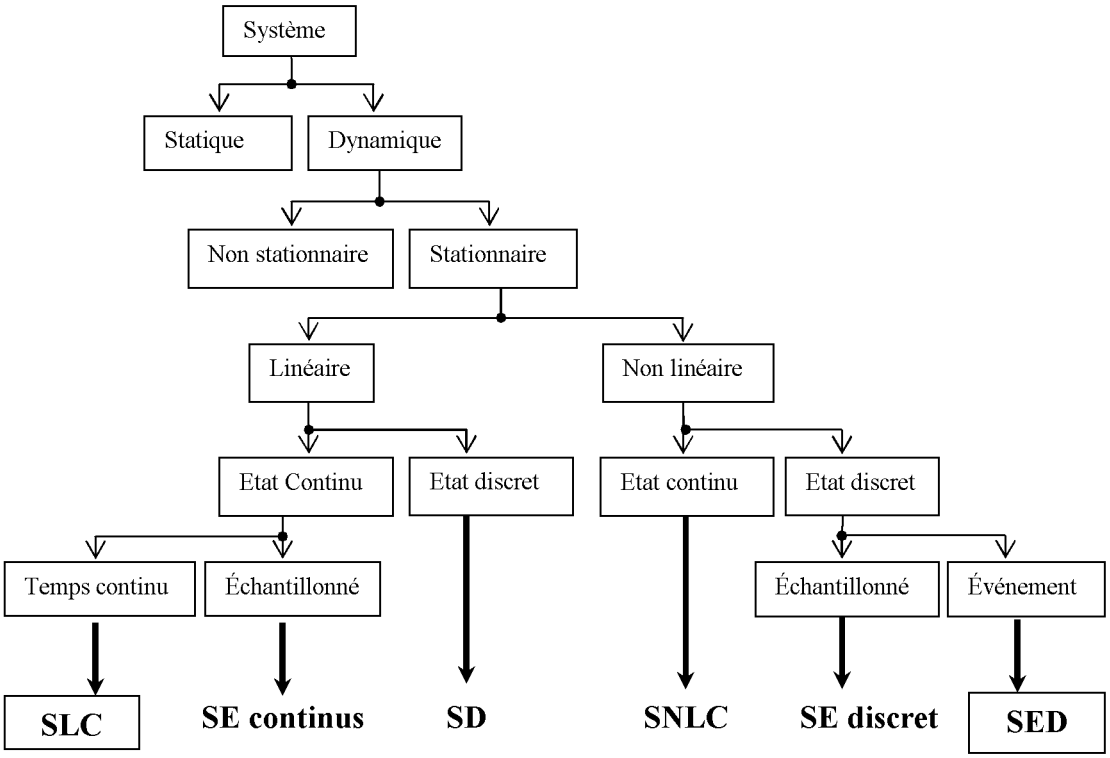


Figure (I.1) : Classifications des systèmes

I.4 Types de systèmes hybrides

Nous distinguons différents types de système hybride, parmi lesquels :

I.4.1 Système avec éléments de commutations

Les systèmes à commutations représentent une catégorie importante des systèmes dynamiques hybrides dont l'évolution est à la fois continue (dynamique différentielle) et événementielle (commutations). Les commutations provoquent des changements abrupts de modes de fonctionnement et peuvent dépendre de l'état et/ou entrées du système, de la structure du système ou encore, peuvent être aléatoires.

Un système à commutation est composé d'une famille de sous-systèmes à dynamique continue (de type équation différentielle ou équation récurrente) et une loi logique qui indique un sous-système actif. **[K.TAARIT, 2010], [A.GIRARD, 2004]**

Formellement le système à commutation est défini par :

$$\dot{x}(t) = f_i\big(x(t)\big)\,; \quad i \in \{1, \dots \dots k\} \qquad (I.2)$$

Tels que $x(t) \in R^n$ est l'état du système, la fonction $f_i : R^n \to R^n$ est continue ; i est choisi de telle sorte à ce qu'il y'ait un nombre de changements finis en un temps fini.

Exemple 1 : système mécanique

Le système mécanique de la Figure (I.2) montre une masse attachée à une paroi à l'aide d'un ressort. Cette masse, en ce déplaçant, va rebondir sur un ensemble ressort amortisseur. Dans ce cas le choc n'est pas instantané, donc nous avons deux dynamiques différentes. La transition entre les deux modes est déterminée par le contact entre les éléments (même position des points extrêmes).

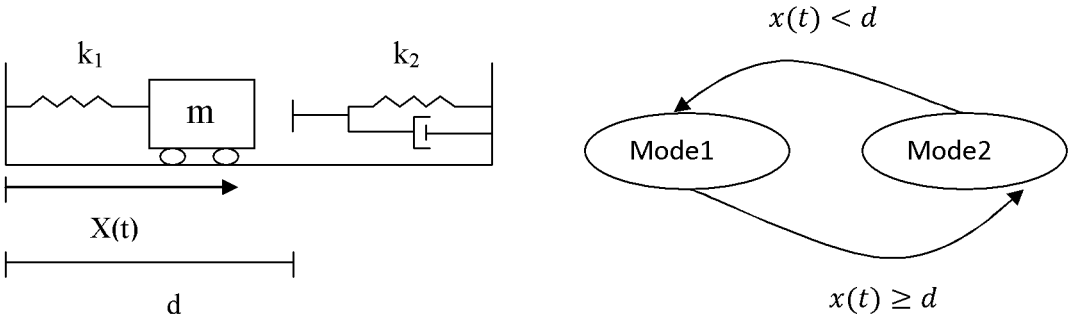


Figure (I.2) : Système mécanique

I.4.2 Système continus pilotés par calculateur discret

Cette classe est très répandue grâce au développement considérable des ordinateurs et des contrôleurs discrets.

Le système pilotés par des calculateurs sont hybrides : la dynamique continue est associe à la partie énergétique (mécaniques, hydrauliques ou électroniques) et la dynamique discrète est liée a la commande numérique et à l'existence d'événements discrets (défaillances, dépassement de seuil). **[6]**

Exemple 2

Ce système est représenté par la figure suivante :

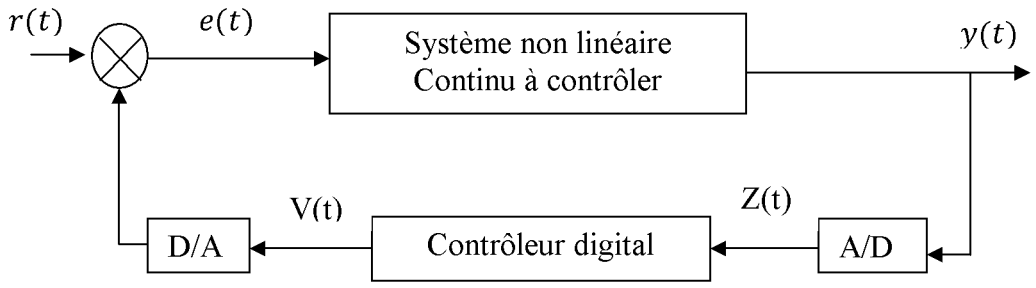


Figure (I.3) : Système non linéaire continu contrôlé par un contrôleur digital

Il peut être décrit par les équations suivantes :

Le système non linéaire est décrit par :

$$\begin{cases} \dot{x}(t) = f(x, e, t). \\ y(t) = h\big(x(t)\big). \end{cases} \qquad (I.3)$$

Le contrôleur digital est décrit par les équations aux différences suivantes :

$$\begin{cases} u[(k+1)T] = F[u(kT), Z(kT)]. \\ V(kT) = H[u(kT)]. \end{cases} \qquad (I.4)$$

En général, les convertisseurs D/A (digital/analogique) et A/D (analogique/digital), sont considérés idéaux et synchronisés dans le temps (T est la période d'échantillonnage), nous avons alors :

$$\begin{cases} e(t) = V(kT) = H\big(u(kT)\big). \\ Z(kT) = y(kT) = h\big(x(kT)\big). \end{cases} \quad (I.5)$$

Le système donc est décrit par :

$$\begin{cases} \dot{x}(t) = f\big(x, H\big(u(kT)\big), t\big). \\ u[(k+1)T] = F\big[u(kT), h\big(x(kT)\big)\big]. \end{cases} \quad (I.6)$$

Dans ce cas, la partie continue est l'état du système et la partie discrète est la commande qui est continue par morceaux (continue sur une période d'échantillonnage).

I.4.3 Systèmes dynamiques impulsionnels

Les systèmes à impulsions sont considérés comme des systèmes hybrides spécifiques. Ces systèmes sont caractérisés par des impulsions singulières ou généralisées et constitués de trois éléments :

1. un ensemble d'équations différentielles (états continus), dont chacune régit le mode de mouvement du système dynamique global entre deux impulsions (les impulsions sont les évènements).
2. un critère régissant l'évolution temporelle des impulsions (évolution des évènements).
3. une équation différentielle qui régit la façon dont les états du système changent au moment de l'impulsion. Par hypothèse de modélisation, ce changement est simultané.

Cette classe peut se décrire par les équations suivantes :

$$\begin{cases} dx/dt = f(x,t) & t \neq \tau_k \\ \Delta x = I_k(x) & t = \tau_k \end{cases} \quad (I.7)$$

Tel que $x(t) \in R^n$ est le vecteur d'état du système, f est le champ de vecteur, $I_k : R^n \to R^n$ décrit l'incrémentation des changements d'états aux instants appartenant à l'ensemble

$E = \{\tau_1, \tau_2, \dots\dots : \tau_1 < \tau_2 < \cdots\} \subset R^+$. E est donc l'ensemble des instants où les sauts se produisent. **[4]**

Exemple 3 : (Balle rebondissante)

On considère une balle de masse m soumise à l'action de la gravité. On la laisse tomber d'une altitude y_0 est une vitesse initiale nulle.

L'altitude y(t) de la balle suit donc l'équation différentielle issue de la mécanique classique $\ddot{y}(t) = -mg$

Nous considérons que les comportements impulsionnels liés au rebondissement de la balle sur le sol obéissent à une loi de restitution :

$$\frac{dy}{dt}(\tau_i^+) = \gamma \frac{dy}{dt}(\tau_i^-), \quad i = 1,2,3 \tag{I.8}$$

Avec : $\gamma < 0$ étant le coefficient de restitution de la vitesse (perte d'une fraction d'énergie à chaque rebond) et τi les instants d'impacts.

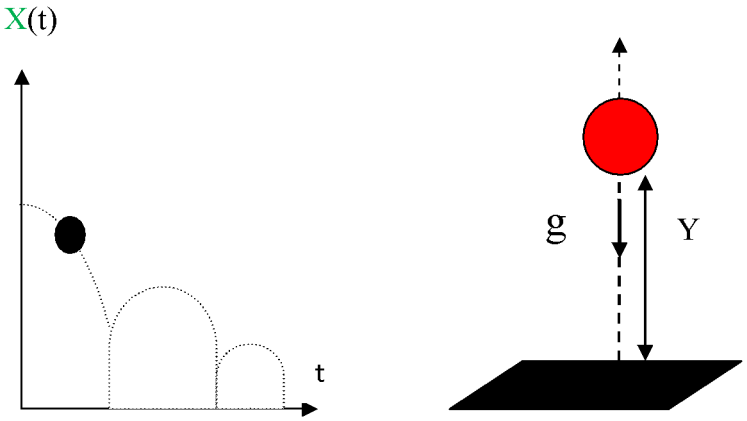


Figure (I.4) : Balle rebondissante soumise à l'action de la gravité.

I.5 Classification des systèmes hybrides proposée par M.S. Branicky

De manière générale, un système hybride à une dynamique continue peut être représenté par une équation différentielle qui dépend de certains phénomènes discrets comme la suivante:

$$\dot{x}(t) = f(x,t) \qquad t \geq 0 \tag{I.9}$$

Tel que :

$x(t)$: représente la composante continue

$f(t)$: est un champ de vecteur dépendant généralement de $x(t)$

τ : est l'instant où intervient une action discrète dont nous verrons ultérieurement comment elle peut être déclenchée. L'ensemble, action et déclenchement, est appelé phénomène hybride. **[4], [3]**

I.5.1 Interactions entre les systèmes : phénomènes hybrides

Nous allons maintenant étudier comment se produisent les interactions entre les deux sous systèmes (discret et continu) qui forment le SDH. Pour cela, nous allons d'abord présenter les actions discrètes qui peuvent intervenir lors de l'évolution d'un système continu décrit par un système différentiel de la forme précédente. **[K.TAARIT, 2010], [L.GHOMRI, 2012]** Branicky a proposé une classification en fonction des phénomènes discrets comme le montre la **Figure (I.5)** suivante :

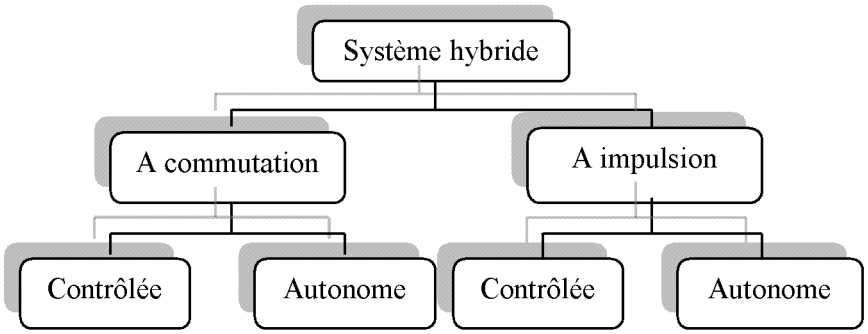


Figure (I.5) : Classification des systèmes hybrides

Dans la suite, nous allons présenter brièvement ces quatre phénomènes physiques et les caractéristiques de chaque catégorie par un exemple illustratif.

I.5.2 Action des phénomènes hybrides

Les actions des phénomènes hybrides sur le système continu sont de deux types :

- Les phénomènes hybrides agissent sur la dynamique du système continu, modifiant ainsi cette dynamique. Le système hybride se situe alors pour $t \geq \tau$ dans un autre mode de fonctionnement. On appelle ce phénomène « commutation de modèle dynamique » et τ instant de commutation.

  Le choix du nouveau modèle est déterminé par le SED. Un exemple simple de modèle formel avec deux modes de fonctionnement est donné comme suit :

$$\dot{x} = f_1(x,t) \qquad pour \ \ t < \tau$$

$$\dot{x} = f_2(x,t) \qquad pour \quad t \geq \tau \tag{I.10}$$

- Les phénomènes hybrides agissent également sur le vecteur d'état du système, le faisant évoluer de manière différente pour $t = \tau$. C'est à dire qu'à l'instant τ, l'état saute de $x(\tau^-)$ à $x(\tau^+)$ sans changement de modèle $x(\tau^-) \neq x(\tau^+)$. On appelle ce phénomène saut de l'état. Un exemple de modèle formel est représenté par les équations suivantes :

$$\begin{aligned} &\dot{x} = f_1(x,t) \quad pour \quad t \geq 0 \ et \ t \neq \tau \\ &\dot{x} = f_2(x,t) \quad pour \quad x(\tau^-) + g \quad pour \ g \neq 0 \end{aligned} \tag{I.11}$$

- Les deux actions peuvent être couplées ; en effet on peut considérer qu'à l'instant τ, on a une commutation de modèle et un saut du vecteur d'état. Un exemple de modèle formel est le suivant :

$$\begin{aligned} &\dot{x} = f_1(x,t) \qquad pour \ t < \tau \\ &x(\tau^+) = x(\tau^-) + g \quad pour \ g \neq 0 \\ &\dot{x} = f_2(x,t) \qquad pour \quad t > \tau \end{aligned} \tag{I.12}$$

L'état continu du système est alors réinitialisé de $x(\tau^-)$ à $x(\tau^+)$. Notons que $\tau^- et \ \tau^+$ correspondant respectivement aux limites à gauche et à droite deτ, La fonction g provoque une discontinuité de l'état à l'instant τ.

I.5.3 Déclenchement des phénomènes hybrides

L'action du phénomène hybride, comme le passage d'un mode de fonctionnement à un autre, est déclenchée par un événement. Ces événements sont alors classés en deux types :

- Les événements peuvent être déclenchés lorsque le vecteur d'état continu atteint certaines valeurs, provoquant ainsi un phénomène hybride dit autonome.
- Les événements déclenchés par une commande discrète externe, provoquant un phénomène, est dit contrôlé.

I.5.4 Classifications des phénomènes

Solon leur action et la manière dont ils sont déclenchés, M.Branicky a proposé une classification des phénomènes hybrides, et qui seront données ci-dessous.

I.5.4.1 Une commutation autonome

Une commutation autonome est caractérisée par un changement discontinu du champ de vecteur $f(t)$ quand l'état atteint certains seuils.

Exemple 4 : [4]

La fonction d'hystérésis $H(x)$ est représentée par la **Figure (I.6).** Quand la valeur de x atteint le seuil $-h/2$ ou $+h/2$, le champ de vecteur est commuté de façon discontinue .Entre les commutations, la dynamique continue est modélisée par $\dot{x} = a$ ou par $\dot{x} = -a$

$$\begin{aligned} \dot{x}(t) &= a \quad si\ x(t) = -h/2 \\ \dot{x}(t) &= -a \quad si\ x(t) = +h/2 \end{aligned} \tag{I.13}$$

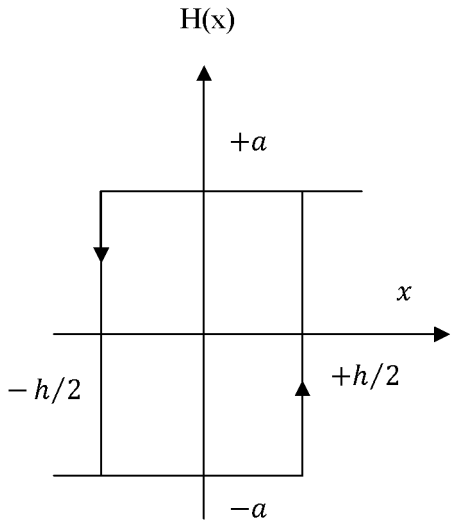


Figure (I.6).a : Fonction d'hystérésis

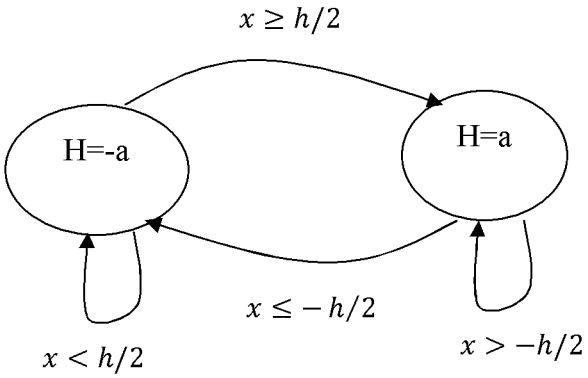


Figure (I.6).b : Automate hybride

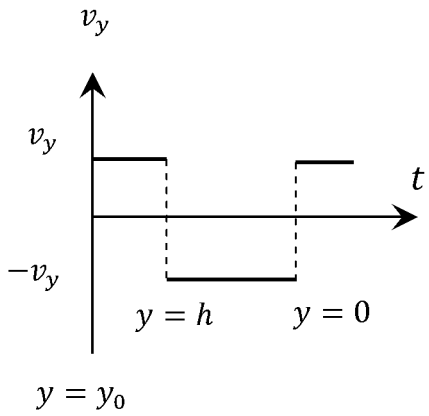
I.5.4.2 Sauts autonomes

Lorsque l'état atteint certaines zones prédéfinies de l'espace d'état, il effectue un saut (discontinuité ou impulsion) de sa valeur courante à une autre.

Un exemple très connu de ce phénomène est le choc entre deux corps où la vitesse change brutalement et subit un saut.

Exemple 5.4.2

Considérons une table de billard de longueur *l* et de largeur *h*, avec une boule, comme l'illustrer la **Figure (I.7).b** La position initiale de la boule est (x_0, y_0) et après avoir été frappée elle commence à se déplacer avec une vitesse *v* (v_x, v_y). 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. La **Figure (I.7).a** représente la composante verticale v_y de la vitesse *v en* fonction de *y*. **[L.GHOMRI, 2012]**



Figure(I.7).a : Evolution de vitesse v_y de la balle de billard

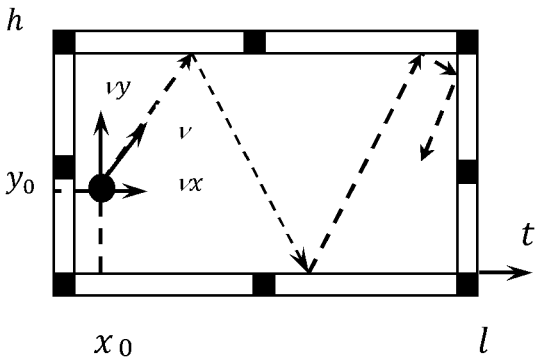


Figure (I.7).b : La trajectoire d'une balle de billard.

I.5.4.3 Commutations contrôlées (commandées)

Dans ce cas, le champ de vecteur $f(t)$ commute en réponse à une loi de commande (action d'une variable externe).

