

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERRI de Tizi-Ouzou



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

PROJET DE FIN D'ETUDES

En vue de l'obtention du diplôme

D'INGENIEUR D'ETAT EN AUTOMATIQUE

Thème

**Conception d'un contrôleur flou pour la
commande d'un système non-linéaire**

Présenté par :

M^r : ZAIDI Cherif

M^r : CHIKHAOUI Mustapha

Proposé et dirigé par :

M^r : SAIDI.kh

Soutenu le : 11/10/2011.

Promotion 2011

Remerciements

Avant tout, nous remercions DIEU le Tout-puissant de nous avoir donné le courage, la volonté et la patience durant toutes ces années d'études et que grâce à lui ce travail a pu être réalisé.

Nos vifs remerciements vont en premier lieu, à nos chers parents de nous avoir aidé pour arriver au terme de ce travail.

A l'issue de ce travail, nous tenons à exprimer nos remerciements et gratitude à notre promoteur Mr : Kh. SAIDI pour son aide et son orientation judicieuse.

Notre profonde gratitude va également aux membres de jury qui fera l'honneur d'évaluer ce travail.

A travers ce mémoire, nous adressons nos reconnaissances à tous nos enseignants qui ont contribué à notre formation depuis la première classe du primaire jusqu'à aujourd'hui.

Nous tenons à exprimer toute notre gratitude à l'ensemble des personnes qui ont participées, chacune à sa manière, à l'accomplissement de ce travail.

A tous merci

Dédicaces

Je dédie ce modeste travail à :

A mon très cher père ; symbole de sacrifices, ma très chère maman ;

Symbole de tendresse et d'amour

A mes très chères sœurs

A mes très chers frères

A mes grands parents

A tous mes amis en particulier : Mourad-Lamraoui, Sofiane-Idir, Slimane-Amoukrane, samir, Haddadi-idir, Mourad-Titanic, Hamza-Hamri, elhadj, et omar, sabha, safia, Moussa-Yahiaoui, mouh.etc.

Sommaire

Introduction générale.....	1
Chapitre I : La logique floue et la commande	
I.1. Introduction.....	2
I.2. La logique floue aujourd’hui.....	2
I.3. Historique de la logique floue.....	2
I.4. Théorie de la logique floue.....	3
I.5. Les concepts.....	5
I.5.1. Sous-ensemble floue.....	5
I.5.2. Variable linguistique.....	6
I.5.3. Opérateurs de la logique flou.....	7
I.5.4. Fonction d’appartenance.....	8
I.5.5. Propositions et règle floue.....	9
I.6. Commande par logique floue.....	10
I.6.1. Principe et réalisation d’un contrôleur flou.....	12
I.6.1.1. Procédé général lors de la conception d’une commande par la logique floue.....	12
I.6.1.2. Propriétés d’une commande par logique floue	12
I.6.2. Les différentes étapes de la commande floue.....	13
I.6.2.1. Base de connaissances.....	14
I.6.2.2. Mise en forme des entrées, normalisation.....	15
I.6.2.3. Interface de fuzzification.....	15
I.6.2.4. Traitement des prémisses composées.....	17
I.6.2.5. Moteur d’interface floue.....	17
I.6.2.6. Interface de défuzzification.....	20
I.6.2.7. Mise en forme des sorties, dénormalisation.....	21
I.6.3. Structure de base d’un contrôleur flou : analogie structurelle avec les P.I.D.....	21
I.6.4. Réglage, stabilité et robustesse d’un contrôleur flou.....	22