Exemple 5.4.3

Un modèle simplifié d'une transmission manuelle :

$$dx_1/dt = x_2$$
$$dx_2/dt = \left[-a\left(\frac{x_2}{v}\right) + u\right]/(1+v) \qquad (I.14)$$

Où :

x_1 : la vitesse relative par rapport à un point fixe

x_2 : la vitesse de rotation de l'engin

v : la position du levier de vitesse $v \in \{1, 2, 3, 4\}$

$u \in \{0, 1\}$: la position d'accélération

a : un paramètre de système, le champ de vecteurs est commuté par la valeur de v .

I.5.4.4 Sauts contrôlés

L'état ou une partie change instantanément sous l'influence d'une action extérieure, la valeur de l'état change impulsivement en réponse à une commande (de façon discontinue).

[K.TAARIT, 2010], [L.GHOMRI, 2012]

Exemple 5.4.4

Considérons un modèle simple d'un stock où on dépose les quantités $\alpha_1, \alpha_2, \alpha_3 \dots etc$ de matière aux instants $t_1 < t_2 < t_3 \dots$ L'évolution de la matière est régie par l'équation suivante :

$$\dot{y} = -Q(t) + \sum_i \delta(t - t_i)\alpha_i \qquad (I.15)$$

Avec: y le niveau de la matière dans le stock, Q la fonction ou d'utilisation de la matière et δ l'impulsion de Dirac.

I.5.5 Exemple de phénomène hybride (Une boite de vitesse automatique)

Une boite de vitesse automatique est un exemple de système complexe où sont présents les différents déclenchements et actions des phénomènes hybrides présentés ci-dessus.

On considère, pour simplifier, que ce système possède comme variables discrètes trois vitesses v_1, v_2, v_3 et deux modes de conduite :

Ville avec les vitesses autorisées v_1 , v_2 et Route où toutes les vitesses sont disponibles.

Le nombre de tours par minute du moteur noté $N(t)$ est la variable continue du système dont la dynamique est représentée par une équation différentielle.

Le passage d'une vitesse à une autre, lorsque la variable continue $N(t)$ atteint certaines valeurs, correspond à un phénomène hybride déclenché de manière autonome. Tandis que le passage d'un mode de conduite à un autre est décidé par le conducteur, cela correspond à un phénomène hybride contrôlé.

Ces deux phénomènes agissent sur la variable continue $N(t)$. En effet lorsqu'on change de vitesse, le nombre de tours par minute du moteur est réinitialisé, on a dans ce cas un saut de l'état. De plus, à chaque vitesse correspond une dynamique différente, on a donc également une commutation de modèle.

I.6 Représentation formelle des systèmes hybrides :

Les systèmes hybrides sont des systèmes dynamiques faisant intervenir explicitement et simultanément des phénomènes ou des modèles de type dynamique continu et événementiel. Donc la description de SDH nécessite l'utilisation de variable continues et de variables discrètes. Alors il s'avère utile de rappeler la définition des systèmes dynamiques continus (SDC) et celle de système à événement discret (SED) avant de donner la représentation d'un SDH. **[B.CEBRON, 2000], [A.GORARD, 2004]**

I.6.1 Système dynamique continue

Un système à temps continu est caractérisé par une évolution continue de ses variables ou états en fonction du temps. Ces variable peuvent prendre une infinité de valeurs c'est-à-dire $X \subset R^n$ où :

X : L'ensemble est l'espace état,

R : L'ensemble des réels,

n : est le nombre de variables d'état, également désigné sous le nom de l'ordre de système

Définition

Un système dynamique, d'état continu et temps continu est décrit par un triple (X, U, F) où :

$X \subset R^n$ est l'espace d'état

$U_c \subset R^m$ est l'ensemble des commandes continues,

$F: X \times U_c \to R^n$ est le champ de vecteur.

Formellement, un système dynamique, continu est régi par les équations différentielles suivantes :

$$\begin{cases} \dot{x}(t) = F\big(x(t), u_c(t)\big), t) & (I.16.a) \\ y(t) = g(x(t), u_c t), t)) & (I.16.b) \\ x(t_0) = x_0 \end{cases} \qquad (I.16)$$

(I.16. a) : Equation d'état différentielle.

(I.16. b) : Equation de sortie (d'observation).

I.6.2 Systèmes dynamiques discrets

Un système physique est de dynamique discrète si l'information véhiculée par les grandeurs d'entrée et de sortie est disponible à des instants discrets.

Formellement, un système dynamique discret régi par les équations de récurrences suivantes :

$$\begin{cases} x(k+1) = F\big(x(k), u_d(k)\big), k) & (I.17.a) \\ y(k) = G(\big(x(k), u_k(k)\big), k)) & (I.17.b) \end{cases} \quad \text{(I.17)}$$

- (I.17. a) : équation aux différences.
- (I.17. b) : équation de sortie.
- k : Entier positif ou nul désignant l'instant discret.

I.6.3 Système à événement discret

Un système à événement discret est un système dynamique défini par un espace d'états discrets et des évolutions, nommées trajectoires, basées sur une succession des états et des transitions.

Les transitions sont étiquetées par des symboles, appelés événement, définis avec les éléments d'un alphabet.

Une approche courante pour l'étude de ces systèmes consiste à ignorer la valeur explicite de temps et à s'intéresser uniquement à l'ordre d'occurrence des événements. En général les systèmes à événement discret(SED) sont représentés par des automates à états fini qui seront définis ultérieurement.

I.7 Systèmes Dynamiques hybrides

I.7.1. Définition

Un système hybride est un septuple $H = (Q, X, \Sigma, U_c, F, G, R)$, avec :

1. $Q = \{q_1, q_2, \dots \dots\}$ est un ensemble fini, appelé l'ensemble des états discrets.
2. $X \subseteq R^n$ est l'ensemble des états continus.
3. Σ est ensemble d'événement ou de symboles d'entrées discrètes.
4. U_c est l'ensemble des commandes continues.
5. F est un champ de vecteur décrivant la dynamique continu.
6. G décrit la dynamique discrète.

7. $R = \{R_e, e \in \Sigma\}$ est la collection des fonctions réinitialisation.

Soit $S = X \times Q$ et soit P un sous-ensemble fermé de $X \times Q$. On définit l'état hybride $s(t) \in S$ du système à l'instant t par la donnée du couple $[x(t), q(t)]$.

Soit :

- $\boldsymbol{x(t) \in X \subset R^n}$
- $\boldsymbol{q(t) \in Q \subset \mathbf{N}^m}$
- $\boldsymbol{u_c(t) \in U_c \subset R^r}$
- $u_d(t) \in U_d \subset N^p$

Où $t \in R^+$, et :

- X : Ensemble des états continus
- Q : Ensemble des états discrets
- U_c : Ensemble des commandes continues
- U_d : Ensemble des commandes discrètes
- t : variable temps

Alors l'évolution d'état (x, q) du système peut être décrite par les équations suivantes :

$$\dot{x}(t) = F\big(x(t), q(t), u_c(t)\big); \qquad \forall s(t) \in S \cap CP \tag{I.18}$$

$$(x^+, q^+) = G(x(t), q(t), u_d(t), \sigma); \quad \begin{cases} si\ s(t) \in \partial P \\ si\ u_d(t) \in U_d\ intervient \end{cases} \tag{I.19}$$

$$x(t_0) = x_0\ ; \tag{I.20}$$

$$q(t_0) = q_0\ ; \tag{I.21}$$

Avec CP c'est le complément de P.

$(.)^+$ se rapporte aux variables après une transition, x_0 et q_0 sont respectivement les conditions initiales des états continus et discrets.

La fonction F représente la dynamique continue du système hybride et G représente la dynamique hybride du système correspondant aux phénomènes hybrides décrits précédemment.

Le sous-ensemble P est celui des zones de déclenchement des phénomènes hybrides et ∂P est la frontière de P. La commande discrète $u_d(t)$ détermine les instants où la fonction G

intervient (instants de commutation/saut). Elle détermine également le nouveau modèle ou l'amplitude du saut de l'état suivant l'action du phénomène hybride.

I.7.2 Trajectoire d'état hybride

La trajectoire hybride est l'évolution des états hybride (x(t), q(t)) dans l'ensemble de temps R^+. Nous disons qu'un système est hybride dès que son état à un instant particulier t peut être décrit par un nombre fini de valeurs réelles, comme précédemment, et additionnellement par une valeur discrète, les temps de commutation sont représentés par τ_k. k prend ses valeurs dans l'ensemble des entiers positifs $N = \{0,1,2,\dots\}$ avec $\tau_k \leq \tau_{k+1}$.

A chaque instant τ_k, l'état discret commute à une autre valeur, cependant, entre deux moments consécutifs de commutation, c'est-à-dire $t \in [\tau_k, \tau_{k+1}]$ la valeur de q(t) demeure constante.

Notation

- G_{CA} : Les Commutations de modèle Autonomes.
- G_{CC} : Les Commutations de modèle Contrôlées.
- G_{SA} : Les Sauts (les impulsions) de l'état Autonomes.
- G_{SC} : Les Sauts de l'état Contrôlés.

La Figure (I.8) illustre une trajectoire d'état hybride où :

$$S_{q0} = (X, q_0) \subset S, S_{q1} = (X, q_1) \subset S, S_{q2} = (X, q_2) \subset S \; avec \; Q = \{q_0, q_1, q_2\}.$$

Une évolution hybride peut être brièvement décrite comme suit :

A l'instant initial $\tau_0, s(\tau_0) = [x_0, q_0] \in S$, le système évolue suivant la dynamique F jusqu'à ce que « s » atteigne l'ensemble P . A cet instant noté τ_1, la dynamique hybride intervient et modifie la valeur de q qui devient égale alors à q_1. Ce phénomène est donc une commutation autonome (G_{CA}).

Ensuite, le système poursuit son évolution avec une nouvelle dynamique définie par F (q a été modifié)

A l'instant τ_2, S atteint à nouveau l'ensemble P, mais cette fois la dynamique hybride modifie x. Ce phénomène est donc un saut autonome(G_{SA}).

Enfin, ces deux phénomènes se reproduisent mais sur l'intervention d'une commande discrète $U_d(t)$ qui détermine dans un cas le nouveau modèle (G_{CC}) et dans l'autre cas la nouvelle valeur du vecteur d'état (G_{SC}). **[3], [7]**

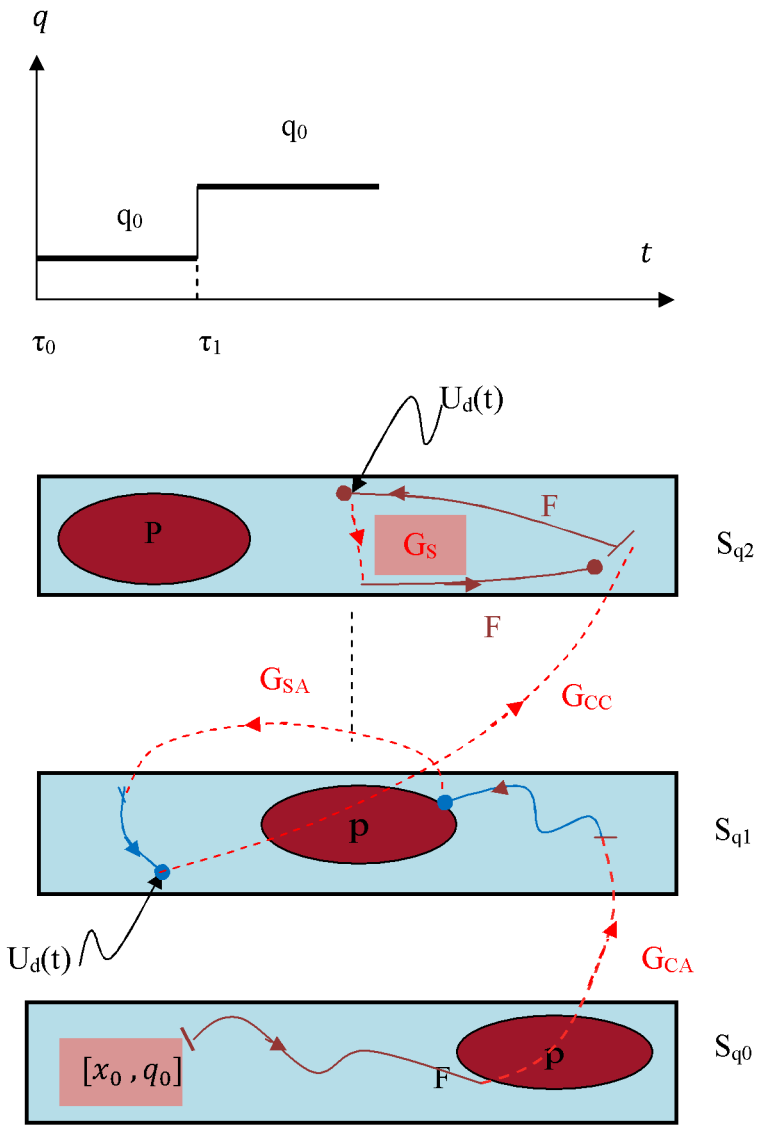


Figure (I.8) : Trajectoire hybride

I.8 Conclusion

Ce premier chapitre nous a permis d'introduire la théorie des systèmes dynamiques hybrides et de mieux les situer par rapport aux systèmes dynamique continus et aux systèmes à événements discrets, grâce notamment à une analyse des phénomènes hybrides.

Vu la complexité croissante des systèmes hybrides, plusieurs outils de modélisation ont été mis en œuvre et ne cessent d'être développées. Dans le chapitre qui suit nous allons voir les outils plus souvent utilisés tels que les **Réseaux de Petri Hybrides, les automates hybrides.**

II.1 Introduction

Dans ce chapitre nous présentons les deux outils utilisés pour la modélisation des SDH grâce, notamment, à leur aspect graphique. Il s'agit des automates hybrides, et des réseaux de Petri hybrides

Nous présentons d'abord les automates à état fini et les automates temporisé ce qui nous permet de donner une représentation générale des automates hybrides et les différentes types de modèle pour sa représentation. Par la suite, nous donnons les notions de base des réseaux de Petri continus et discrets et temporels, cela a conduit à introduire des réseaux de Petri hybrides qui sont utilisés pour modéliser les SDH, principalement dans le but de simuler et d'analyser ces systèmes complexes. Nous allons voir aussi quelques méthodes de commandes des systèmes dynamiques hybrides.

II.2 Modélisation par les 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 par définir les automates à états finis en premier lieu, car c'est à partir de ce formalisme que les automates hybrides ont été définis.

II.2.1 Les automates à états finis

La technique la plus ancienne et la plus classique pour représenter le comportement dynamique d'un système à évènements discrets est d'utiliser un automate fini (on dit aussi machine à états finis), est formée d'un ensemble fini d'états et d'une fonction définissant les transitions entre états (ce sont les événements). Un automate à états finis peut être décrit par le quadruplet (Q, Σ, A, q_0) et noté AEF. **[12]**

$$AEF = (Q, \Sigma, A, q_0) \text{ avec :}$$

- Q : est un ensemble fini de sommets représentant les états discrets ;
- Σ : est un ensemble fini de symboles (évènement) alphabet ;

- A : est un l'ensemble des transitions entre états. Une transition est un triplet (sommet source, événement, sommet but). Une transition traduit le passage du système d'un état à un autre, suite à l'occurrence d'un évènement appartenant à Σ ;
- q_0 : est l'état initial ;

Exemple II.1

Un objet se déplace de manière spontanée (il agit sans intervention extérieure) à l'intérieur d'un labyrinthe (Figure II.1.a)

Les salles $\mathbf{S_i}$ communiquent par des portes unidirectionnelles $\mathbf{p_1}$ et $\mathbf{p_3}$ et bidirectionnelle $\mathbf{p_2}$.

On note par « $\mathbf{p_i}$ » l'évènement : « l'objet passe par la porte $\mathbf{p_i}$ ».

Initialement, l'objet se trouve dans la salle $\mathbf{S_0}$.

On conçoit que les situations (ou comportements) possibles sont :

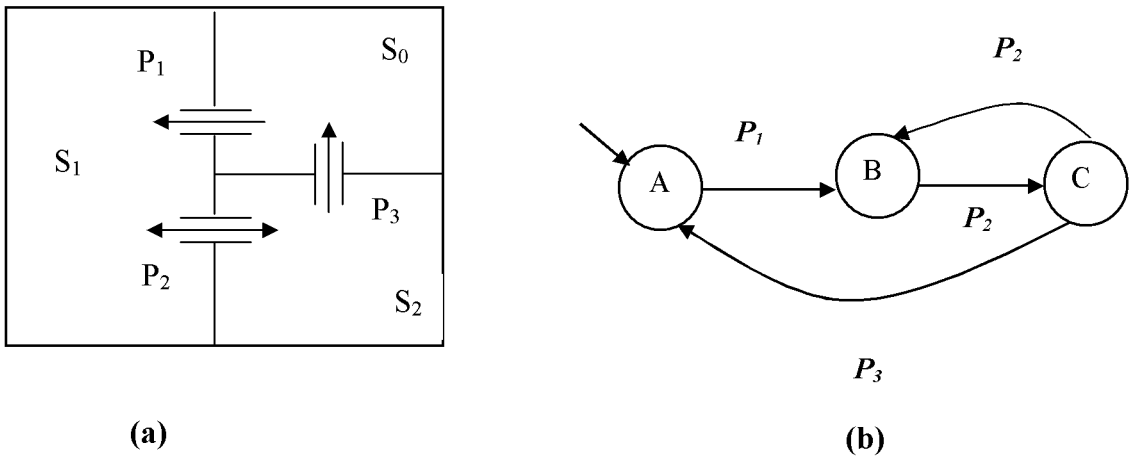
L'objet est dans la salle $\mathbf{S_0}$

L'objet est dans la salle $\mathbf{S_1}$

L'objet est dans la salle $\mathbf{S_3}$

Ce qui définit trois états différents, relatifs aux possibilités d'occupation des salles.

Le passage de l'état « l'objet en salle $\mathbf{S_0}$ » (état A) à « l'objet en salle $\mathbf{S_1}$ » (état B) a lieu sur l'occurrence de l'évènement $\mathbf{P_1}$. On peut alors construire l'automate à état fini (Figure II.1.b)



(a) **(b)**

Figure (II.1) : (a) Labyrinthe, (b) Automate à états fini

On a alors :

- $Q = \{A, B, C\}$
- $\Sigma = \{P_1, P_2, P_3\}$
- $A = \{(A, P_1, B), (B, P_2, C), (C, P_2, B), (C, P_3, A)\}$
- $q_0 = A$

II.3 Les automates temporisés

Les automates temporisés sont des automates à états (discrèts) finis étendus par un ensemble des variables réelles, appelées horloge, dont les valeurs croissent uniformément avec le passage du temps et qui peuvent être remises à zéro.

Ainsi, une horloge « i » est définie par une variable réelle $\boldsymbol{x_i}$ tel que $\boldsymbol{dx_i/dt = 1}$. Des contraintes portant sur ces horloges sont associées aux sommets et aux transitions. Ces contraintes temporelles définissent respectivement le temps pendant lequel le système peut rester dans un sommet et la possibilité de franchissement d'une transition (gardes).

Un automate temporisé est défini par un quintuple $(\boldsymbol{Q, q_0, \Sigma, Y, A, Inv})$, avec :

- Q : un ensemble fini de sommets dont le sommet initial q_0.
- Σ : un ensemble d'évènements.
- Y : un ensemble fini d'horloge définis par des variables réels prenant leur valeur dans R^+.
- A : l'ensemble des arcs (transitions discrètes) de l'automate temporisé. Un arc est défini par un quintuple de la forme $(\boldsymbol{q, \Psi, \sigma, R, q'})$. Les sommets q et q' sont des sommets source et état respectivement de l'arc. Ψ est la condition (garde) que doivent satisfaire les valeurs des horloges de l'automate temporisé pour pouvoir franchir l'arc. $\sigma \in \Sigma$ est l'évènement (l'étiquette associée à la transition) qui peut être reçu ou généré lors du franchissement, $R \subset Y$ défini l'ensemble des horloges à remettre à zéro lorsque l'arc est franchi.
- $Inv : Q \rightarrow \Psi(Y)$ associe à chaque sommet une contrainte temporelle (portant sur les valeurs des horloges) appelée invariant. Tant que l'invariant d'un sommet des horloges est vrai, il est possible de rester dans ce sommet. **[12]**

Exemple II.3

Considérons une machine simple à trois états : arrêt (a), marche (m) et panne (p). L'état initial « a » est désigné par la flèche entrante, et il y a quatre transitions associées. Cette machine est modélisée par l'automate temporisé comme le montre la figure suivante :

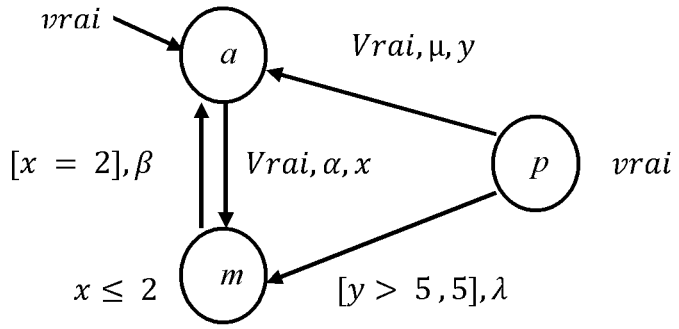


Figure (II.2): Automate temporisé correspondant à une machine simple.