Sommaire

I.6.5. Différents types de régulateurs flous.....	24
I.6.5.1. Régulateur de type Mamdani.....	24
I.6.5.2. Régulateur de type Sugeno.....	25
I.6.6. Avantages et inconvénients de la commande par la logique floue.....	25
I.7. Conclusion.....	26
Chapitre II : Modélisation et commande classique des robots	
II.1. Introduction.....	27
II.2. Définition générale.....	27
II.2.1. Difficultés.....	28
II.2.2. Architecture des robots.....	28
II.3. Modélisation géométrique.....	28
II.3.1. Description de la géométrie des robots à la structure ouverte simple.....	28
II.3.1.1. Les paramètres de Danavit-Hardenberg.....	29
II.3.2. Modèle géométrique direct.....	32
II.3.3. Modèle géométrique inverse.....	32
II.4. Modélisation cinématique.....	33
II.4.1. Modèle cinématique direct.....	33
II.4.2. Modèle cinématique inverse.....	34
II.5. Modélisation dynamique.....	36
II.5.1. Formalisme de Lagrange.....	37
II.5.1.1. Forme générale des équations dynamiques.....	38
II.5.1.2. Calcul de l'énergie cinétique et potentielle.....	39
II.5.2. Formalisme de Newton-Euler.....	39
II.5.2.1. Equations de Newton-Euler linéaires par rapport aux paramètres inertiels.....	40
II.5.2.2. Forme parasitique des équations Newton-Euler.....	40
II.6. Aperçu sur la modélisation dynamique des robots manipulateurs à chaine ouverte simple.....	41

Sommaire

II.6.1. Modélisation du robot à deux degré de liberté	42
II.6.1.1. Calcul des matrices de passages.....	42
II.6.1. 2. Calcul des tenseurs d'inertie.....	43
II.6.1.3 .Calcul de la contribution du corps C_1 à la matrice d'inertie.....	43
II.6.1.4. Calcul de la contribution du corps C_2 à la matrice d'inertie.....	44
II.6.1.5. Calcul de la matrice d'inertie du robot.....	44
II.6.1. 6. Calcul de la matrice de Coriolis du robot.....	44
II.6.1.7. Calcul de l'énergie potentielle totale.....	45
II.6.1. 8 .Calcul du vecteur des forces de gravité	46
II. 6.2. Modèle dynamique du robot à 2 d d l.....	46
II.7. Quelques lois de commande classique des robots manipulateurs.....	46
II.7.1. Loi de commande proportionnelle dérivée avec compensation de l'effet de gravité (Point à point).....	47
II.7.2. Commande en poursuite de trajectoire.....	48
II.7.2.1. Générateur de trajectoire.....	48
II.7.2.1.1. Interpolation polynomiale.....	48
II.7.2.1.2. Interpolation Bang-Bang.....	52
II.7.3. Loi de commande du couple calculé (computed torque control).....	53
II.8. Exemples de simulation.....	54
II.8.1. Bras manipulateur à 2ddl.....	54
II.8.1.1. Application de la loi de commande point à point.....	55
II.8.1.2. Application de la loi de commande du couple calculé.....	55
II.9. conclusion.....	57
Chapitre III : Application de la commande floue à une pendule inversé	
III.1. Introduction.....	59
III.2. Présentation de pendule inversé.....	59
III.3. La commande par logique floue de pendule inversé.....	50

Sommaire

III.4. Conception du contrôleur flou pour la commande du pendule inversé.....	62
III.4.1. Résultats de simulation.....	65
III.5.Conclusion.....	68
Chapitre IV : La réalité virtuelle	
VI.1.Introduction à la réalité virtuelle.....	69
VI.2.Démonstration sur un bras manipulateur à 2ddl	69
VI.3.Conclusion.....	73
Conclusion générale	74

Introduction générale

Parmi les récents développements des techniques pour l'Automatique, l'introduction de nouveaux formalismes tels que la logique floue, a suscité un intérêt sans cesse croissant depuis les années soixante-dix. Il suffit de voir les nombreuses applications industrielles qui en découlent et de consulter l'abondante littérature sur le sujet pour s'en convaincre.

L'intérêt de la logique floue réside dans sa capacité de traiter l'imprécis, l'incertain et le vague. Elle est issue de la capacité de l'homme à décider et agir de façon pertinente malgré le flou des connaissances disponibles a été introduit dans le but d'approcher le raisonnement humain à l'aide d'une représentation adéquate des connaissances. Aussi, le succès de la commande de la logique floue trouve en grande partie son origine dans sa capacité de traduire une stratégie de contrôle d'un opérateur qualifié en un ensemble de règles linguistiques "si ... alors...." facilement interprétable.