$Q = \{a, m, p\}$

$q_0 = a$

$\Sigma = \{\alpha, \beta, \lambda, \mu\}$

$X = \{x, y\}$

$A = \{(a, vrai, \alpha, x, m), (m, x = 2, \beta, a), (m, y > 5.5, \lambda, p), (p, vrai, \mu, y, a)\}$

$Inv(a) = vrai, Inv(m) = (x \leq 2), Inv(p) = vrai$

Deux horloges sont utilisées dans le modèle temporisé pour spécifier que la durée de l'état de marche sur cette machine est de deux unités du temps (horloge « x ») et qu'une panne ne peut avoir lieu qu'à partir de 5,5 unités du temps après la dernière action de réparation (horloge «y»). Ainsi, les trois éléments suivants permettent à l'horloge «x » de forcer le retour au sommet « a » après un séjour d'exactement 2 unités du temps dans le sommet « m » si une panne ne survient pas durant cet intervalle :

i) remise à zéro de « x » par la transition entrante dans le sommet « m »

ii) impossibilité de rester au-delà de 2 unités du temps « m » à cause de son invariant ($x \leq 2$)

iii) autorisation de l'exécution de la transition de « m » vers « a » uniquement lorsque $x = 2$.

Quant à l'horloge « y », elle est remise à zéro par l'action de réparation (la transition partant du sommet « p ») pour autoriser ultérieurement l'occurrence d'une panne (la transition entrant dans le sommet «p ») après un séjour d'au moins 5.5 unités du temps dans les sommets « a » et « m».

II.4 Les automates hybrides [11]

Un automate hybride consiste en un ensemble de places et d'arcs entre chaque place. Une place représente un mode de fonctionnement et un arc décrit une transition d'un mode de fonctionnement à un autre. A chaque place est associée une inclusion différentielle, et chaque arc est étiqueté par une condition de transition et une relation de saut ou de réinitialisation. A chaque instant, l'état de l'automate hybride est donné par la paire (l, x) où l est la place, x est l'état continu et (l_0,x_0) est l'état initial. La trajectoire de l'automate évolue d'une part en fonction des changements instantanés de places, d'autre part en fonction de l'état continu évoluant suivant l'inclusion différentielle dans chaque place.

II.4.1 Définition

Un automate hybride est défini par la donnée de H = (L, X, E, I, F) où :

- L : est un ensemble fini de sommets appelés places.
- $X \subset R^n$: est l'espace d'état continu de l'automate. Un état est une paire (l, x) composé d'une place et d'un vecteur x $\in X$.
- E : est un ensemble de transitions, une transition étant représentée par un arc entre deux places.
- I : est une collection d'ensembles appelés invariants de la place.
- F : est un ensemble d'applications associant à chaque vecteur d'état un sous ensemble de l'espace d'état, noté $f_L(x)$. Une application est associée à chaque place. La dynamique continue est définie par une inclusion différentielle $\dot{x} \in f_l(x)$, où $f_l(x)$ est appelé l'activité de la place L.

Exemple II.4.1

Cet exemple s'agit de la description d'une fontaine décorative à deux bassins avec :

— écoulement direct entre le bassin 1 (supérieur) et le bassin 2 (inférieur), suivant deux modalités dépendant de l'état du système lui-même : fuite le long de la colonne centrale pendant le remplissage du bassin 1, débordement décoratif quand ce bassin est plein ;

— relevage permanent par pompe, à débit constant, entre le bassin 2 et le bassin 1, avec également un effet décoratif.

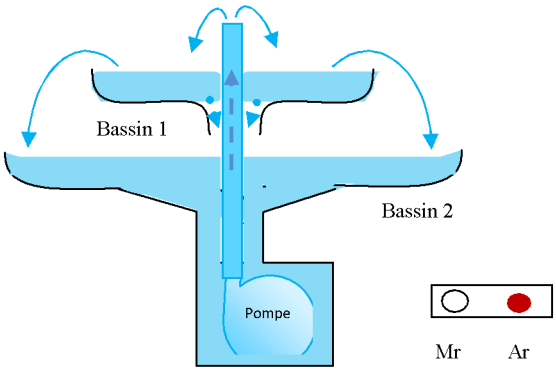


Figure (II.3) : Fontaine à deux bassins

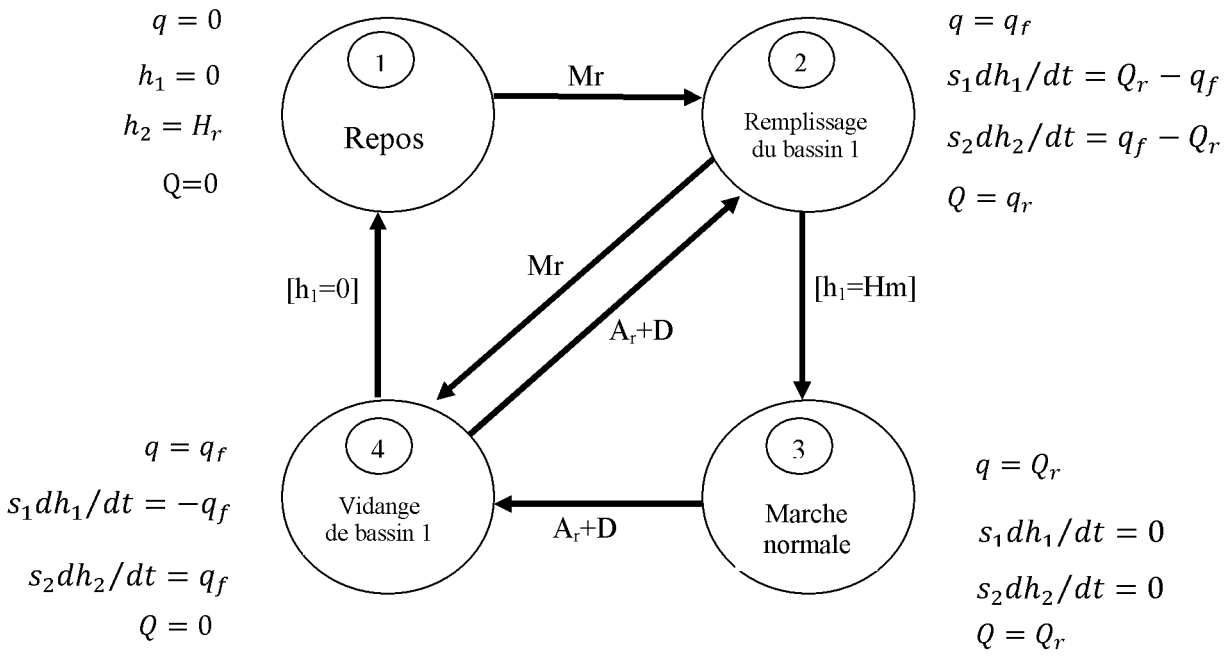


Figure (II.4) : Fonctionnement d'une fontaine par automate hybride

h_1 : niveau dans le bassin 1	Q : débit de 2 vers 1
H_m : niveau max dans le bassin 1	q_f : débit de fuite de 1 vers 2
S_1 : section moyenne du bassin 1	Q_r : débit nominal de la pompe
H_2 : niveau dans le bassin 1	Mr: mise en marche
H_r : niveau max dans le bassin 2	Ar : arrêt
S_1 : section moyenne du bassin 2	D : détection d'un défaut
q : débit de 1 vers 2 (global)	Entre crochets : les conditions (vraies ou fausses)

La pompe peut être mise en marche et arrêtée volontairement, ou tomber en défaut. Le fonctionnement normal de cette installation suppose que, le bassin 1 soit vide et le bassin 2 à son niveau maximal lorsque l'on met en marche initialement la pompe. On admet pour établir la partie continue du modèle que les variations des sections des bassins peuvent être négligées. L'état d'arrêt complet (q=0, Q=0, h1=0) n'est pas strictement nécessaire du point de vue de la commande, mais il permet une description plus cohérente de l'installation en différenciant l'état de repos (1) de l'état de vidange lente après arrêt de la pompe (4). On notera aussi qu'il y a des événements commandés (Ar), d'autres non (D).

II.4.2 Exemples de modélisation par automate hybride

Exemple II.4.3.1

Nous allons procéder à la modélisation d'un thermostat sous la forme d'un automate hybride figure (II.5.b). Le rôle du thermostat est de contrô1er la température d'une chambre, cela revient à tester instantanément la valeur de la température et à mettre le radiateur à l'état « *on* » ou « *off* » figure (II.5.a).

La variable x modélise la température du système. Les deux variables discrètes $q_1 = 1(on)$ et $q_2 = 0(off)$.

En q_1, x suit une équation différentielle de la forme :

$$\dot{x} = -x + 4 \qquad \text{(II.1)}$$

Et en q_2

$$\dot{x} = -x \qquad \text{(II.2)}$$

L'objectif de contrôle est de maintenir la température entre 1 et 2.

Le système ainsi décrit peut être représenté comme suit :

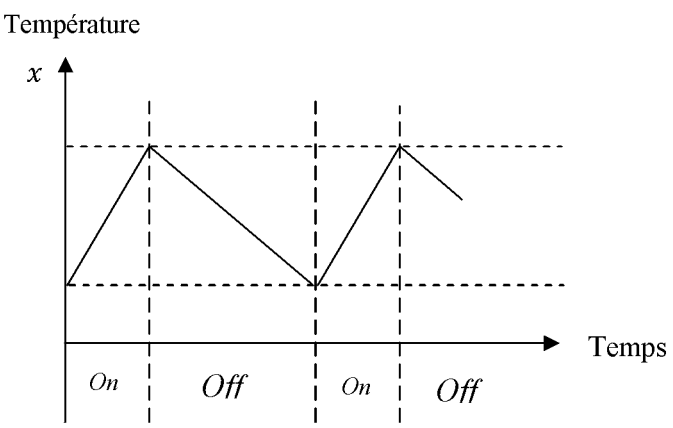


Figure (II.5.a) : Évolution de la température d'une chambre

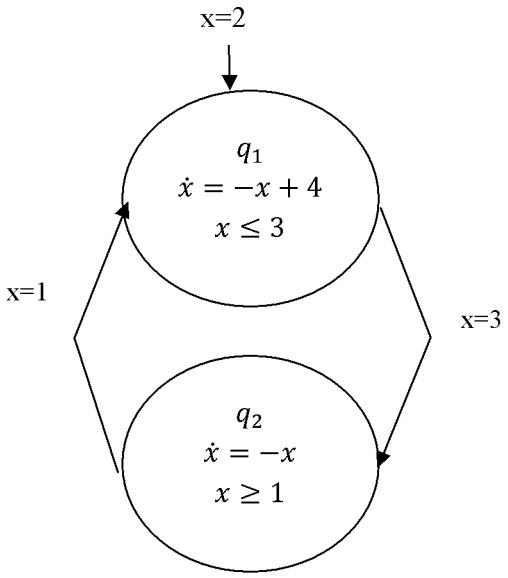


Figure (II.5.b) : l'automate hybride du thermostat

Dans ce cas :

- X : est l'ensemble des variables d'état continues.
- $V = \{q_1, q_2\}$
- $g(q_1) = (\dot{x} = -x + 4) \wedge (x \leq 3)$
- $g(q_2) = (\dot{x} = -x) \wedge (x \geq 1)$
- On a deux commutations possibles : $E = \{e_1, e_2\}$ avec $e_1 = (q_1, q_2)$ et $e_2 = (q_1, q_2)$
- $Saut(e_1) = (x = 3)$
- $Saut(e_2) = (x = 1)$
- $\Sigma = (3,1)$ $avec$ $Evénement(e_1) = 3$ et $Evénement\ (e_2) = 1$
- $Initial(q_1) = (x = 2)$ et $Initial(q_2) = \emptyset$
- $Final(q_1) = \emptyset$ et $Final(q_2) = \emptyset$

II.4.3 Types de représentation de l'automate hybride [12]

Les automates hybrides peuvent être représentés par les types de modèle suivants :

II.4.3.1 Modèle analytique

Le modèle suivant a été introduit par Alur, l'automate hybride est décrit par :

$$H =< X_D, X_C, \mu_1, \mu_2, \mu_3, I >$$

Où :

- X_D : est l'espace d'état discret.
- X_C : est l'espace d'état continu.
- μ_1 : est la fonction qui associe à chaque phase (état discret) q, une fonction continue f_q décrivant les dynamiques continues relatives à cette phase.
- μ_2 : est la fonction associée à chaque phase q, un ensemble invariant $\mu_2(q) \subseteq X_C$. si le système est dans l'état discret q, l'état continu doit alors être dans $\mu_2(q)$, sinon, une exception générée.
- μ_3 : est la fonction associée à chaque paire de phases (q, q'), une relation de transition $\mu_3(q, q') \subseteq (X_d \times X_c)$. $\mu_3(q, q')$ est une application de l'espace d'état de q vers celui de q'.
- I : est l'ensemble des conditions initiales.

L'état global du système hybride est représenté par une phase discrète et un état continu. L'état peut changer, soit instantanément en présence d'événement discret et ce en suivant les relations de transition données par μ_3 soit d'une façon continue suivant des fonctions non définies par μ_1. L'évolution d'un système dynamique hybride dans $T = [t_i, t_f]$ consiste en une collection (τ, q, x) qui satisfait les propriétés suivantes :

- Condition initiale : $(q(\tau_0), x(\tau_0)) \in I$.
- Evaluation discrete : $(q(\tau_i), x(\tau_i), q(\tau_{i+1}), x(\tau_{i+1})) \in \mu_3(q(\tau_i), q(\tau_{i+1})), \forall i$.
- Evaluation continue: $\forall i, \tau_i \tau_{i+1}$ et $\forall [\tau_i, \tau_{i+1}[$.

$$\begin{cases} \dot{x}(t) = f_q(x(t)). \\ q(t) = (\tau_i). \\ x(t) \in \mu_2(q). \end{cases} \quad \text{(II.3)}$$

Exemple II.4.3.1

Considérons un système hybride à deux variables états continues x_1 et x_2 et possédant trois phases discrètes, comme le montre la figure suivante :

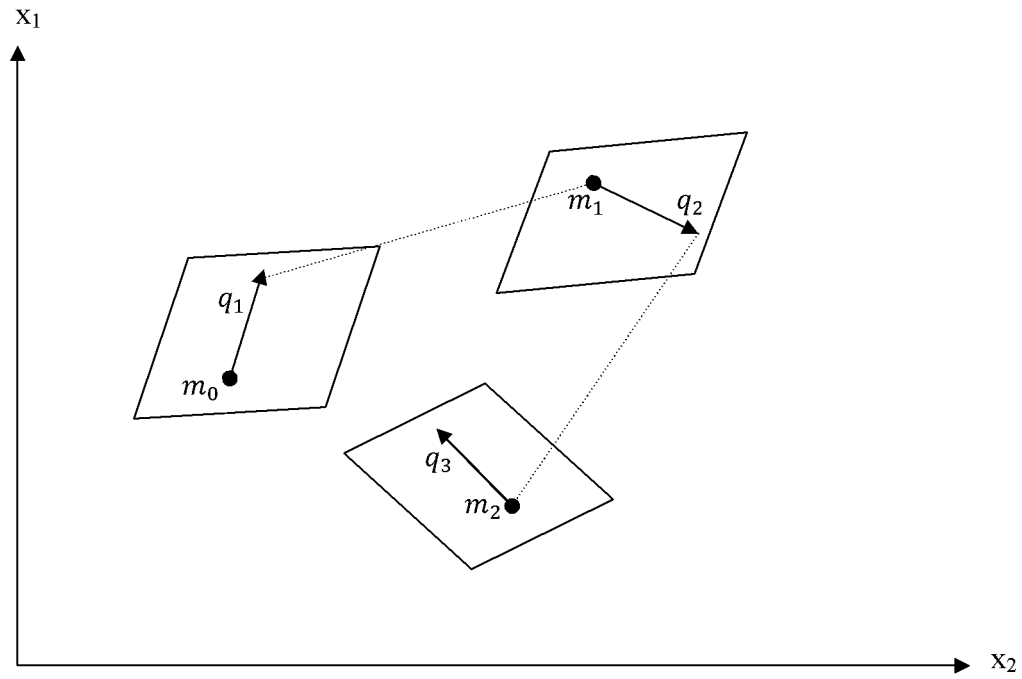


Figure (II.6) : Exemple d'un système hybride

Un tel système peut être modélisé par un automate hybride à trois états discrets (phase) q_1, q_2 *et* q_3, à chaque phase sera associée un ensemble invariant $\mu_2(q_i), i = 1,2,3,$ et un ensemble d'activités (fonctions) donné par $(q_i)\,, i = 1,2,3$. L'automate hybride fonctionne comme suit :

Au départ, il est en phase q_1, et l'état m_0 $(m_0 = (x_{10}, x_{20}))$, l'état continu change suivant la fonction $\mu_1(q_i)$. Après un certain temps, une transition se produit et l'état passe d'une façon discontinue, suivant une relation de transition donnée par $\mu_3(q_1, q_2)$, a une nouvelle phase q_2, et un nouvel état m_1, et ainsi de suite jusqu'à atteindre l'état m_2 dans la phase q_3.

Remarque II.4.3.1

1. Un automate hybride est dit déterministe si pour un état donné (q, x), plusieurs destinations (q', x') sont possibles, et ceci peuvent provoquer des conflits. Certains auteurs ont proposé quelques méthodes d'analyse relatives à des cas particuliers d'automates hybrides non déterministes pour lesquels ces conflits apparaissent.
2. L'automate hybride est dit non bloquant si pour tout $(q, x) \in (X_D \times X_C)$ le système continu suit son évolution par un saut discret, soit en suivant une fonction continue, il n'a donc aucun point de blocage.
3. Il faut noter que nous pouvons modéliser les systèmes purement continus par un automate hybride à un seul état discret. Prenons par exemple, le système régit par les équations suivantes :

$$\begin{cases} \dot{x} = f(x,t) \\ x(0) = x_0 \\ x \in R^n \end{cases} \qquad \text{(II.4)}$$

Un tel système peut être modélisé par un automate hybride à une seule phase q à laquelle correspond $\mu_1(q) = f$, $\mu_2(q) = R^n$, $\mu_3(q,q) = (\{q\} \times R^n)$ et $I = \{x_0\}$.

II.4.3.2 Modèle graphique

Si l'état discret ne prend qu'un nombre fini de valeurs, il est intéressant de représenter l'automate hybride par un graphe. Nous pouvons associer un graphe à un automate hybride en usant des notions suivantes :

1. Les nœuds : le nombre de nœuds dans un graphe est équivalant au nombre de valeurs possibles de l'état discret.
2. L'évolution continue : pour chaque nœud q, nous associons une fonction f_q décrivant l'évolution des dynamiques continues. Elle est donnée par :

$$f_q : X \times T \to X_C$$
$$(q, x, t) \to f_q(x, t)$$

3. L'ensemble invariant d'un nœud : à chaque nœud q, nous correspondant un ensemble invariant défini par : $inv(q) = \{x \in X'_C \mid x \in \mu_2(q)\}$. Ceci implique qu'un système demeure en q si et seulement si $x \in inv(q)$.

4. Le garde transition : à toute transition $q \rightarrow q'$, nous associons un garde qui détermine les conditions de transition. Il est noté *En (q, q')*
5. Evénement de transition : à la transition $q \rightarrow q'$, nous correspondons l'ensemble des valeurs donné par :$Res(q,q') = \{x' \in X_C \mid x' \in I(q')\}$. Ceci implique qu'après un franchissement de transition émanant de (q, x) et atteignant l'état *(q', x'), x'* doit appartenir à l'ensemble des conditions initiales q'.

En utilisant toutes ces notions, le graphe de l'automate hybride aura la forme suivante :

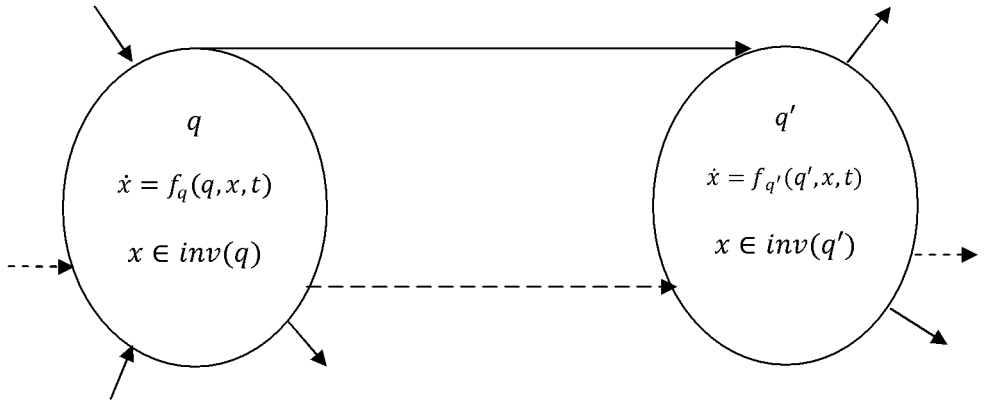


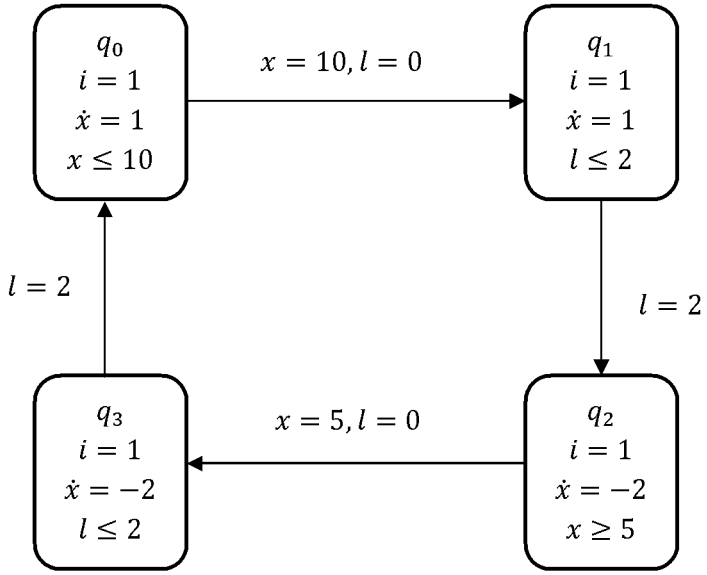
Figure (II.7) : exemple de graphe d'automate hybride

Exemple II.4.3.2

Considérons un réservoir d'eau dans lequel le niveau est détecté à tout instant par un capteur qui, à son tour le transmet à un contrôleur (moniteur) qui actionne une pompe à deux états « on/off ». Nous supposons que lorsque la pompe est en positon « off », le niveau diminue de deux unités par seconde, quand elle est en position « on », il augmente d'une unité par seconde. Supposons aussi qu'initialement le niveau d'eau vaut une unité, et la pompe est en position « on ». L'objectif du système de commande est de maintenir ce niveau entre une et douze unités.

Cependant, depuis l'instant où le moniteur signale une nécessité de changement d'état, de la pompe et celui dont il devient effectif, il existe un délai de deux secondes. Le moniteur doit alors signaler une mise en position « on » de la pompe avant d'atteindre le niveau un, et la position « off » avant d'atteindre douze unités.

La figure suivante est une représentation graphique de l'automate hybride correspondant au système décrit précédemment. L'automate a quatre états discrets notés $q_0,\ q_1, q_2\ et\ q_3$ respectivement ; L'horloge l est utilisée pour spécifier les délais. Lorsque le système est en $q_1 ou\ q_3$, le signal faisant passer la pompe de on à off où de off à on est envoyé l seconde avant.



Figure(II.8): l'automate hybride correspondant au moniteur du niveau d'eau

Dans ce cas :

$$inv(q_0) = \{x \in R / x \leq 10\},\ \ Res(q_0, q_1) = \{(0, 10)\}, En(q_0, q_1) = \{x = 10\}, \mu_1(q_0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$inv(q_1) = \{l \in R^+ / l \leq 2\},\ \ Res(q_1, q_2) = \{(2, x)\}, En(q_1, q_2) = \{l = 2\}, \mu_1(q_1) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$inv(q_2) = \{x \in R / x \geq 5\},\ \ Res(q_2, q_3) = \{(0, 5)\}, En(q_2, q_3) = \{x = 5\}, \mu_1(q_2) = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

$$inv(q_3) = \{l \in R^+ / l \leq 2\},\ \ Res(q_2, q_3) = \{(2, x)\}, En(q_2, q_3) = \{l = 2\}, \mu_1(q_3) = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

II.4.3.3 Interconnexion d'automate

Les systèmes complexes sont souvent modélisés par l'interconnexion de plusieurs automates hybrides. Chacun d'eux est vu comme un sous-système utilisé pour décrire une fonctionnalité particulaire du système global. Dans une interconnexion, les sorties des uns deviennent les entrées des autres. Il faut distinguer trois types d'entrées pour chaque sous-système :

- Les entrées locales : qui sont spécifiées localement par les interconnexions des sorties
- Des différents automates hybrides (sous-système) constituant le système complexe.
- Les entrées de l'environnement sont les perturbations(les bruits ou les dynamiques qui ne sont pas modélisées) qui agissent sur le système complexe.
- Les entrées de coordination : qui sont utilisées pour la coopération avec d'autres systèmes complexes

Ces entrées peuvent être schématisées par la figure suivante :

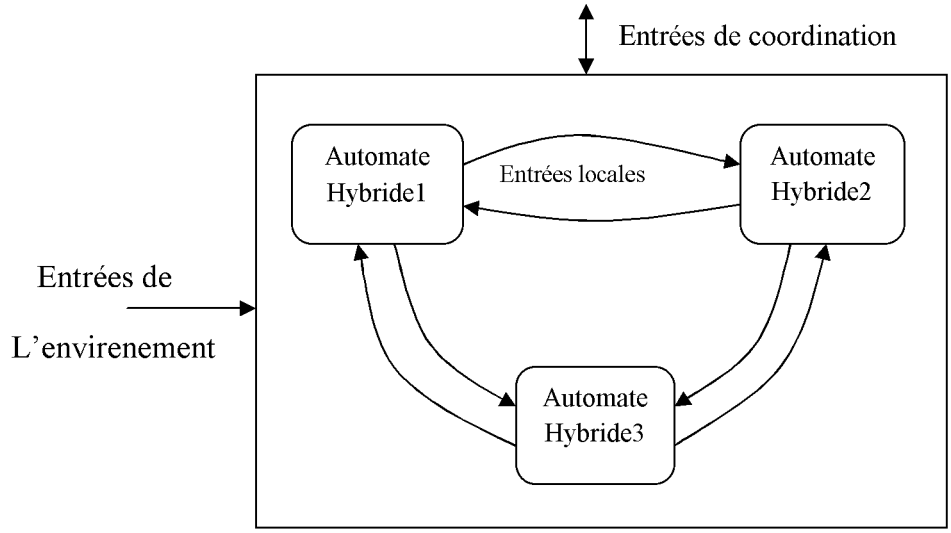


Figure (II.9) : Exemple de schématisation d'un système complexe

II.5 Modélisation des systèmes hybrides par les systèmes à interface

Cette modélisation est une représentation purement analytique des systèmes hybrides basée essentiellement sur un ensemble d'équations différentielles ou d'équations aux différences. Dans cette modélisation, le contrôleur est explicite puisqu'il est entièrement séparer du système commandé. Une interface est dans ce cas utilisée pour relier la partie discrète et la partie continue.

Nous allons présenter dans ce qui suit, les deux structures, la structure modulaire générale et la structure adaptée, ainsi que le modèle proposé par Brockett et le modèle à interface généralisée.

II.5.1 Structure modulaire générale (modèle d'Antsaklis)

Nous présentons la structure modulaire générale que nous adaptons pour les systèmes dynamiques hybrides. C'est principalement cette représentation qui sera utilisée par la suite comme aide à la simulation des SDH. [3]

Cette structure est composée de trois niveaux, comme représenté sur la figure suivante :

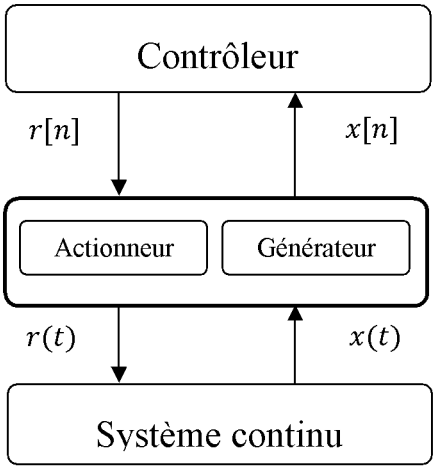


Figure (II.10) : Structure modulaire générale.

- Le système continu : il est représenté par la fonction différentielle suivante **:**

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

Avec :

$$x(t) \in X \subset R^n \quad : Le\ vecteur\ d'état$$
$$r(t) \in S \subset R^m \quad : Le\ vecteur\ d'entrée$$
$$f : X \times S \to X \quad : Une\ fonction\ continue$$

- Le contrôleur : c'est un SED représenté par un automate note A défini par le quintuple :

$$A = \big(\tilde{S}, \tilde{X}, \tilde{R}, \delta, \phi\big)$$

Avec :

$\tilde{S}$: L'ensembles des états .

$\tilde{X}$: L'ensembles des événements.

$\tilde{R}$: L'ensembles des symboles de sortie du contrôleur .

$\delta : \tilde{S} \times \tilde{X} \to \tilde{S}$: La fonction de transition.

$\phi : \tilde{S} \to \tilde{R}$: La fonction de sortie.

Le fonctionnement du contrôleur est défini par les équations suivantes :

$$s[n] = \delta(s[n-1], x[n]). \tag{II.5}$$

$$r[n] = \phi(s[n]) \tag{II.6}$$

Avec: $s[n] \in \tilde{S},\ x[n] \in \tilde{X}\ et\ r[n] \in \tilde{R}$.

- L'interface permet au système continu et au contrôleur de communiquer. Le générateur engendre des événements lorsque le vecteur d'état atteint certaines valeurs. Ces événements permettent à l'automate de changer d'état grâce à la fonction de transition δ. Suivant ce nouvel état, on obtient avec la fonction de sortie ϕ un nouveau symbole qui permet par l'intermédiaire de l'actionneur d'attribuer une valeur à l'entrée continue r(t).

  L'interface est donc composée de deux fonctions d et ξ telles que :

  $d : \tilde{R} \to R^m$ Convertir chaque événement du contrôleur en un signal d'entrée pour le système continu, nous avons alors $r(t) = d(r[n])$.

  $r[n]$ Étant l'événement le plus récent. L'entrée du système est alors constante par morceaux (escalier) qui change de valeur si l'événement du contrôleur se produit.

$$\xi : R^n \to \tilde{X}$$

$$x[n] = \xi(x(t)) \tag{II.7}$$

Remarque II.5.1

L'ensemble système continu et interface se comporte comme un SED dans la mesure où il accepte des entrées discrètes via l'actionneur et qu'il génère des sorties discrètes via le générateur.

On peut le représenter par un automate noté A_c défini par le quintuple :

$$A_c = (\tilde{P}, \tilde{X}, \tilde{R}, \psi, \lambda) \quad \text{(II.8)}$$

Avec :

$\tilde{P}$: L'ensembles des états .

$\tilde{X}$: L'ensembles des événements déja défini pour A.

$\tilde{R}$: L'ensembles des symboles de sortie du contrôleur déja défini pour A.

$\psi : \tilde{P} \times \tilde{R} \to \tilde{P}$: La fonction de transition.

$\lambda : \tilde{P} \to \tilde{X}$: La fonction de sortie.

II.5.2 Structure adaptée

Dans le but de nous conformer à la représentation formelle des systèmes dynamique hybrides, nous adaptons la structure générale afin qu'elle représente les quatre phénomènes hybrides et afin d'y faire apparaître la fonction G qui décrit ces phénomènes. Cette structure adaptée est représentée sur la figure suivante :

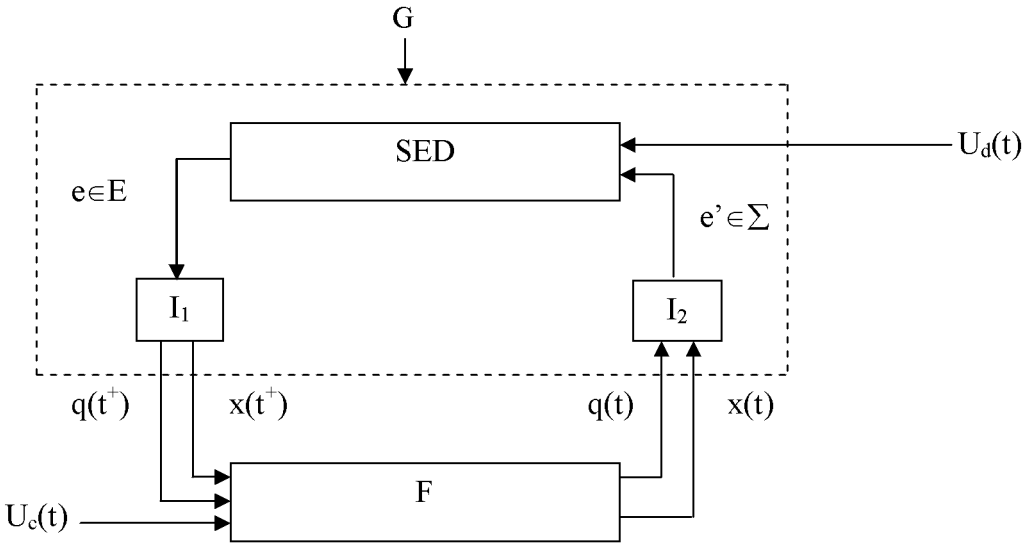


Figure (II.11): Structure adaptée

Cette structure possède également trois niveaux :

- La partie continue définie par les fonctions F ;
- Le SED est composé de k automates en parallèle notés $A_j \; avec \; j = 1, \dots, \dots, k$, définis par :

$$A_j = (E_j, \Sigma_j, \delta_j, e_{0j})$$

Avec :

E_j : L'ensembles des états .

Σ_j : L'ensembles des événements .

$\delta_j : E_j \times \Sigma_j \rightarrow E_j$: La fonction de transition.

e_{0j} : l'état initial.

Chaque automate A_j correspond à un phénomène hybride représenté par la fonction G. Cependant, on peut représenter le SED par un seul automate A en effectuant une composition parallèle des automates A_j. De cette manière, on peut utiliser les outils d'analyse des automates discrets.

L'automate A peut être défini de la manière suivante :

$$A_j = (E, \Sigma, \delta, e_0)$$

Avec :

E : L'ensembles des états .

$\Sigma : \Sigma_1 \cup \dots \dots \cup \Sigma_k$.

$\delta : E \times \Sigma \rightarrow E$.

$e_0 = (e_{01} \dots \dots e_{0k})$

De plus, on assimile les événements $u_d(t) \in U_d \subset \Sigma$ à des événements contrôlables et les événements $e' \in U_a$ à des événements incontrôlables (autonomes), avec $U_d \cup U_a = \Sigma$

L'interface est composée de deux fonctions I_1 et I_2.

- La première fonction I_1 affecte les valeurs à $q(t+)$ et $x(t+)$ suivant l'état $e \in E$ dans lequel est l'automate A.
- La deuxième fonction I_2 permet de déclencher un événement $e \in \Sigma$ lorsque l'état hybride $s(t) = [x(t), q(t)]$ atteint l'ensemble P. La commande discrète $u_d(t) \in U_d \subset \Sigma$ permet de déclencher des phénomènes hybrides contrôlés.

- La fonction G est composée des automates A_j mais également des fonctions I_1 et I_2. En effet, d'après la fonction G dépend de $x(t)$, q(t) et $u_d(t)$ et agit sur $x(t)$ et $q(t)$. Plus précisément, la fonction I_2 correspond au déclenchement des phénomènes tandis que l'automate et la fonction I_1 correspondent à l'action des phénomènes.**[12]**

I.5.3 Le modèle de Brockett

Ce modèle est décrit par les équations suivantes :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), z[p]) \\ \dot{p}(t) = r(x(t), u(t), z[p]) \\ z[p] = \eta(x[t], z[p], \eta[p]) \end{cases} \quad \text{(II.9)}$$

$x(t) \in X_c \subset R^n, u(t) \in U_c \subset R^m, p(t) \in R, v[p] \in V, z[p] \in Z,$

$f: X \times U \times Z \to R,$

$r: X \times U \times Z \to R,$

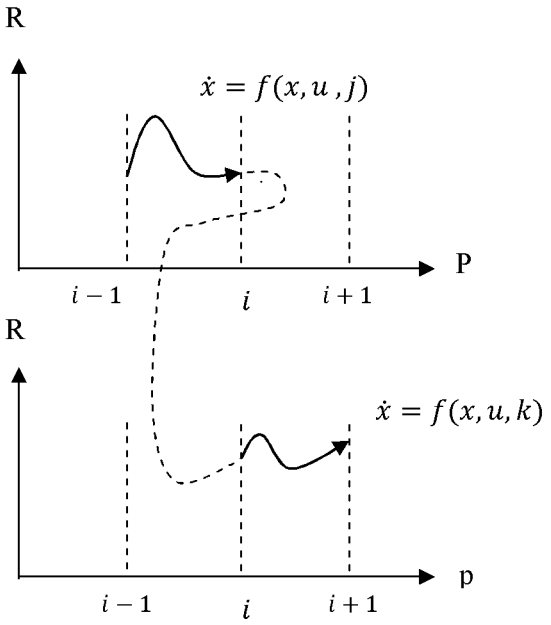
$\eta: X \times Z \times V \to Z$

V et Z sont des sous ensemble de N. la notation [t] dénotée la plus récente valeur de t à laquelle p devient entier. Ce type de modèle est noté MB (modèle de Brockett). Brockett a utilisé d'une façon simultanée les commandes continues et discrètes en introduisant une « horloge » ou un « compteur » variable p. la première équation est celle des dynamiques continues et la dernière est celle des traitements symboliques du système. **[D. Djouadi, 2003]**

u(t) : est la commande exercée en temps t,

$\eta[p]$: est la P^{eme} commande discrète (symbolique).

La figure suivante montre les dynamiques de MB.



Figure(II.12) : exemple des dynamiques du modèle de Brockett

Remarque II.5.3

$$t[p] = \{t \in R^+ : p(t) = [p]\}$$

Exemple II.5.3

Description d'un système de contrôle d'une automobile.

Cet exemple consiste en un système de conduite d'une automobile régit par les équations suivantes :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left[-a\big(x_2(t) + u(t)\big)/1 + z([p])\right] \\ \dot{p}(t) = 1 \\ z([p]+1) = f\left(z([p]), x_1\big(t_{[p]}\big), x_2\big(t_{[p]}\big)\right) \end{cases}$$

Où :

$x_1,\ x_2 \in R^n$: sont la vitesse linéaire et celle angulaire de l'engin respectivement,

$u(t) \in R^m$: est l'entrée externe telle que la position, le terme a(.) détermine la diminution de la capacité du système à produire des rotations à des vitesses élevées, $z \in N$ représnte le changement de la position, et $f: N \times R \times R^n \rightarrow N$ est la fonction de déplacement. La variable p représente une « horloge » ou un « compteur », $l \in R^+$ dépend implicitement de x_1, x_2, u et z.

Ce modèle de Brockett peut être vu comme un système à interface. Il sera schématisé par les blocs donnés par la figure suivante :

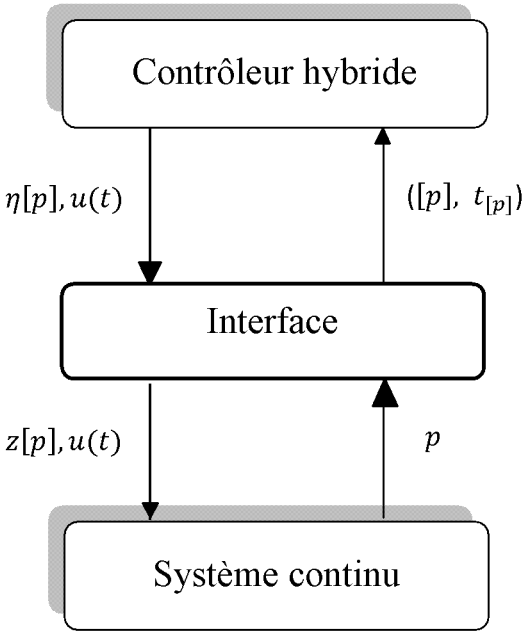


Figure (II.13) : Structure à interface équivalent au modèle de brockett.