L'utilisation de la commande floue est particulièrement intéressante lorsqu'on ne se dispose pas de modèle mathématique précis du processus à commander ou lorsque ce dernier présente de trop fortes non linéarité ou imprécisions. Cependant, beaucoup de travaux relatifs à la commande de systèmes mono variables ont été reportés dans la littérature, on ne trouve, en comparaison, que très peu d'articles sur la commande floue multi variable. Ceci n'est guère surprenant étant donné que le nombre de règles est une fonction exponentielle des variables en entrée du contrôleur flou.

Ce mémoire est constitué de quatre chapitres articulés comme suit :

Le premier chapitre introduit les concepts de base nécessaires à la bonne compréhension de la logique floue et pour cela ce chapitre est principalement consacré à la théorie des ensembles flous et le principe de la commande floue est ensuite abordé en présentant les différents types de régulateurs flous et en explicitant les éléments constitutifs de chaque type.

Le deuxième chapitre traitera la modélisation des bras manipulateurs à structure ouverte simple. Ainsi que l'application de quelques lois de commande classique et des simulations sur un bras manipulateur à deux degré de liberté (2ddl).

En premier temps nous avons essayé de faire la conception d'un contrôleur flou pour la commande d'un bras manipulateur à 2ddl, malheureusement, on n'a pas pu trouver de bon résultats, pour cela le troisième chapitre sera consacré à l'application de la commande floue à un autre système non linéaire à savoir le pendule inversé.

En fin dans le quatrième chapitre on va essayer de montrer une animation avec la réalité virtuelle qui nous donne le comportement du bras pendant la simulation.

I.1.Introduction

L'objectif de ce chapitre est de présenter les notions principales de théorie de la logique floue, ainsi que la commande floue. D'abord, nous présentons quelques concepts de base de la logique floue ainsi que leurs applications pour la synthèse d'une commande floue. Les éléments de base d'un régulateur par logique floue, les différents types de régulateurs flous, ainsi que leurs avantages et leurs inconvénients seront donnés.

I.2.La logique floue aujourd'hui

La logique floue, dans la plupart des applications, permet de prendre en compte toutes sortes de connaissances qualitatives des concepteurs et de l'opérateur dans l'automatisation des systèmes. Elle suscite un intérêt médiatique depuis les années 90. Les nombreuses applications dans l'électroménager et l'électronique grand public réalisées notamment en Japon en ont été l'élément déclenchant. Machines à laver sans réglage, caméscopes anti-bougé et de nombreuses autres innovations ont fait connaître le terme « LOGIQUE FLOUE » à un large public. Dans le domaine l'automobile les transmissions automatiques, les contrôles d'injection et d'ante-cliquetis, l'air conditionné sont réalisées sur des véhicules de série grâce à la logique floue. Dans le domaine de processus de la production, continue et par lots, et dans les automatismes (qui nous intéresse essentiellement ici) les applications se sont également multipliées. La logique floue s'y développe car il s'agit d'une approche essentiellement pragmatique et efficace. On dit parfois qu'elle permet de systématiser ce qui est du domaine de l'empirisme, et donc difficile à maîtriser. La théorie des ensembles flous forment une méthode pertinente et facilement réalisable dans les applications temps réel; elle permet de transcrire et rendre dynamique les connaissances des concepteurs ou des opérateurs.

Cet aspect adaptable et universel de la logique floue permet de s'attaquer à l'automatisation des procédures telles que la mise en route, le réglage des paramètres pour lesquelles peu d'approches existaient auparavant.

I.3.Historique de la logique floue

❖ Apparition de la logique floue

Le terme d'ensemble flou apparaît pour la première fois en 1965 lorsque le professeur Lotfi A.Zedah, de l'université de Berkeley aux USA publie un article intitulé «Ensembles flous» (Fuzzy sets). Il a réalisé depuis de nombreuses avancées théoriques majeures dans le domaine et a été rapidement accompagné par de nombreux chercheurs développant des travaux théoriques.

❖ Premières applications

Parallèlement, certains chercheurs se sont penchés sur la résolution par la logique floue de problèmes réputés difficiles. Ainsi en 1975, le professeur Mamdani à Londres développe une stratégie pour le contrôle des procédés et présente des résultats très encourageantes qu'il a obtenus sur la conduite d'un moteur à vapeur. En 1978, la société Danoise F.L.Smidth réalise le contrôle d'un four à ciment. C'est la première véritable application industrielle de la logique floue.