Dans ce cas, l'interface a pour rôle de détecter les instants où p passe des valeurs entières. C'est donc un échantillonnage du temps qu'elle effectue en fonction des valeurs de l'horloge. Une fois que p passe par des entiers, un événement est généré par l'interface et est envoyé au contrôleur qui doit calculer, en fonction de celui-ci, les commandes discrètes et continues. **[12]**

II.5.4 Modèle à interface généralisé

Le modèle à interface généralisé que nous présentons ci-dessous tient compte de toutes les particularités des deux modèles déjà exposés. C'est ainsi que ce modèle est capable de décrire l'aiguillage entre les champs de vecteurs, les sauts discontinus de l'état de l'interface de la partie discrète sur la partie continue et vice versa. Il est constitué d'un système hybride en boucle ouverte (SHBO) contrôlé par un calculateur hybride (CH) comme le montre la figure suivante :

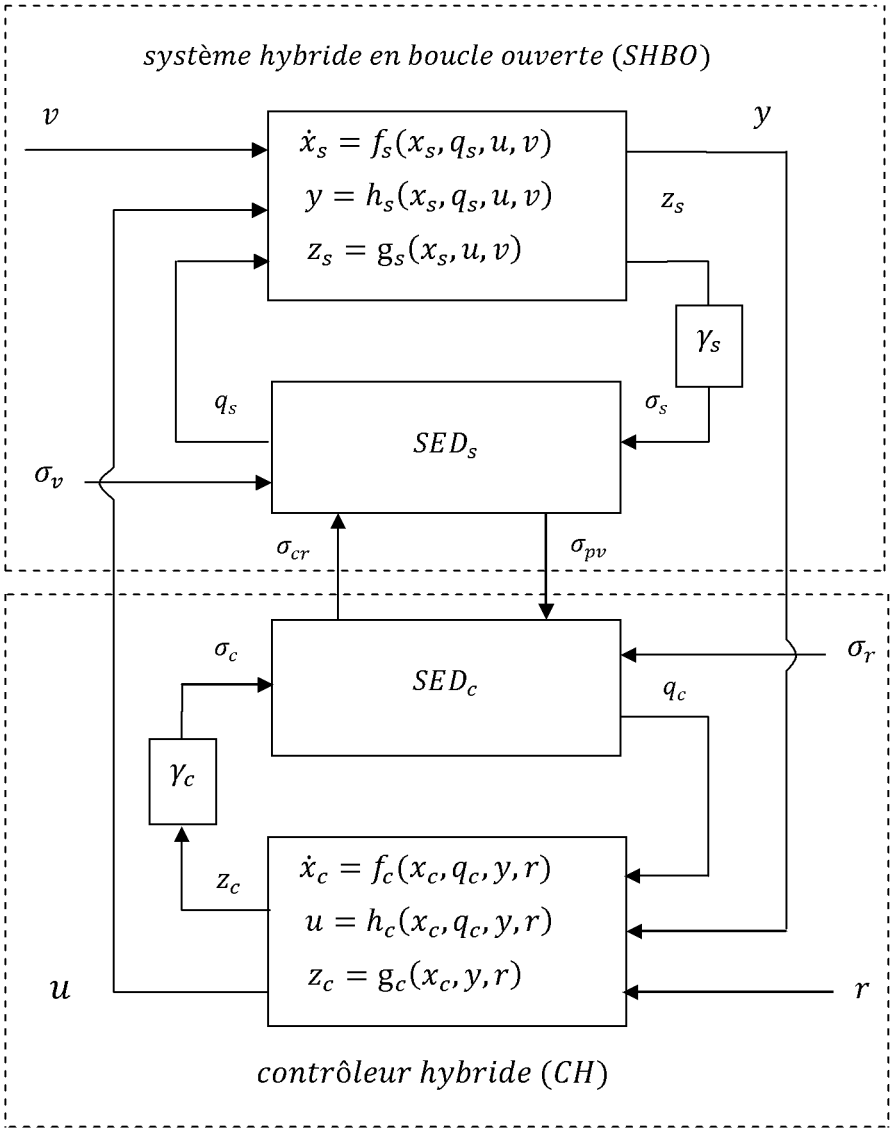


Figure (II.14) : Schéma détaillé d'un modèle de système hybride à interface généralisé

- Le SHBO est défini par le triplet :

$$SHBO : < D_{SHBO}, C_{SHBO}, I_{SHBO} >$$

a) D_{SHBO} : il représente la partie discrète du SHBO (c'est le système à événements discrets SEDs du système à contrôleur) et est défini par le triplet :$D_{SHBO} =< Q_{SHBO}, \Sigma_{SHBO}, \Psi_{SHBO} >$ qui consiste en un ensemble d'états discrets du SEDs, l'ensemble des événements d'entrée et la fonction de transition d'état respectivement.

- L'état du système SEDs au temps t est $q(t) = q_i \in Q_S$ tel que $Q_S = \{q_1, q_2, \dots \dots, q_i\}$ où chaque $q_i \in R^q$.
- Σ_S : est l'ensemble des événements d'entrée, $\Sigma_S = \Sigma_s \cup \Sigma_v \cup \Sigma_{cr}$, $\sigma_s \in \Sigma_S$ est l'événement interne du système généré par le processus continu à l'aide de générateur d'événements γ_s ; $\sigma_v \in \Sigma_v$ est un événement externe qui se produit sans influence d'aucune dynamique modélisée, il n'est pas donc inclue dans Σ_s; $\sigma_{cr} \in \Sigma_{cr}$ est l'événement issu de la partie discrète du contrôleur σ_s et σ_v sont incontrôlables et σ_{cr} est contrôlable.
- La fonction de transition d'état $\psi_s: Q_s \times \Sigma_{s\prime} \rightarrow Q_s$ détermine l'état suivant après un événement de $\sigma_{s\prime} \in \Sigma_{s\prime}$; $q(t) = \psi_s(q(t^-), \sigma_{s'})$.

b) C_{SHBO} : décrit les dynamiques de la partie continue du SHBO et est représenté par les équations donnée par la **figure (II.12)**

- $x(t) \in R^n$ est le vecteur d'état continu du système.
- $u(t) \in R^m$ est le vecteur de contrôle externe.
- $v(t) \in R^k$ est un vecteur externe de perturbation qui affecte l'état continu.

La fonction f_c est supposée continue pour l'état constant q(t). La sortie est $y(t) \in R^p$ et est générée par la fonction h_s, la fonction g_s , donne le signal $z \in Z_s \subseteq R^d$ qui est l'entrée du générateur d'événement γ_s de l'interface.

Pour exprimer les sauts dans la trajectoire d'état continue, nous utilisons la fonction de transition ϕ telle que :

$$x(t) = \phi_s(x(t^-),\ q(t^-),\ \sigma_{s'}). \qquad \text{(II.10)}$$

Elle détermine l'état prochain après un événement $\sigma_{s\prime} \in \Sigma_{s\prime}$ qui a provoqué un saut dans l'espace d'état (c'est un saut sans la même phase).

c) I_{SHBO} : c'est l'interface qui assure la génération des événements $\sigma_s \in \Sigma_s$ du système continu au SEDs $\gamma_s : Z_s \rightarrow \Sigma_s$ les événements se produisant pour deux raisons :
   - Si le système entre dans une région avec un comportement différent.
   - S'il ya un saut dans l'état continu.

Le contrôleur hybride (CH) est défini par le triplet :

$$CH: < D_{CH}, C_{CH}, I_{CH} >$$

Où chaque partie est définie d'une façon identique aux éléments du SHBO. L'interaction entre les parties continues et les parties discrètes peut être utilisée pour représenter différentes sortes de comportement des systèmes hybrides.

Etant donné un SHBO, la conception des CH a pour but de satisfaire des spécifications pour le système hybride en boucle fermée (SHBF) .en plus des buts généraux de robustesse et d'optimalité, les spécifications peuvent impliquer des restrictions sur le comportement du système en boucle fermée. Ces restrictions sont divisées en deux catégories :

- Restrictions sur la partie continue : pour les parties continues, les spécifications sont introduites sous fourme invariants. Un invariant est considéré comme une répartition de l'espace d'état continu du SHBO.
- Restrictions sur la partie discrètes : les spécifications discrètes sont introduites par un ensemble d'états interdits et autorisés pour le SED dans le but de limiter le comportement dynamique discret.**[12]**

II.6 modélisation par réseaux de Petri

II.6.1 Notion de base des réseaux de Petri

Les réseaux de Petri ont connu depuis leur apparition en 1962 par Carl Adams Petri [Pet62], un réel succès en raison de leur simplicité mathématique, des avantages de leur représentation graphique et de leur compacité. C'est l'un des outils les plus populaires pour la modélisation de systèmes à événements discrets et ses domaines d'applications sont très vastes.

Un réseau de Petri peut être vu comme étant un ensemble d'automates finis communicants. Cette vision doit être tempérée par deux remarques. La première remarque est que les réseaux de Petri sont plus généraux que les automates finis puisque, comme nous l'avons vu, des piles impliquant un nombre infini d'états peuvent être représentées. La deuxième remarque c'est que, par contre, les communications prises en compte sont statiques, c'est-à-dire qu'elles ne peuvent pas dépendre de données internes aux processus représentés par les automates. **[8]**

D'une manière informelle, Un réseau de Petri est un graphe orienté ayant deux types de nœuds, les places (représentées par des cercles) et les transitions (représentées par des arcs), des arcs permettent de relier une place à une transition ou une transition a une place. Un poids (nombre entier strictement positif) est affecté a chaque arc, ce poids vaut 1 quand ce n'est pas précis.

Le RdP est dit ordinaire si les valeurs de tous ses poids valent 1, et il est dit généralisé dans le cas contraire. L'ensemble des places ainsi que l'ensemble des transitions sont finis et non vides. Chaque place contient un nombre entier (qui peut être nul) de jetons ou marques, c'est le mouvement de ces jetons entre les places qui décrit la dynamique du système. Le marquage d'un RdP est un vecteur dont la dimension est égale au nombre de places et dont les composantes sont des entiers positifs ou nuls. La $i^{ième}$ composante indique le nombre de jetons dans la $i^{ième}$ place. Le graphisme des RDP hybrides est réalisé en adoptant les conventions suivantes :

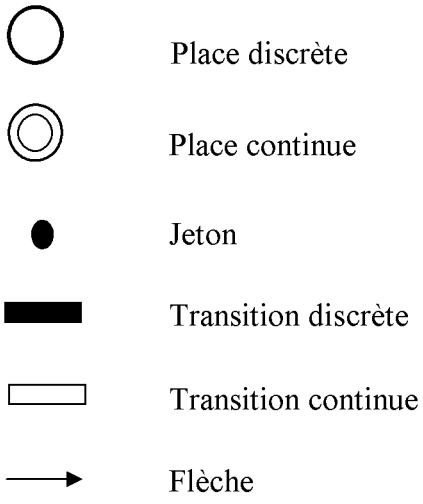


Figure (II.15) : Éléments composant un RdP hybride

II.6.2 Les Réseaux de Petri discrets (RDPD)

II.6.2.1 Définition

D'une manière plus formelle les RdP sont définis comme suit :

Un RDPD est entièrement décrit par la donnée de cinq paramètres : deux ensembles **P** et **T** et trois applications ***Pré***, ***Post*** et M_0. Il est donc représenté par les quintuplés :

$$\boldsymbol{Q : < P, T, Pré, Post, M_0 > \ .}$$

Où :

- $P = \{P_1, P_2, \dots, \dots, P_n\}$ est un ensemble fini, non vide de places.
- $T = \{T_1, T_2, \dots, \dots, T_n\}$ est un ensemble fini, non vide de transitions.

Les trois applications **Pré, Post** et **M** sont définies comme suit :

- **Pré** : $(P * T) \longrightarrow N$

  $(P_i, T_j) \longrightarrow Pré(P_i, T_j)$ = poids de l'arc reliant la place P_i à la transition T_J

- **Post** : $(P\ x\ T) \longrightarrow$ N

  $(P_i,\ T_j) \longrightarrow$ **Post** $(P_i,\ T_j)$ = poids de l'arc reliant la transition T_j à la place P_i

Ces deux applications (**Pré** et **post**) sont appelées respectivement application d'incidence avant et application d'incidence arrière.

- **M** : $P \longrightarrow N$

  $p_i \longrightarrow$ M(P_i) = le nombre de marque dans P_i $\ i = 1 \ldots\ldots\ldots\ldots n$

- $\mathbf{M_0}$: est le marquage initial.

On dit qu'un RDP est marqué, si on connait le marquage de chaque place P_i .Le marquage correspondant définit l'état du système décrit par le RDP. L'évolution du marquage correspond à l'évolution de l'état du système. **[6]**

Exemple II.6.2.1

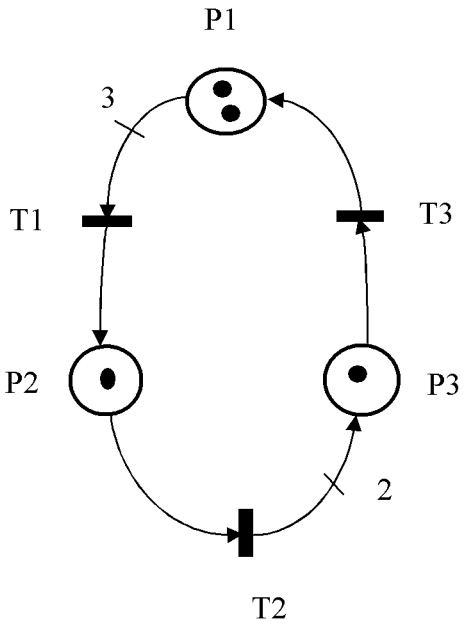


Figure (II.17) : Exemple de réseaux de Perti discret

- Un **RDP généralisé** est un RDP pour lequel les poids (nombres entiers strictement positifs) sont associés aux arcs.
- Un **RDP autonome** est un modèle de base qui permet d'étudier les propriétés logiques des systèmes, c'est-à-dire les propriétés qui dépend de l'ordre de événements, mais sans faire référence aux instants où ils se produisent (le temps n'intervient pas).

II.6.3 Règle de fonctionnement

Le fonctionnement d'un **RDP** marqué dépend de la validation de ses transitions. Pour qu'une transition T_j puisse être activée il faut que le nombre de marques dans chaque place **P_i** appartenant à l'ensemble des places d'entrée de T_j , soit supérieur ou égale au poids de l'arc $P_i \rightarrow P_j$ autrement dit T_j est validée si et seulement si **:**

$$\boldsymbol{M}(P_i) \geq \boldsymbol{Pr}\acute{e}(P_i, T_j) \text{ pour tout } P_i \in {}^{\circ}T_j.$$

Où : ${}^{0}T_j$ est l'ensemble des places en amont de T_j.

Une fois qu'une transition est validée, son franchissement est instantané et il modifie les marquages des places en amant (entrée) et en aval (sortie). Sachant qu'une seule transition à la fois est franchie dans un RDP ordinaire, l'effet de l'activation de la transition est de prélever un nombre de marques égales aux poids des arcs pour les places en amont de la transition et de rajouter dans chaque place en aval un nombre de marques égal aux poids des arcs les reliant à la transition. **[6]**

II.6.4 Les Réseaux de petri continus (RDPC)

Une limite pratique des RDP généralisés intervient lorsque le nombre de marques est très grand. En effet, cela provoque une augmentation du nombre d'états atteignables et donne, par conséquent, des durées de simulation importantes. Pour remédier à ce problème, on peut utiliser une approximation continue du modèle discret, ce qui conduit aux RDP continus. Dans ce modèle, les marques sont des valeurs réelles strictement positives.

II.6.4.1 Définition

La définition d'un **RDP continu** est similaire à celle d'un **RDP discret** sauf que les fonctions **Pré**, **Post** et le marquage initial prennent leurs valeurs dans $\mathbf{R}^{+}$ au lieu de **N.**

II.6.5 Les réseaux de petri T-temporels

L'idée fondatrice des RDP T-temporels est d'associer à un intervalle de temps $[\alpha_j, \beta_j]$ à chaque transition T_j. Si cette dernière est validée de façon continue pendant au moins α_j unités de temps, elle peut être franchie. De plus si elle est validée pendant $\beta_j - \alpha_j$ unités de temps de manière continue, elle doit être b franchie. On trouve ici les notions de délais

minimum et maximum dans un état au lieu de durée d'un état dans les modèles temporises. Intuitivement, les RdP temporels généralisent les RdP temporises et autonomes.

II.6.5.1 Les réseaux de petri temporisés

Un RDP temporisé permet de décrire un système dont le fonctionnement dépond du temps, c'est-à-dire qu'il peut s'écoulé une certaine durée de temps entre le début et la fin de celle-ci, les RDP temporisés sont utilisés pour rendre compte de cette durée. **[6]**

II.6.5.1.1 Définition

Un RDP hybride temporisé est un sextuple $\mathbf{Q_H = <P_H, T_H, Pré, Post, \Psi, M_0>}$,

avec :

- $\mathbf{P_H}$: un ensemble fini de places continues et discrètes.
- $\mathbf{T_H}$: un ensemble fini de transitions continues et discrètes.
- **Pré :** $\mathbf{P_H \times T_H} \longrightarrow A = \{0, 1\}$ ou N ou R^+, est l'application d'incidence avant.
- **Post :** $\mathbf{P_H \times T_H} \longrightarrow A = \{0, 1\}$ ou N ou R^+, est l'application d'incidence arrière.
- $\mathbf{M_0}$: est le marquage inital.
- $\mathbf{\Psi}$: $T \longrightarrow R^+ \cup \{\infty\}$, est une fanction qui attribue une vitesse à chaque transition

Si on considère une place quelconque notée $P_i \in P_H$ et une transition $T_j \in T_H$.

- Si $P_i \in P_c$ et $T_j \in T_C$ alors $A = \{0 , 1\}$.
- Si $P_i \in P_d$ et $T_j \in T_C$ alors $A = \{0 , 1\}$.
- Si $P_i \in P_d$ et $T_j \in T_d$ alors $A = N$.
- Si $P_i \in P_c$ et $T_j \in T_d$ alors $A = R^+$.

Exemple II.6.5.1.1

Considérons une vanne notée V_1, qui commande le remplissage du bac R_2 cela fait intervenir une variable discrète dont l'action peut être modélisée par un RDP généralisé.

Enfin, une fois le bac R_2 est rempli, 60% du produit va dans un bac R_3 commandé par la vanne V_2 avec un débit D_2 , ensuite, le reste de produit va dans un bac R_4 commandé par la vanne V_3 avec un débit D_3. Ce système peut être modélisé par le RDP hybride de la figure (II.20) suivant :

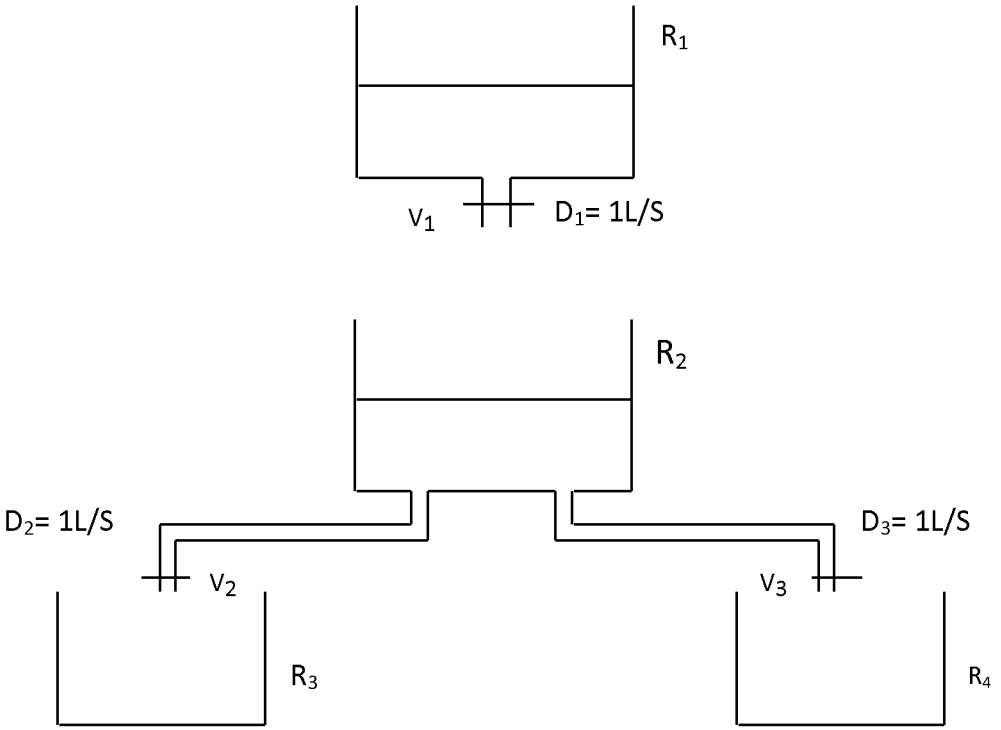


Figure (II.19) : Le remplissage des bacs

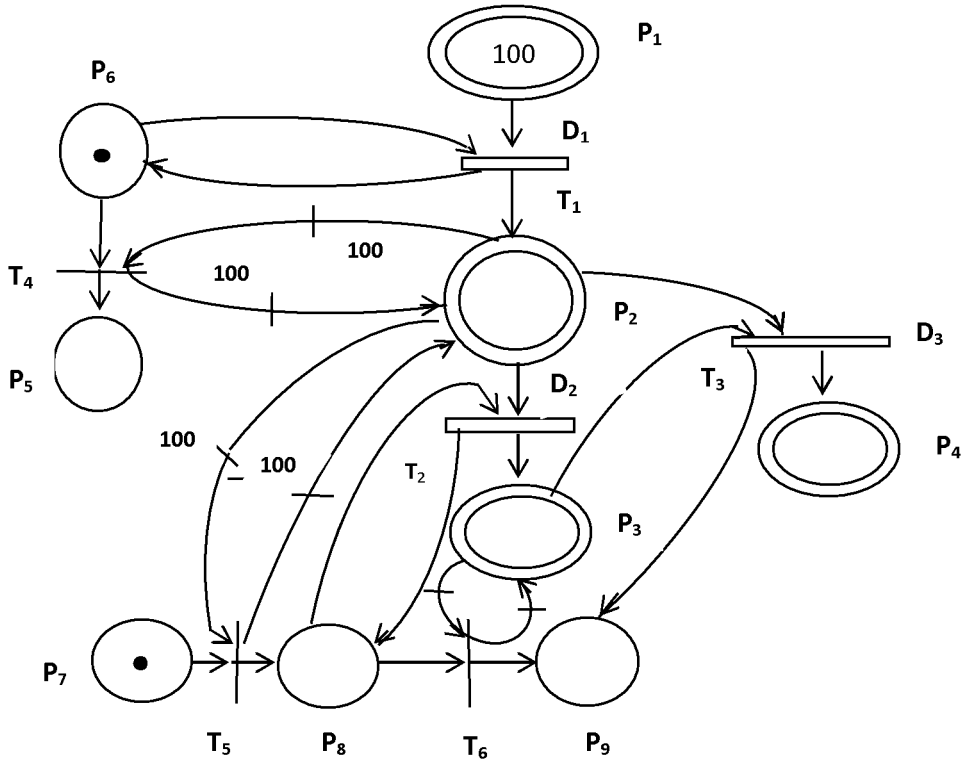


Figure (II.20) : La Modélisation par un réseau de petri hybride de remplissage des bacs

II.7 Méthodes de commande des systèmes hybrides

Contrairement au cas des systèmes continus ou des systèmes à événements discrets pour lesquels la notion de commande et les problèmes de synthèses associés sont bien identifiés et clairement définis, la commande des systèmes hybrides est une notion beaucoup plus large. Le fait que ses systèmes font intervenir deux types de dynamiques, une dynamique continue et une autre discrète, favorise une diversité des formulations du problème de commande hybride rencontrées dans la littérature.

II. 7.1 Commande à base de modèles moyens

Les méthodes de moyennisation se situent à l'extrémité « continue » des approches utilisées pour modéliser la dynamique des systèmes hybrides. Ils agissent de trouver un modèle différentiel unique pour approximer le fonctionnement global d'un système dynamique hybride constitué de plusieurs modèles différentiels commutant. Ces méthodes sont largement répondues en électronique et électrotechnique pour approcher le fonctionnement de système presque périodiques.

L'application de ces méthodes concerne tous les systèmes qui présentent un fonctionnement « presque périodiques » que nous caractérisons par un cycle le parcours de l'automate décrivant les modèles actifs. La durée de parcours du cycle étant constante ou lentement variable.

II.7.1.1 Quelques définitions

- Forme standard : « la forme standard »utilisée dans les problèmes de moyennisation est la suivante :

$$\dot{x}(t) = \varepsilon f(t,x) \qquad x(t_0) = x_0 \qquad \text{(II.11)}$$

Où ε est un « petit » paramètre, f un champ de vecteur défini de $R^n \times R^n \; sur \; R^n, \; et \; x_0 \in D$

Avec D un ouvert borné de R^n.

- Valeur moyenne d'une fonction : lorsque f est T périodique en t, on appelle « moyenne de f », l'application $f_0(x)$ défini de R^n sur R^n avec :

$$f_0(x) = \frac{1}{T}\int_0^T f(x,t)dt \qquad \forall x \in D \tag{II.12}$$

Lorsque f ne présente pas de périodicité, on peut étendre la notion de valeur moyenne par la formule (II.11) lorsque cette limite existe. On dit alors que f est Krylov, Bogoliobov, Mitropolsky (KBM).

$$f_0(x) = \lim_{T\to\infty} \frac{1}{T}\int_0^T f(x,t)dt \qquad \forall x \in D \tag{II.13}$$

- Moyenne glissante d'une fonction : on appelle « moyenne glissante » du champ de vecteur f, le champ de vecteur f_T défini par :

$$f_T(t,x) = \frac{1}{T}\int_0^T f(t+\tau,x)d\tau = \frac{1}{T}\int_t^{T+1} f(\tau,x)d\tau \tag{II.14}$$

A partir de ces définitions, on pourra définir ce qu'on appelle un modèle moyen.

- Modèle moyen : on appellera « modèle moyen du modèle » (II.11) le modèle défini par l'équation suivante :

$$\dot{\bar{x}} = \varepsilon f_0(\bar{x}) \qquad \bar{x}(t_0) = x_0 \tag{II.15}$$

Avec $\bar{x}$ le vecteur d'état moyen.

- Modèle moyen glissant : on appellera « le modèle moyen glissant » le modèle :

$$\dot{\tilde{x}} = \varepsilon f_T(t,\tilde{x}) \tag{II.16}$$

Avec $\tilde{x}$ le vecteur d'état moyen glissant et $\tilde{x}(t_0) = x_0$.

Théorèmes

Si les champs de vecteurs satisfont quelques propriétés classiques grâce auxquelles le problème (II) admet une solution unique dans $D \subset R^n$, alors on a plusieurs théorèmes qui assurent que les solutions des problèmes moyens $(\tilde{x})$et $(\bar{x})$ seront d'autant plus proches de (x) et d'autant plus longtemps que ε est petit. Ce sont donc des théorèmes de limite quand $\varepsilon \to 0$, ce qui garantit la régularité de l'approche de moyennisation. Nous ne citerons ici que deux résultats.

Théorème 1 : [1]

Si la fonction f satisfait à :

a) f est bornée par une constante M sur D.

b) f satisfait à une condition de lipschits avec la constante λ .

c) f est KBM $\forall \in D$.

Alors le modèle moyen $\bar{x}$ admet une solution :

$\forall \eta > 0, \forall L, \exists \varepsilon_0$ tel que :

$\forall \varepsilon \in]0, \varepsilon_0]$ et $\forall\, t \in [t_0, t_0 + L/\varepsilon\,[$ on a : $\|x(t) - \bar{x}(t)\| < \eta$ $(II.17)$

Quant au deuxième résultat, si f est KBM $f_T(t,x) = f_0(x) + o(\frac{\delta(\varepsilon)}{\varepsilon T})$ avec :

$$\delta(\varepsilon) = \sup_{x \in D} \sup_{t \in [t_0, \frac{T}{\varepsilon}]} \varepsilon \left\| \int_0^T [f(\tau, x) - f_0(x)]\, d\tau \right\| \qquad (II.18)$$

Rappelons que l'on note $\delta_1(\varepsilon) = o(\delta_2(\varepsilon))$ quand $\varepsilon \to 0$, s'il existe un scalaire k > 0 tel que $|\delta_1(\varepsilon)| \leq k|\delta_2(\varepsilon)|$ quand $\varepsilon \to 0$ .on a alors le théorème suivant :

Théorème 2

Si f satisfait aux conditions du théorème précédant, alors les solutions des modèles moyen $(\bar{x})$ et moyen glissant $(\tilde{x})$ sont telles que :

$$\tilde{x}(t) = \bar{x}(t) + o(\frac{\delta(\varepsilon)}{\varepsilon T}) \quad \text{pour } t \in [t_0,\ t_0 + L/\varepsilon] \qquad (II.19)$$

Remarque II.7.1

Même la commande à base de modèles moyens occulte les problèmes lies à la partie séquentielle des systèmes hybrides, elle s'applique à de nombreux systèmes hybrides pour lesquels on ne s'intéresse qu'aux valeurs moyennes des variables continues. C'est alors qu'il faudra penser à utiliser ces techniques en restant conscient de leurs limites (ε petit et possibilité de traiter des changements de modes de marche par rapport au cycle de fonctionnement étudié).

II.7.2 Problème de commande optimale des systèmes dynamiques hybrides

On s'intéresse à une classe de systèmes dynamiques hybrides pouvant être considérées comme des processus continus évoluant dans un intervalle de temps $[t_0, t_f]$ et dans lesquels des événements discrets contrôlés ou autonomes provoquent des modifications de la structure de leur modèle. On suppose que ces changements de modèle n'entraînent que des discontinuités dans la dérivée du vecteur d'état du système.

On rappelle que la dynamique continue de ces systèmes est définie par l'équation (II.6) où q(*t*) représente la variable discrète du système :

$$\dot{x}(t) = F\big(x(t), q(t), u(t)\big) \tag{II.20}$$

Si les commutations sont contrôlées, la dynamique hybride s'écrit :

$$q(t^+) = G_{cc}(v(t)) \tag{II.21}$$

Dans ce cas, la commande discrète v(t) détermine les instants de commutation.

Si les commutations sont autonomes, la dynamique hybride s'écrit :

$$q(t^+) = G_{CA}(x(t), q(t)) \tag{II.22}$$

II.8 Formulation du problème de commande

Dans cette partie, on formule le problème de manière générale, en incluant les commandes discrète, hybride et continue. On distinguera ensuite de différents cas.

Les changements de modèle interviennent dans l'intervalle de temps $[t_0, t_f]$ à des instants $\tau_1, \tau_2, \dots, \tau_r$ appelés instants de commutation que l'on suppose en nombre fini et vérifiant :

$$t_0 = \tau_0 < \tau_1 < \dots\dots < \tau_r < \tau_{r+1} = t_f$$

On note $\xi_i = x(\tau_i), i = 1, \dots\dots, r.$

Ce système est régi, dans chaque intervalle de temps $[\tau_{i-1}, \tau_i]$ par une équation notée comme ceci :

$$\dot{x}(t) = f_i(x(t), u(t), t) \;\; t \in (\tau_{i-1}, \tau_i), i = 1, \dots\dots, r+1 \tag{II.23}$$

Où $\dot{x}(t) = f_i(x(t), u(t), t)$ est une fonction à valeur dans R^n de classe C^1 des ses trois arguments sur l'intervalle $[\tau_{i-1}, \tau_i]$

Le problème de commande optimale que l'on pose consiste à chercher une commande u*, les instants de commutation τ* et le temps final t_f* qui minimisent le critère :

$$J(u, \tau_1, \dots\dots, \tau_r, \tau_f) = \sum_{i=1}^{r+1} \int_{\tau_{i-1}}^{\tau_i} M_i(x(t), u(t), t)dt + \sum_{i=1}^{r} N_i(\xi_i, \tau_i) + K(x_f, t_f) \qquad \text{(II.24)}$$

Où :

$M_i(x(t), u(t), t)$: est une fonction à valeur dans R de classe C^1 de ses trois arguments sur l'intervalle $[\tau_{i-1},\ \tau_i]$.

$N_i(\xi_i, \tau_i)$: est une fonction à valeur dans R de classe C^1 de ses deux arguments, appelée coût du changement de modèle.

$K(x_f, t_f)$: est une fonction à valeur dans R de classe C^1 de ses deux arguments, appelée coût final. **[1]**

Remarques II.8

i. Les points de discontinuités de la dérivée $\dot{x}(t)$ sur l'intervalle $[t_0, t_f]$ sont en nombre finis, on peut donc étudier le cas d'une seule commutation de modèle se produisant à un instant τ lorsque l'état du système est $x(\tau) = \xi$ L'extension au cas de plusieurs instants de commutations de modèle est immédiate.

ii. L'instant de commutation τ n'est en général pas connu a priori, il faut donc envisager également des variations par rapport à ce paramètre.

II.9 Commande prédictive des systèmes hybrides sous forme MLD

Un système dynamique hybride permet, de façon générale de réaliser l'interface entre des ensembles possédant des dynamiques continues, au sens classique des équations différentielles, et des éléments décisionnels à évènements discrets pour une classification des systèmes hybrides. [2]. [1]

II.9.1 La forme MLD

L'idée de base de la forme MLD, consiste à exprimer une règle logique par des contraintes linéaires portant sur des variables entières, opérant une traduction des règles logiques en inégalités linéaires. Ainsi, à partir de la description standard d'un système, les propositions logiques sont transformées en équations présentant une certaine dynamique,

similaire à ce qui est réalisé pour la partie continue, mais avec des contraintes additionnelles issues des variables logiques. Le tout est regroupé au sein d'une représentation incluant un ensemble d'inégalités, comme indiqué par la relation suivante :

$$\begin{aligned} X(t+1) &= Ax(t) + B_1u(t) + B_2\delta(t) + B_3Z(t) \\ Y(t) &= Cx(t) + D_1u(t) + D_2\delta(t) + D_3Z(t) \\ E_2\delta(t) + E_3Z(t) &\leq Y(t) = E_1u(t) + E_4x(t) + E_5 \end{aligned} \qquad (\text{II.25})$$

Où :

$$X = \begin{pmatrix} x_c \\ x_l \end{pmatrix} \in R^{nc} \times \{0,1\}^{nl}, \quad u = \begin{pmatrix} u_c \\ u_l \end{pmatrix} \in R^{mc} \times \{0,1\}^{ml}$$

$$Y = \begin{pmatrix} y_c \\ y_l \end{pmatrix} \in R^{pc} \times \{0,1\}^{pl}, \quad \delta = \begin{pmatrix} u_c \\ u_l \end{pmatrix} \in R^{rl} \times \{0,1\}^{rl}, z \in R^{rc}$$

Sont respectivement les vecteurs des variables d'état continues et binaires du système, des signaux de commande continus et binaires (on/off), des signaux de sortie, des variables auxiliaires continues et binaires. Les variables auxiliaires sont introduites lors de la traduction de la logique propositionnelle en inégalités linéaires (**figure II.19**)

A; $\{B_j\}_{j=1\ldots3}$; C; $\{D_j\}_{j=1\ldots3}$; $\{E_j\}_{j=1\ldots5}$ Sont des matrices réelles

II.10 Conclusion

Dans ce chapitre, nous avons dressé un panorama des diverses modélisations des systèmes hybrides proposées dans les littératures. La représentation du système sous forme d'un automate hybride ou d'un réseau de petri hybride ou d'une structure modulaire permet d'avoir une vue plus fonctionnelle du SDH. Ainsi l'automate hybride et le réseau de petri hybride ont l'avantage de mettre en évidence les modes de fonctionnement du système ainsi que le passage d'un mode de fonctionnement a un autre. La structure modulaire quant à elle permet de mieux visualiser les interactions entre le système continu et le système à évènement discrets. Nous avons aussi distingué deux types de représentation à savoir une représentation analytique et une autre graphique. Les modèles à interface qui sont essentiellement basent sur la représentation analytique utilisent une interface pour relier la partie continue à la patrie discrète. Nous avons présenté deux types de modèle à interface très répandus en littérature et qui sont : le modèle d'Antsaklis et le modèle de broket. La synthèse de la bibliographie disponible nous a permis de représenter un modèle à interface généralisé capable de décrire les phénomènes hybrides.

Dans la seconde patrie de ce chapitre, quelques approches de commande de ces systèmes ont été présentées. L'approche à base de modèle moyens et le formalisme MLD qui s'adapte à de très nombreux systèmes hybrides, tels que les systèmes à entrées et états mixtes continus et discrets.

III.1 Introduction

Nous allons, dans cet exemple d'application, mettre en œuvre Stateflow pour simuler le comportement d'un système hybride. Le processus (partie opérative) est un convertisseur DC-DC de puissance caractérisé par un comportement fortement non linéaire. La partie commande est un automate (machine à état finis). La boucle de fonctionnement est un système hybride que nous allons simuler grâce à Simulink et Stateflow.

III. 2 Description de système étudié

III.2.1 Les convertisseurs continu-continu

Ce type de convertisseur, admet des tensions et des courants d'entrée continus et produit en sortie des tensions et des courants continus.

Comme convertisseurs DC-DC de base, on trouve six : Buck, Boost, Buck-Boost, Cuk, Sepic, Zeta. Les trois premiers sont les plus simples et les plus utilisés, les trois derniers peuvent être construits par une combinaison des premiers.

Notre application est consacrée à l'étude de Buck et boost ; chaqu'un de ces convertisseurs est constitué par un interrupteur « Tp », un semi-conducteur passif unidirectionnel souvent la diode « D », des éléments de stockage : l'inductance « L » et le condensateur « C », E : est la tension d'entrée, Vs : est la tension obtenue en sortie et R : la charge vers laquelle l'énergie est transmise.

Dans ces convertisseurs les grandeurs à convertir sont dirigées par la mise en ON ou OFF de l'interrupteur « Tp ».

- **A) Le convertisseur Buck**

Il s'appelle aussi hacheur série ou abaisseur de tension, il convertit une tension continue en une autre tension continue de plus faible valeur. Son schéma est donné par la Figure (III.1). Tp et D sont les interrupteurs du système (électriquement ce sont des semi-conducteurs) fonctionnant en tout ou rien. D est une diode à commutation autonome et Tp est un transistor (ou thyristor) commandé. Le transfert d'énergie pour ce convertisseur est directe.

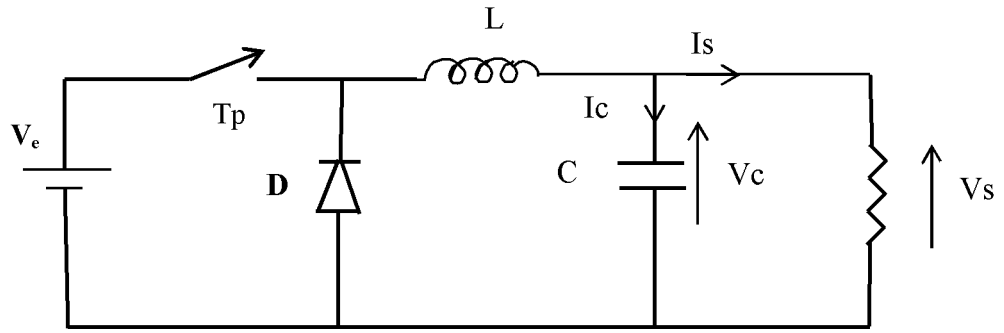


Figure (III.1) Convertisseur Buck

- **B) Le convertisseur Boost**

La fonction principale de ce convertisseur est de délivrer une tension de sortie supérieur à la tension d'entrée, c'est un élévateur de tension ou survoler, il s'appelle aussi hacheur parallèle, son schéma est donné par la Figure (III.2).

Le transfert d'énergie est indirecte, c'est seulement la charge de l'inductance « L » qui aura lieux quand « Tp=1 », et lorsque « Tp=0 » ou OFF le transfert d'énergie se fait grâce à la décharge de l'inductance « L », et la charge de condensateurs se produit seulement quand « Tp=0 »

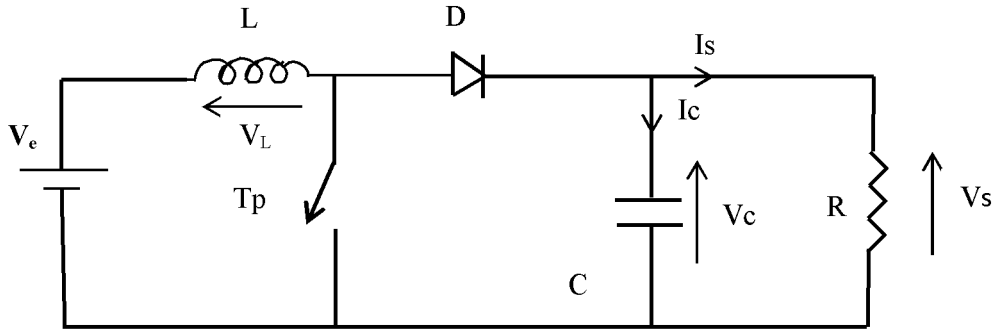


Figure (III.2) Convertisseur Boost

III.3 Mode de fonctionnement

Les convertisseurs de puissance DC-DC présentent deux modes de fonctionnement : mode de conduction continue (CCM) et mode de conduction discontinue (DCM).En mode de conduction discontinue, le circuit est décrit par trois configurations différentes. Les deux premières sont identiques avec les configurations en mode (CCM), tandis que la troisième configuration apparait si la diode devient bloquée durant le temps que l'interrupteur est dans la position **ouverte**.

Ces modes de fonctionnement sont différenciés en fonction de la forme d'onde du courant « i_L » qui traverse l'inductance « L » en régime stationnaire. Pour la suite, on considère le MOSFET comme un interrupteur parfait à deux positions : **ouverte** si le transistor est bloqué et **fermée** s'il est en conduction, et commandé par une horloge périodique binaire de signale (t), avec la période « Ts= Ton + Toff », Ton et Toff sont les temps pendant lesquels l'interrupteur « Tp=1 et Tp =0 » respectivement.

Dans ce mode de fonctionnement on constate trois états **Son**, **Soff** et **Snc**. Les états **Son** et **Soff** associé aux états marche et arrêt de commutateur respectivement et l'état **Snc** associé à l'état au la diode « D » ne conduit pas (l'intensité $\mathbf{i_L}$ est nulle). Le schéma suivant résume la forme de i_L pendant les deux fonctionnements.

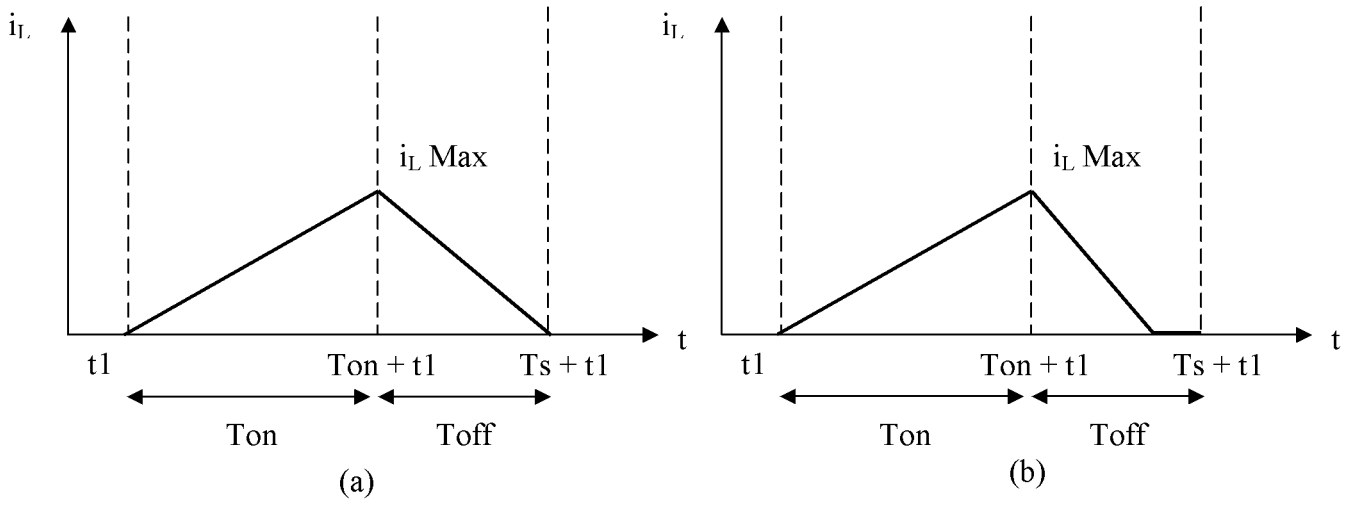


Figure (III.3) : Forme d'onde du courant en CCM (a) et DCM (b).