❖ Essor

C'est au Japon, où la recherche n'est pas seulement théorique mais également très applicative, que la logique floue connaît son véritable essor. A la fin des années 1980, c'est un véritable boum qu'il fait parler. Les produits grand public, machines à laver, appareils photographiques et autres caméscopes estampillés «fuzzy logic» ne se comptent plus.

Dans l'industrie, le traitement des eaux, les grues portuaires, les métros, les systèmes de ventilations et de climatisation sont touchés. En fin des applications existent dans des domaines très différents tels que la finance ou le diagnostic médical.

A partir de 1992, c'est en Allemagne que des applications apparaissent en grand nombre ainsi qu'à une moindre échelle aux USA.

I.4. Théorie de la logique floue

Les compréhensions de l'univers dans lequel nous évoluons sont généralement imparfaites dans la mesure où elles peuvent être entachées d'incertitudes et/ou d'imprécisions, ne serait-ce qu'à travers la perception que nous en avons. Or, nous pouvons constater que l'homme intègre naturellement ces imperfections dans la vie de tous les jours, en particulier au niveau de raisonnement et de la décision. L'idée du professeur Lotfi A.Zadeh [Zadeh, 65], en 1965, à travers le nouveau concept ensembliste d'appartenance graduelle d'un élément à un ensemble, a été de définir une logique multi évaluée permettant de modéliser ces imperfections c'est à dire prendre en compte les états intermédiaires entre le tout ou rien. L'utilité de cette approche peut être illustrée de la manière suivante :

Une température de 10°C, pour un humain, est généralement considérée comme froide ; une autre de 40°C est qualifiée de chaude. Si chacune de ces valeurs appartient à une catégorie (ensemble) bien définie, qu'en est-il pour des valeurs intermédiaires ? Une réponse intuitive consiste à affirmer qu'elles appartiennent à une ou deux des catégories précédentes avec des niveaux ou des degrés (normalisés) différents. On évite ainsi des transitions brusques entre différentes catégories, comme cela est le cas en logique binaire (figure.1). Il semble en effet

surprenant de considérer qu'une température de 40°C est chaude, alors qu'une température de 39,9°C ne l'est pas.

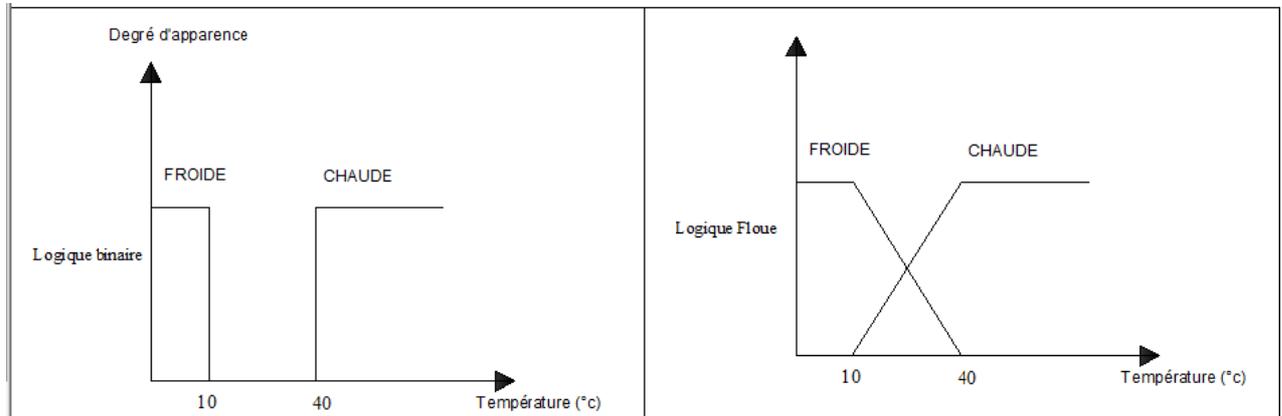


Figure I.1. Exemple de définition d'ensemble sur un univers de discours en logique binaire et en logique floue

Cet exemple permet d'illustrer le fait qu'une logique binaire classique soit, dans certains cas, trop limitative. Donc il est nécessaire de faire appel à une autre logique multi évaluée qui sera vue comme une extension de la précédente, c'est bien la logique floue.