III.3.1 Le fonctionnement de convertisseur Buck

Selon la position de l'interrupteur, on peut différencier deux configurations possibles du circuit. Dans la première configuration l'interrupteur se trouve dans la position **fermée** et la diode est bloquée, ce qui permet de stocker l'énergie fournie par la source d'alimentation dans l'inductance. La valeur du courant augmente linéairement. Le schéma équivalnt pour cette situation est le suivant :

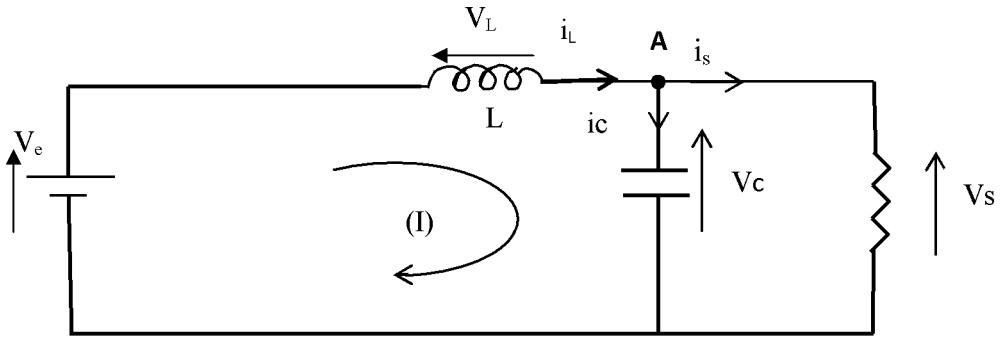


Figure (III. 4) : Convertisseur Buck avec Tp fermé

En appliquant les lois des nœuds et des mailles, on obtient les équations suivantes :

- Maille (I) :

$$Ve(t) - V_L(t) - Vc(t) = 0$$

$$V_L(t) = -Vc(t) + Ve(t)$$

$$\frac{Ldi_L(t)}{dt} = -Vc(t) + Ve(t)$$

$$\frac{di_L(t)}{dt} = -\frac{1}{L}Vc(t) + \frac{1}{L}Ve(t) \qquad (III.1)$$

- Nœud (A) :

$$i_L(t) = i_c(t) + i_s(t)$$

$$i_L(t) = \frac{dq(t)}{dt} + \frac{Vs(t)}{R}$$

$$i_L(t) = C\frac{dVc(t)}{dt} + \frac{Vc(t)}{R}$$

$$\frac{dVc(t)}{dt} = \frac{1}{C}i_L(t) - \frac{1}{RC}Vc(t) \qquad (III.2)$$

Une fois que les équations différentielles sont obtenus, en écrit le modèle d'état comme suit :

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} Ve(t) \qquad (III.3)$$

Dans la seconde configuration, lorsque l'interrupteur est dans la position **ouverte,** la diode devient passante et l'énergie emmagasinée est restituée à la charge « R » et le condensateur « C », ce qui se traduit par la diminution de la valeur du courant. Pour l'analyse nous pouvons remplacer la diode par un conducteur et supprimer le commutateur et la source de tension « Ve », le circuit en résultat est composé de deux mailles et il est décrit par les équations et le schéma suivants avec Ve =0 et i_L>0.

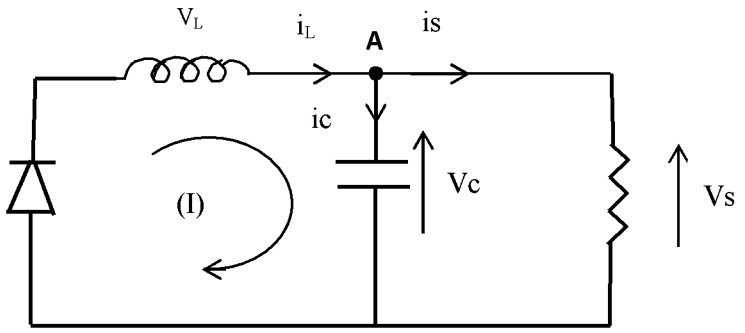


Figure (III.5) : Convertisseur Buck avec Tp ouvert et Ve = 0

- Maille (I) :

$$V_L(t) + Vc(t) = 0$$

$$V_L(t) = -Vc(t)$$

$$\frac{di_L(t)}{dt} = -\frac{1}{L}Vc(t) \qquad (III.4)$$

- Nœud (A) :

$$i_L(t) = i_c(t) + i_s(t)$$

$$i_L(t) = \frac{dq(t)}{dt} + \frac{Vs(t)}{R}$$

$$i_L(t) = C\,\frac{dVc(t)}{dt} + \frac{Vc(t)}{R}$$

$$\frac{dVc(t)}{dt} = \frac{1}{C}i_L(t) - \frac{1}{RC}Vc(t) \qquad (III.5)$$

On résume les équations $(III.4)$ et $(III.5)$ dans le modèle suivant :

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} Ve(t) \qquad (III.6)$$

Dans le mode où le commutateur est ouvert et « $i_L > 0$ » la tension « Vc » dans le condensateur augmente et l'intensité « i_L » diminue ; si le temps **Toff** est assez long il sera un moment où il est annulé ($i_L=0$), plus tard le condensateur essayant de décharger par la diode, ce qui n'est pas possible, car elle devient bloquée, dans ce cas nous pouvons considérer le circuit réduit à une maille simple décrite par les équations suivantes :

$$\frac{di_L(t)}{dt} = 0 \qquad (III.7)$$

$$\frac{dVc(t)}{dt} = -\frac{1}{RC}Vc(t) \qquad (III.8)$$

Et leur modèle est comme suit :

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} Ve(t) \qquad (III.9)$$

III.3.2 Le fonctionnement de convertisseur Boost

Dans le premier mode, l'interrupteur se trouve dans la position fermée et la diode est bloquée, donc elle peut être éliminée pour l'analyse. Dans cet état la maille du côté gauche est isolée, et son schéma est donné comme suit :

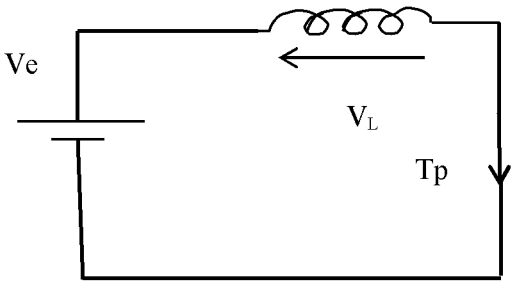


Figure (III. 6) : Convertisseur Boost avec Tp fermé

En appliquant les lois des nœuds et des mailles, on aboutit aux équations suivantes :

- Maille (I) :

$$Ve(t) - V_L(t) = 0$$

$$V_L(t) = Ve(t)$$

$$\frac{Ldi_L(t)}{dt} = Ve(t)$$

$$\frac{di_L(t)}{dt} = \frac{1}{L}Ve(t) \qquad (III.10)$$

- Nœud (B) :

$$i_c(t) + i_s(t) = 0$$

$$\frac{dVc(t)}{dt} = -\frac{1}{RC}Vc(t) \qquad (III.11)$$

Et le modèle d'état est comme suit :

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} Ve(t) \qquad (III.12)$$

Dans le deuxième mode, lorsque l'interrupteur est dans la position **ouverte,** la diode devient passante et l'énergie emmagasinée est restituée à la charge « R » et le condensateur « C ». Pour l'analyse nous pouvons remplacer la diode par un conducteur et éliminer le commutateur « Tp », le circuit résultant avec deux mailles est décrit comme suit :

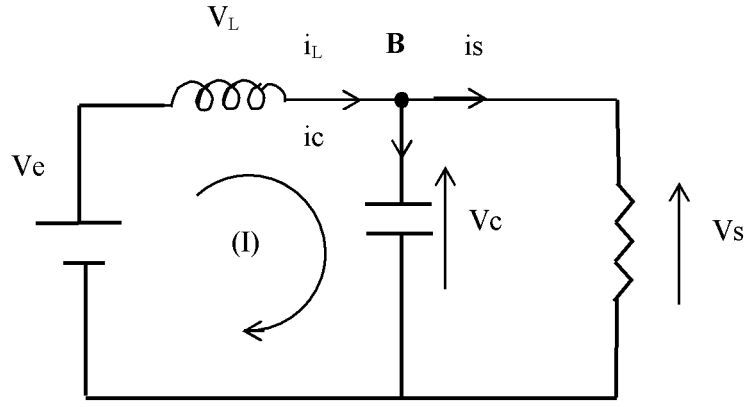


Figure (III. 7) : Convertisseur Boost avec Tp ouvert

- Maille (I) :

$$Ve(t) - V_L(t) - Vc(t) = 0$$

$$V_L(t) = -Vc(t) + Ve(t)$$

$$\frac{Ldi_L(t)}{dt} = -Vc(t) + Ve\ (t)$$

$$\frac{di_L(t)}{dt} = -\frac{1}{L}Vc(t) + \frac{1}{L}Ve(t) \qquad (III.13)$$

- Nœud (B) :

$$i_L(t) = i_c(t) + i_s(t)$$

$$i_L(t) = \frac{dq(t)}{dt} + \frac{Vs(t)}{R}$$

$$i_L(t) = C\ \frac{dVc(t)}{dt} + \frac{Vc(t)}{R}$$

$$\frac{dVc(t)}{dt} = \frac{1}{C}i_L(t) - \frac{1}{RC}Vc(t) \qquad (III.14)$$

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} Ve(t) \qquad (III.15)$$

Dans le mode où le commutateur est ouvert et « $i_L > 0$ » la tension « Vc » dans le condensateur augmente et l'intensité « i_L » diminue; si le temps **Toff** est assez long il arrivera un moment où il est annulé ($i_L=0$), plus tard le condensateur essayant de se décharger à travers la diode, ce qui n'est pas possible, car elle devient bloquée, dans ce cas nous pouvons considérer le circuit réduit à une maille simple décrite par les équations suivantes :

$$\frac{di_L(t)}{dt} = 0 \qquad (III.16)$$

$$\frac{dVc(t)}{dt} = -\frac{1}{RC}Vc(t) \qquad (III.17)$$

Et le modèle d'état est le suivant :

$$\begin{bmatrix} \frac{di_L(t)}{dt} \\ \frac{dVc(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix} \begin{bmatrix} i_L(t) \\ Vc(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} Ve(t) \qquad (III.18)$$

III.4 Modèle d'état

En fonction de l'état passant ou bloqué de la diode, et la position ouverte et fermée de l'interrupteur, on a trois dynamiques linéaires différentes caractérisant le comportement dynamique du convertisseur et présentés sous le modèle d'état suivant :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

Avec : u(t) = Ve(t) et $x(t)$ est le vecteur d'état

Généralement, l'ordre du modèle d'un convertisseur et la dimension du vecteur d'état considéré sont déterminés par le nombre d'éléments du circuit qui emmagasinent d'énergie.

Dans le cas du convertisseur buck et Boost, l'inductance et le condensateur représentent ces éléments. Ainsi, le vecteur d'état $x(t) = [x_1(t)\ x_2(t)]^T = [i_L(t)\ Vc(t)]^T$est composé par le courant de l'inductance $i_L(t)$et la tension aux bornes du condensateur Vc(t).

Cependant, du point de vue pratique il est préférable de pouvoir mesurer tous les états du modèle du convertisseur. Par conséquence, la deuxième variable d'état Vc(t)est remplacée par la tension aux bornes de la résistance de charge Vs(t)et le vecteur d'état est défini par $x(t) = [i_L(t)\quad Vs(t)]^T$.

Les matrices A et B qui représentent le modèle d'état on les donne dans le tableau suivant :

	A	B	A	B
Son	A1	B1	A2	B1
Soff	A1	B2	A1	B1
Snc	A2	B2	A2	B2
	Back		Boost	

Tableau (III.8) : Tableau des matrices

Avec :

$$A_1 = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix}; \qquad B_1 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 \\ 0 & \frac{-1}{RC} \end{bmatrix}; \qquad B_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

La matrice C sera adaptée aux variables considérées comme sorties dans chaque application.

III.5 Modèle Hybride

En combinant le mode continu et discret de système, et on représentant les éléments du modèle hybride H = (Q , E , D , F , G) .

Pour le cas de convertisseur Buck, nous avons déjà dit que on a trois états (Son, Soff, et Snc), donc on définit Q = {1, 2, 3}, qui sont les nombres qui indiquent les états, et E = {(1, 2),(2, 1),(2, 3),(3, 1)}, qui est l'ensemble d'évènements.

A l'origine nous avons deux variables d'état $x_1 = i_L(t)\ et\ x_2 = V_c\ (t)$ et nous allons ajouter au système une troisième équation différentielle $\dot{x}_3(t) = 1$ et sa solution est $x_3(t) = t$ avec : $x_3(0) = 0\ et\ \ t \geq 0$ donc x_3 représente exactement le temps.

Les champs vectoriels qui décrivant la dynamique continue sont obtenus à partir du tableau (III.8) :

$$F(t) = \begin{bmatrix} F_1(t) \\ F_2(t) \\ F_3(t) \end{bmatrix} \qquad (III.19)$$

Avec :
$$F_i(t) = \begin{bmatrix} A_{ij} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} B_{ij} \\ 0 \end{bmatrix} U(t), \qquad i \in Q \qquad (III.20)$$

« i » indique la ligne dans le tableau et « j=1 » indique la colonne 1 dans le même tableau pour le convertisseur Buck.

La dynamique discrète de système est décrit comme suit :

$$G(1,2) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in R^3 : x_3 = T_{on} + T_s Z \right\}$$

$$G(2,1) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in R^3 : x_3 = T_s Z \right\}$$

$$G(2,3) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in R^3 : x_1 = 0 \right\}$$

$$G(3,1) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in R^3 : x_3 = T_s Z \right\}$$

Pour le Boost convertisseur, il est pratiquement égal au Buck. La seule différence est les champs vectoriels qui sont obtenu à partir du tableau (III.8) avec j=2

III.6 Simulation

Stateflow est un outil graphique interactif intégré à SIMULINK pour modéliser et simuler des machines d'état fini, des systèmes qui réagissent à des événements, dits systèmes réactifs. Ces systèmes passent d'un état à un autre en réponse à des événements et des conditions.

III.6.1 Schéma global

La première étape de la modélisation consiste à saisir le schéma global sous Simulink. Pour cela nous ouvrons un modèle Simulink et on positionne dans la fenêtre (untitled) ainsi disponible un sous-système (subsystem) et un chart pris dans la bibliothèque de simulink.

Dans la figure suivante, nous pouvons voir qu'il y a un bloc de Statflow nommé **chart** et les autres sont des blocs de simulink.

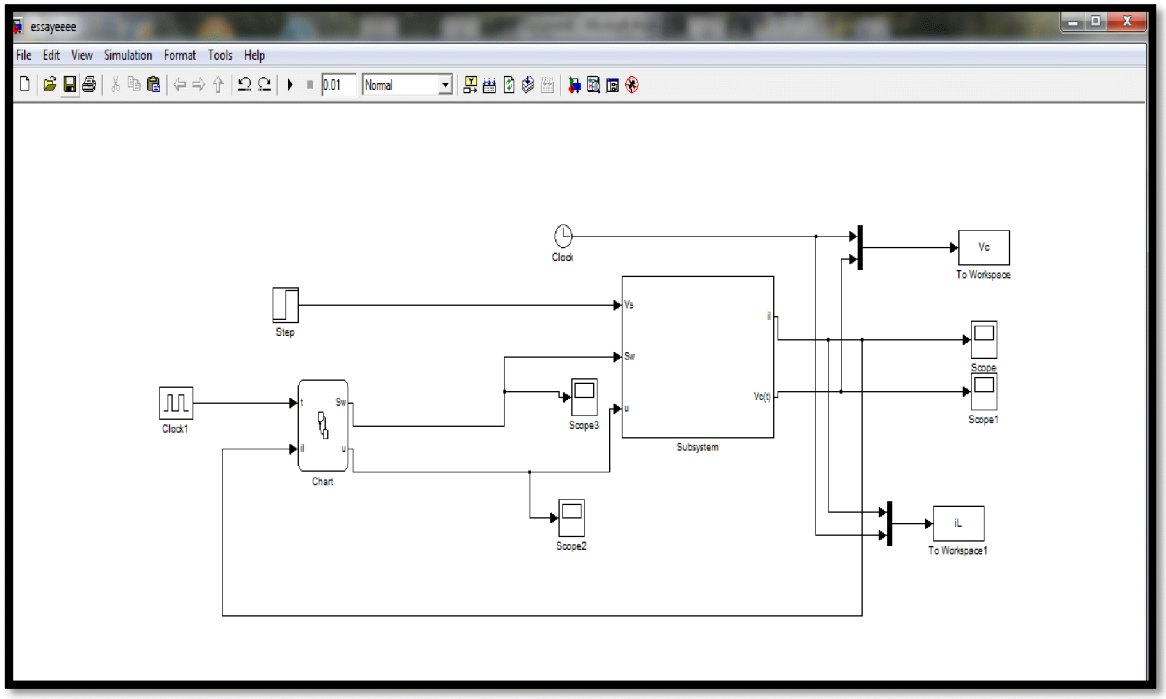


Figure (III.9) : Le diagramme hybride dans Simulink

- Le bloc chart de diagramme met en application un statechart avec 3 états, qui représentent les états de Q= {1.2.3} de SDH.
- L'ensemble d'évènements E= {(1.2), (2.1), (2.3), (3.1)} de SDH sont représentés par des transitions de statechart (flèches).

- Afin de choisir les matrices appropriées par Simulink, nous avons employé un mécanisme de commutation composé par les commutateurs A et B ce qui sont conduits par les sorties Sw et u de statechart.
- les gardes G sont représentés par des états et elles sont dénotées en écrivant entre crochets.

On peut voir toutes les conditions sur les fiqures (III.11) et (III.14) indiquant le diagramme Stateflow de convertisseur Buck et Boost.

III.6.2 Le Statechart de convertisseur Buck

Le passage d'un état à un autre est conditionné par le temps et le courant « i_L ». Nous pouvons alors construire l'automate à état fini suivant :

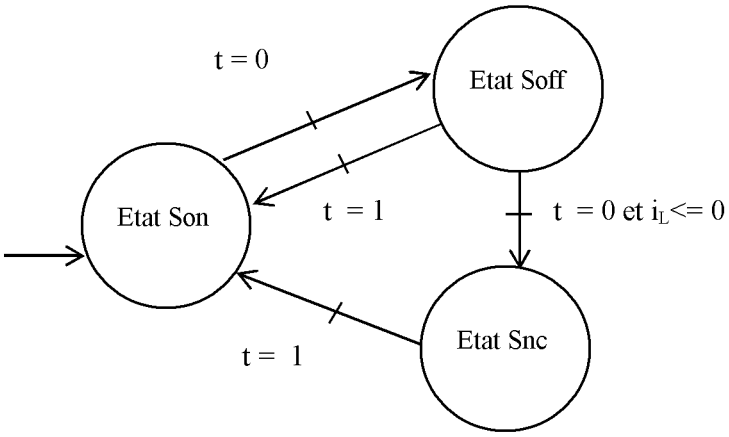


Figure (III.10) : Automate à état finis du système discret

Nous devons maintenant saisir le bloc **automate** sous l'environnement Stateflow. Double clic sur le bloc automate, on sera dirigé vers l'éditeur de diagramme Stateflow, puis nous plaçons tous les états et toutes transitions nécessaires.

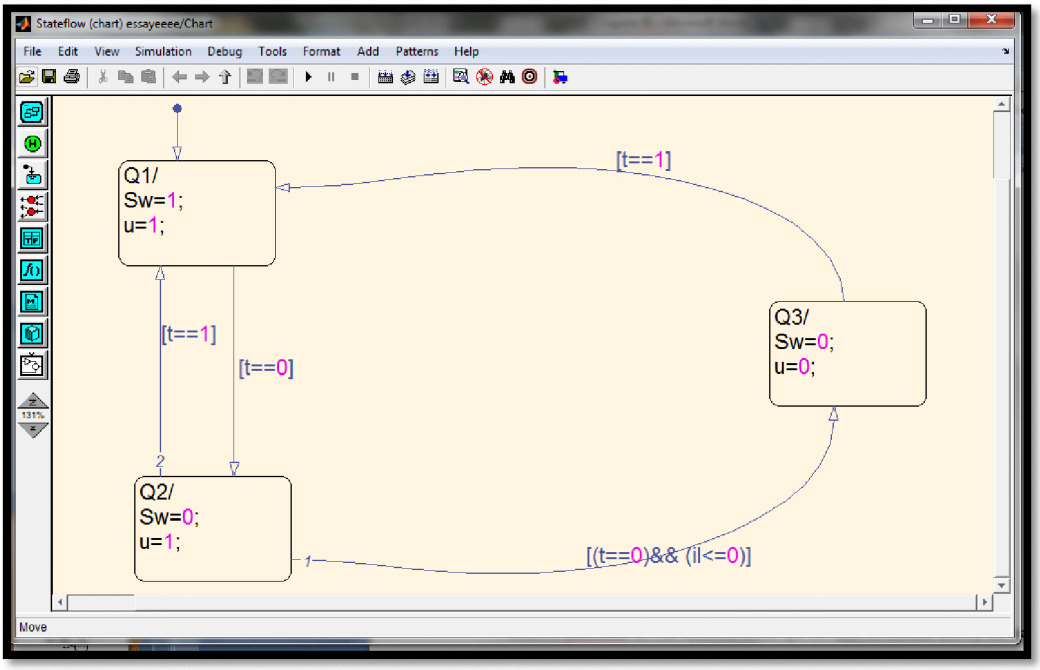


Figure (III.11) : Fenêtre d'adition du diagramme Statflow pour le Buck

Dans le statechart du convertisseur Buck qui est dans la figure (III.9), La machine d'état met en application le signale d'horloge (t), l'essentiel de l'exploitation du système est basé sur l'événement « **t** ».

Quand la simulation commence, le système va déclarer l'état Q1=Son, parce que c'est l'état dirigé par la transition de commencement (●⟶). En entrant dans cet état, le système est toujours dans le même état, t = Ton , et met les sorties à Simulink, Sw=1, u=1 et restes dans cet état jusqu'à la condition [t = 0] (t = Toff) de la transition qui affirme l'état Q2 = Soff.

Quand le système entre dans état Soff, il passe immédiatement à Sw=0 et u=1, puis les valeurs des matrices changent et la simulation de système continue avec $A=A_1$ et $B=B_2$. Pendant cet état, dû à la présence de la diode, le courant i_L doit être toujours positif.

Dans cet état, le système peut sauter aux autres deux états : à l'état Son et à l'état Snc (état de non-conduction de la diode). Afin de sauter de état Soff à l'état Snc, il est nécessaire que le temps passé dans cet état soit plus grand que le temps Toff et le courant i_L doit être négatif ou zéro.

Dans l'état Snc les variables Sw et u sont tous les deux égale mise à 0, ainsi les matrices A_2 et B_2 sont choisies. Dans cet état le courant i_L est nul, et le courant de charge est fourni par le condensateur C.

III.6.2.1 Exécution et résultats graphiques de Buck

Dans la fenêtre de Matlab, nous présentons les données suivantes :

>> Ts=1e-3 s; ton=Ts/2; Vs=20 v ;

>> R=5 Ω ; L=5e-3 H ; C=47e-6 F;

>> A1= [0 -1/L ; 1/C -1/(R*C)] ;

>> A2= [0 0 ; 0 -1/(R*C)] ;

>> B1= [1/L ; 0] ; B2= [0 ; 0] ;

La figure (III.12) suivante montre le mode continu de conduction du convertisseur Buck.

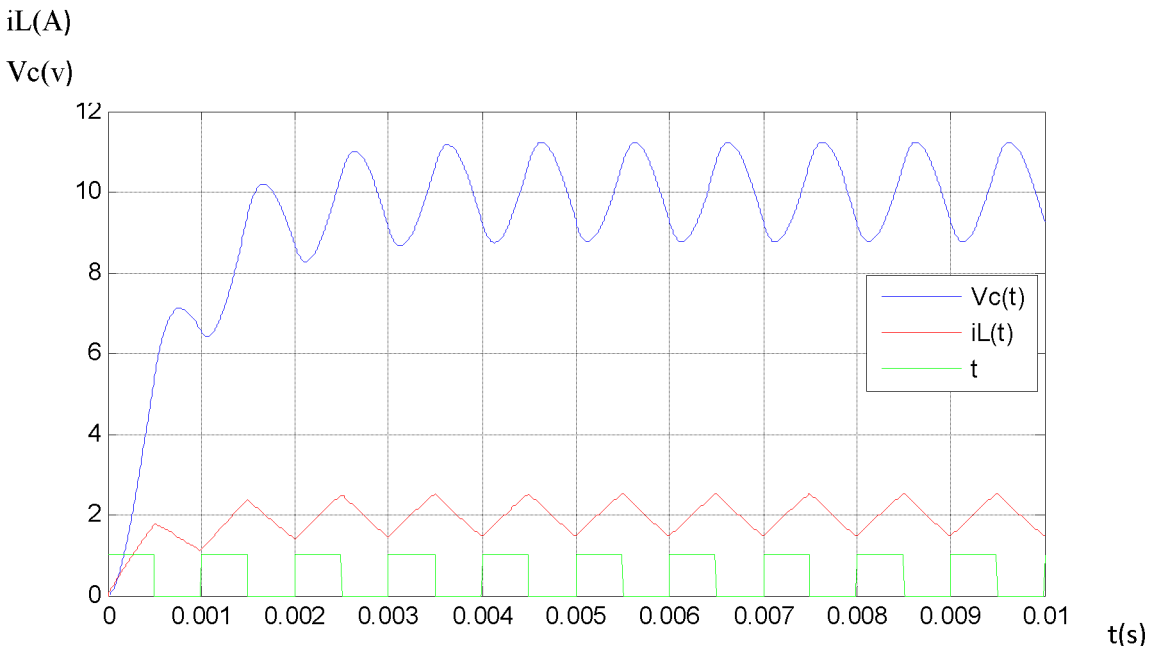


Figure (III.12) : iL(t) et Vc(t) de convertisseur Buck dans le mode CCM

Si on change la valeur de la résistance R on aura une nouvelle exécution.

>> R=25 Ω; donnant comme résultat les graphiques qui sont montrées dans la figure (III.13), qui montre le mode de conduction discontinu du convertisseur Buck.

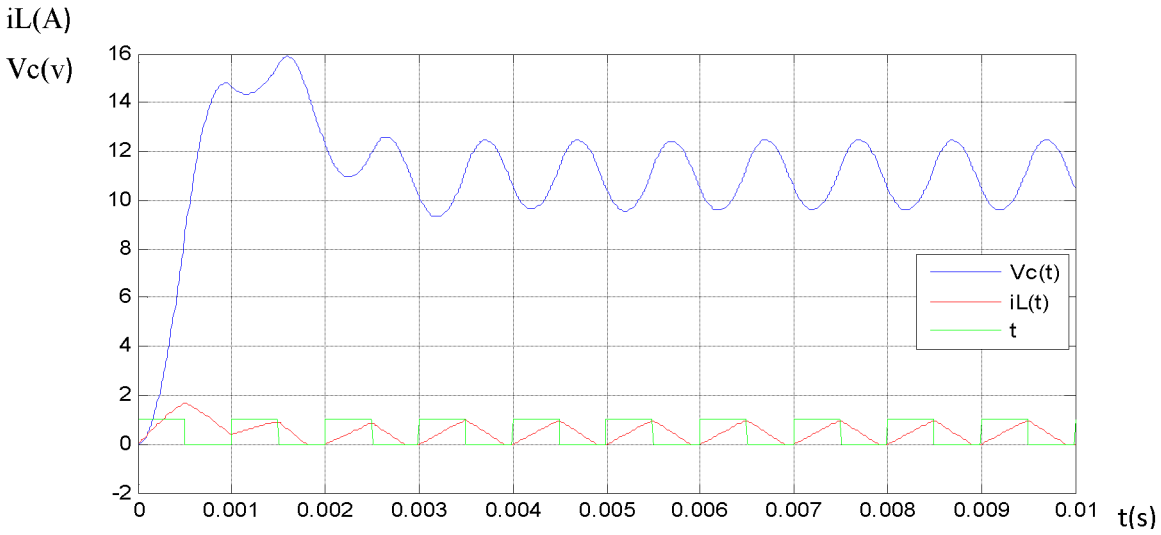


Figure (III.13) : iL(t) et Vc(t) de convertisseur Buck dans le mode DCM

III.6.3 Le Statchart de convertisseur Boost

Le statechart de ce convertisseur est semblable au statechart du Buck. Les seules différences sont les valeurs assignées aux Sw et u de Simulink, qui sont employés pour changer les valeurs des matrices A et B. (Voir la Figure suivante).

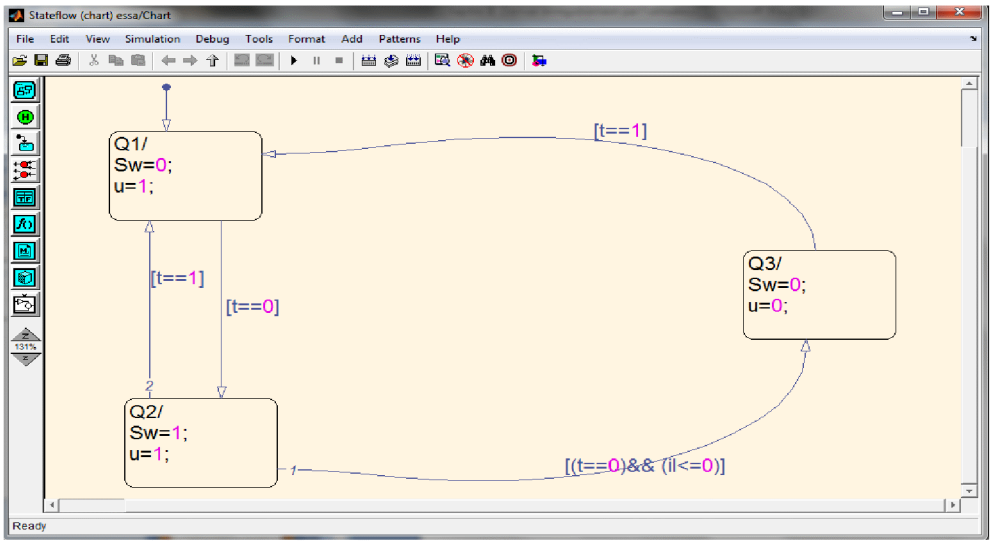


Figure (III.14) : Fenêtre d'adition du diagramme Statflow pour le Boost

III.6.3.1 Exécution et résultats graphiques de Boost

Avec les mêmes paramètres que nous avons employés avant pour l'algorithme et dans la fenêtre de Matlab, nous présentons les données suivantes :

>>Ts=5e-4 s ; ton=1.2e-4 s ; Vs=20 v ;

>> R=4 Ω ; L=250e-6 H ; C=100e-6 f ;

>> A1= [0 -1/L ; 1/C -1/(R*C)] ;

>> A2= [0 0 ; 0 -1/(R*C)] ;

>> B1= [1/L ; 0] ; B2= [0 ; 0] ;

Après le lancement de la simulation, les graphiques de variables $i_L(t)$ et Vc(t) apparaissent dans la fenêtre des figures.

La figure (III.15) montre le mode de conduction continu du convertisseur Boost.

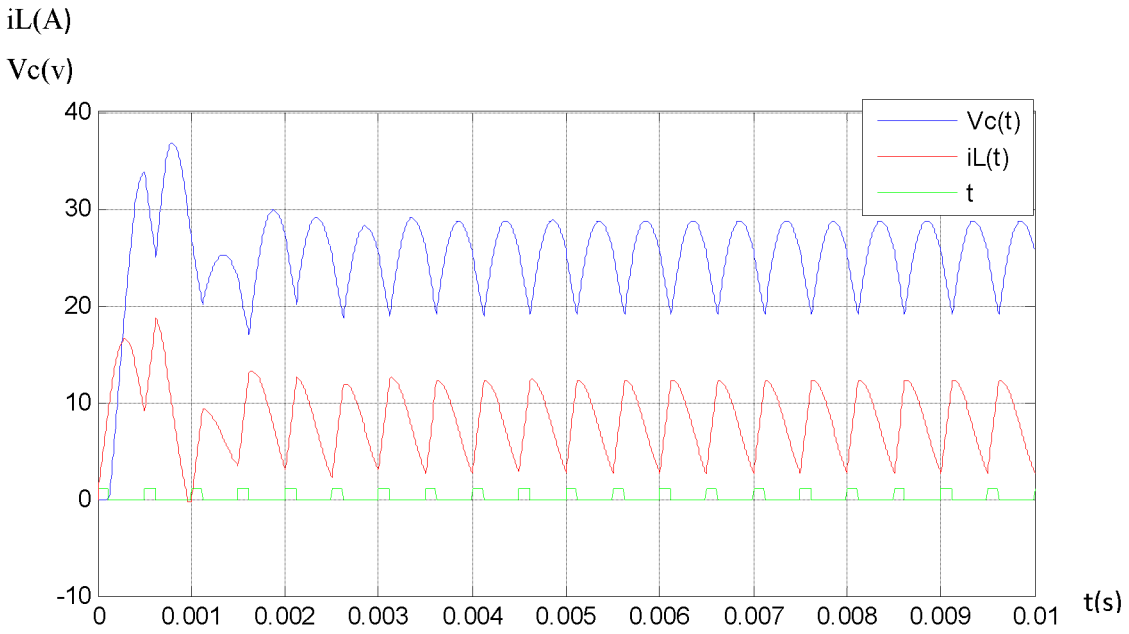


Figure (III.15) : iL(t) et Vc(t) de convertisseur Boost dans le mode CCM

Si on change la valeur de la résistance R on aura une nouvelle exécution.

>> R=10 Ω; donnant comme résultat les graphiques qui sont montrés dans la figure (III.16), qui montrent le mode de conduction discontinu du convertisseur Boost.

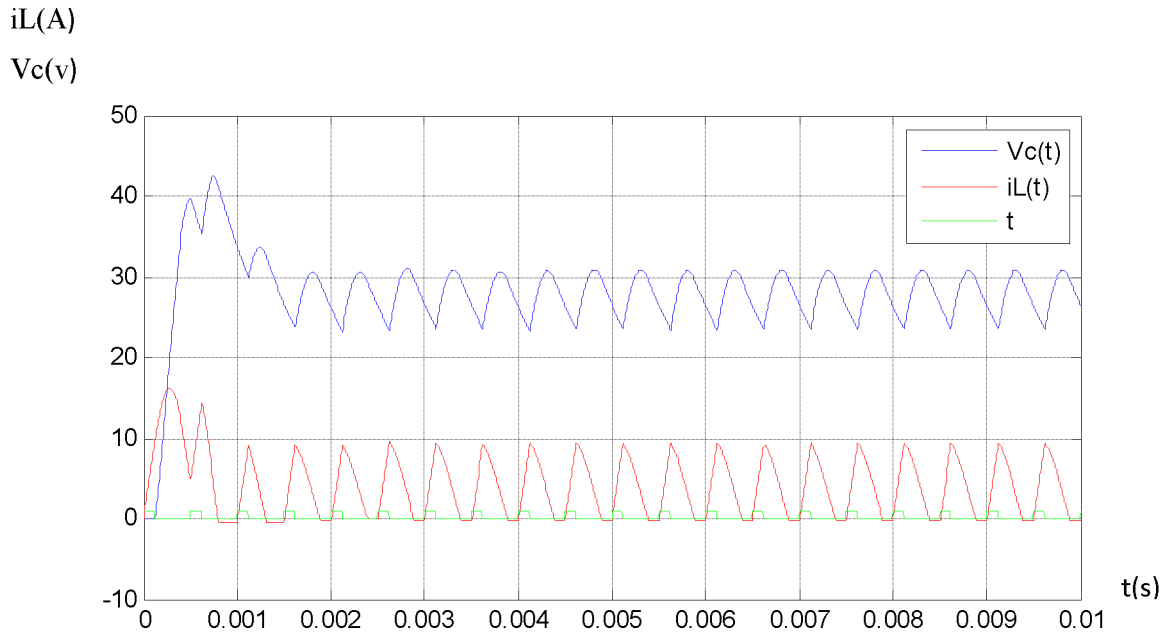


Figure (III.16) : iL(t) et Vc(t) de convertisseur Boost dans le mode DCM

III.7 Conclusion

En fait, le commutateur qui apparaît dans les circuits des deux convertisseurs est mis en application par un transistor, c'est la raison du courant pour laquelle le courant traversant, peut circuler dans une seule direction. Ce fait, on a pu envisager d'ajouter facilement une nouvelle garde G=(1,3), dans le convertisseur Boost, cette garde serait représentée dans le statechart par une transition associée avec l'état [il<=0 && t==0].

Nous avons considéré dans ce mémoire des systèmes hybrides définit comme des systèmes dynamiques continus en interaction avec des systèmes à événement discrets. Nous nous somme intéressé à la modélisation et la simulation de ces systèmes à l'aide de MATLAB/ Simulink/Statflow.

Dans le premier lieu, nous avons mentionné quelques exemples des systèmes hybrides, les plus rencontrés dans la littérature. Nous avons également proposé une représentation mathématique de ces systèmes. Par la suite, nous avons dressé un panorama des diverses modélisations des systèmes hybrides proposées dans les littératures. La représentation du système sous forme d'un automate hybride ou d'un réseau de petri hybride ou d'une structure modulaire permet d'avoir une vue plus fonctionnelle du SDH. Ainsi l'automate hybride et le réseau de petri hybride ont l'avantage de mettre en évidence les modes de fonctionnement du système ainsi que le passage d'un mode de fonctionnement a un autre, la structure modulaire quant à elle permet de mieux visualiser les interactions entre un système continu et le système à évènement discrets. Nous avons aussi distingué deux types de représentation à savoir une représentation analytique et une autre graphique. Les modèles à interface qui sont essentiellement basé sur la représentation analytique et utilisant une interface pour relier la partie continue à la patrie discrète. Nous avons présente deux types de modèle à interface très répandus en littérature et qui sont : le modèle d'Antsaklis et le modèle de brokett. La synthèse de la bibliographie disponible nous permis de représenter un modèle à interface généralisé capable de décrire les phénomènes hybrides.

Comme exemple d'application nous, avons cité le convertisseur buck et Boost. La méthode de modélisation hybride a été donnée afin de permettre la représentation du comportement dynamique des convertisseurs DC-DC dans les deux modes de fonctionnement (conductions continue et discontinue). En effet des variables discrètes (état des interrupteurs et diode) et des variables continus (courant et tension) apparaissent simultanément.

On a traité, à l'aide de Simulink/Stateflow, le problème de la simulation de ces convertisseurs continu-continu. La démarche que nous avons suivie s'appuie sur l'utilisation d'un automate hybride. On a analysé les interactions entre les différentes grandeurs du système, en mettant en évidence les liens entre la dynamique continue et la dynamique discrète. Cette méthode de modélisation hybride permettant d'obtenir un modèle hybride

commun afin de caractériser le comportement d'un convertisseur dans les deux modes de fonctionnement (MCC/MDC).

Dans la continuité du travail présenté dans ce mémoire, nous pourrions dans de futurs travaux en s'intéresser à :

- L'application de la méthode de modélisation hybride sur d'autres convertisseurs DC-DC d'ordre élevé (Cuk, SEPIC) en MCC pourrait être envisagée.
- La technique de modélisation pourrait être appliquée sur d'autres types de systèmes de conversion d'énergie (par exemple des onduleurs).
- Développer un modèle hybride caractérisant des transitions entre les deux modes de fonctionnement (MCC/MDC) pour ces convertisseurs avec un comportement fortement non-linéaire.
- Développer des méthodes de modélisation hybride afin de procéder à l'analyse et à l'élaboration de la loi de commande des convertisseurs de puissance DC-DC.
- Développement de logiciels de simulation des systèmes hybrides.

Bibliographique

[1] D. DUMUR, 2003 : « évolution du formalisme MLD en vue d'une optimisation discrète des SDHs, thèse de doctorat. 2003

[2] C.VLAD, 2013 : « Commande prédictive des systèmes hybrides et application à la commande de systèmes en électronique de puissance » ; thèse de doctorat. Mars 2013.

[3] B.CEBRON, 2000 : « Commande de systèmes dynamiques hybrides », thèse de doctorat, Université d'Angers, décembre 2000.

[4] K.TAARIT, 2010 : « Contribution à l'identification des systèmes à retards et d'une classe de systèmes hybrides » ; thèse de doctorat décembre 2010.

[5] A.GIRARD, 2004 : « Analyse Algorithmique des Systèmes Hybrides », thèse de doctorat septembre 2004

[6]S.MOKHTAR, L.SAROUI, 2007 : « modélisation, analyse, stabilité et commande des système dynamique hybrides», mémoire d'ingéniorat, université Mouloud Mammeri, 2007

[7] B.SEDGHI, 2003 : « control design of hybrid systems via dehybridation», thèse de doctorat, école polytechnique Fédérale de Lausanne, 2003.

[8] L.GHOMRI, 2012 : « Synthèse de contrôleur de systèmes hybrides à flux continu par réseaux de Petri hybrides », thèse de doctorat, Université Abou-BEKR BELKAÏD – Tlemcen,
mars 2012.

[9] Rapport de TIPE 2005/2006: Ludivine FRANCART Pierre THIROUINE : « Les systèmes hybrides »

[10] J.BEN SALAH, 2009 : « Analyse et commande des systèmes non linéaires complexes, Application aux systèmes dynamiques à commutation » , thèse de doctorat, Université

CLAUDE BERNARD - LYON 1, Mars 2009

[11] **Hadyan FIBRIANTO 2001** : « Contribution a la commande du procede de decongelation haute pression : approches classiques et approche hybride », thèse de doctorat, Ecole Doctorale Electronique, Electrotechnique, Automatique, Télécommunication, Signal, Novembre 2001

[12] J.ZAYTOO, 2001 : Les Systèmes Dynamiques Hybrides. 3e Conférence Internationale sur l'Automatisation des Processus Mixtes ADPM'98, Reims (19-20 mars 1998).