En ce qui concerne la commande d'un processus quelconque, la logique floue permet une approche fondatrice par rapport à l'automatique classique. En automatique, en général, on s'attache à modéliser le processus à travers un certain nombre d'équations différentielles. Cette modélisation est rendue difficile et par fois impossible à mesurer que la complexité des processus à contrôler augmente. D'une manière radicalement opposée, un contrôleur va décrire non pas le processus mais la façon de le contrôler, tout comme le fait un expert humain à travers des règles intégrant naturellement imprécisions et incertitudes.

Un certain nombre d'applications utilisant la logique floue ont vu le jour ces dernières années. Les plus médiatisées sont certainement les réalisations des chercheurs et industriels japonais qui, depuis les années 80, se sont intéressés notamment au contrôle/commande de processus. Ces applications s'appuient sur les travaux de Mamdani [Mamdani, 74], [Mamdani et Assilian, 75] qui fut certainement le premier à avoir la potentialité de la théorie des sous-ensembles flous dans ce domaine. Nous pouvons trouver d'autres applications dans des domaines généraux, tels que le marketing les traitements médicaux, le domaine de traitement de l'image, le traitement de signal, la robotique, et autres.

I.5. Les concepts

Le concept de la théorie des sous-ensembles flous, s'appuie sur la notion de degré d'appartenance d'un élément à un sous-ensemble flou. Tandis que les ensembles traditionnels sont caractérisés par une fonction d'appartenance notée \mathbf{X} , (également appelée fonction caractéristique) définie sur $\{0,1\}$, les sous-ensembles flous sont, eux, caractérisés par une fonction d'appartenance notée μ défini sur $[0,1]$.

En d'autres termes, dans le langage ensembliste classique, un élément appartient ou n'appartient pas à un ensemble tandis qu'un élément appartient à un sous-ensemble flou avec un certain degré (éventuellement nul). En résumé, pour un sous-ensemble A défini sur un univers de discours U , on peut écrire :

A sous-ensemble classique : fonction caractéristique $X_A : U \rightarrow \{0,1\}$

A sous-ensemble flou : fonction d'appartenance $\mu_A : U \rightarrow [0,1]$

Par extension, ce nouveau concept définit une logique multi évaluée qui apparait comme une génération de la logique binaire. Nous allons maintenant définir un certain nombre de termes propres au domaine de la logique floue auxquels nous pourrons référer [Lee, I, 90], [Buhler, 94], [Passino, Yurkovich, 98].

I.5.1. Sous-ensemble flou

Nous venons de voir ce que l'on entend par sous-ensemble où, A sur un univers de discours U , est représenté dans la figure I.2 à travers sa fonction caractéristique μ_A . Il peut également être décrit par un certain nombre de caractéristiques comme :

- **Son support :**

$$\text{supp}(A) = \{x \in U / \mu_A(x) \neq 0\} \quad (\text{I.1})$$

- **Sa hauteur :**

$$\text{ht}(A) = \sup_{(x \in U)} \{\mu_A(x)\} \quad (\text{I.2})$$

- **Son noyau :**

$$\text{noy}(A) = \{x \in U / \mu_A(x) = 1\} \quad (\text{I.3})$$

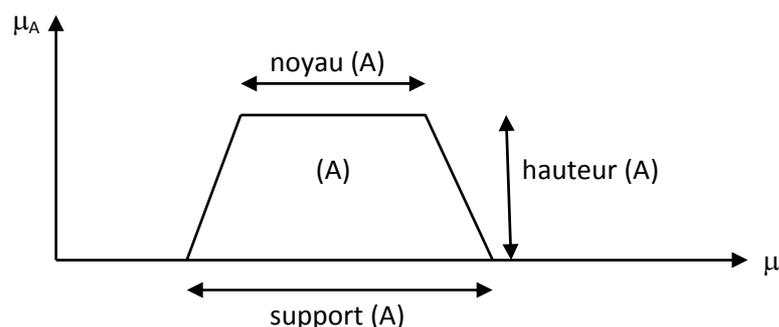


Figure I.2. Représentation d'un sous ensemble flou et principales caractéristiques

Le concept de sous-ensemble flou a été introduit pour éviter les passages brusques d'une classe à une autre et autoriser des éléments à n'appartenir complètement ni à l'une ni à l'autre ou encore appartenir partiellement à chacune.

La notion de sous-ensemble flou permet de traiter :

- des catégories aux limites mal définies (comme « centre ville » ou « ancien »),
- des situations intermédiaires entre le tout et le rien (« presque noir »),
- des valeurs approximatives (« environ 2 Km »),
- des classes en évitant l'utilisation arbitraire de limites rigides (il est difficile de dire qu'une maison située à 200 m de la plage en est proche, mais qu'à 210 m elle en est éloignée).

Remarque

Le terme « sous-ensemble » flou provient du fait que celui-ci est considéré comme une partie de l'univers de discours U . Dans la littérature, on peut trouver parfois ensemble flou, qui constitue un abus de langage.

I.5.2. Variable linguistique

Une variable linguistique sert à modéliser les connaissances imprécises ou vagues sur une variable dont la valeur précise peut être inconnue. Elle est définie par un triplet $(V; U; T_V)$ où V représente la variable classique (âge, température,..) définie sur l'univers de discours U . T_V est l'ensemble des instanciations possibles de la variable V : il s'agit de sous-ensembles flous repérés par leur label A_i : on écrit ainsi $T_V = \{A_1; A_2; K; A_n\}$.

Graphiquement, une variable linguistique peut être représentée comme dans la figure I.3

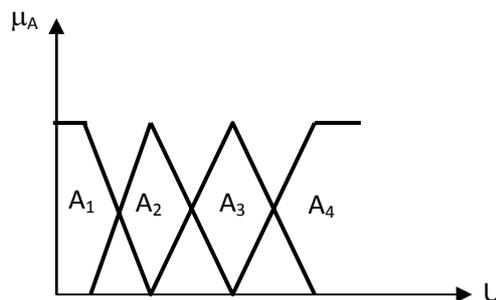


Figure I.3. Représentation d'une variable linguistique définie comme : $\{U, V, T_V = \{A_1, A_2, A_3, A_4\}\}$

Un sous-ensemble flou permet de représenter différentes notions, en particulier la spécificité et la précision. Ces différentes notions apparaissent dans la figure I.4 :

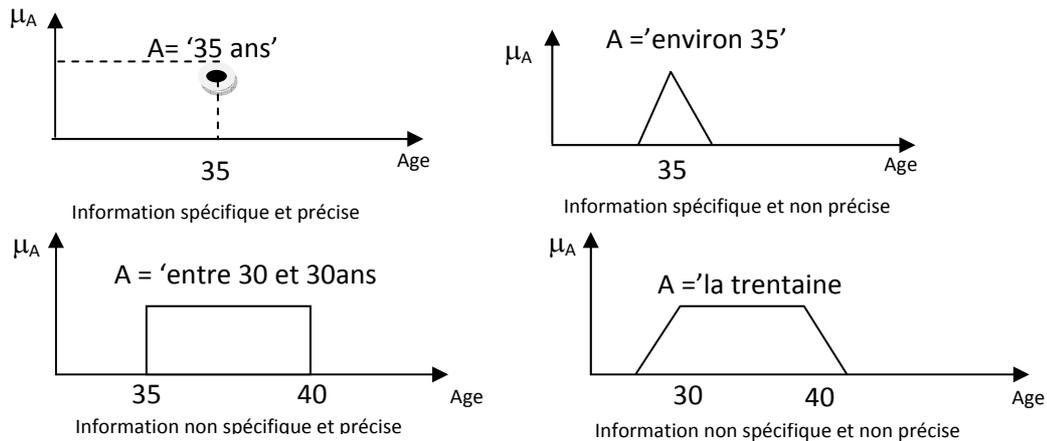


Figure I.4. Notions de spécificité et de précision représentées à l'aide de sous-ensembles flous

I.5.3. Opérateurs de la logique floue

Les variables linguistiques sont liées entre elles au niveau des inférences par des opérateurs ET ou OU sont réalisées respectivement par les règles « min » et « max ». Il s'agit d'opérateurs de la logique floue qui interviennent sur les fonctions d'appartenance représentant les variables linguistiques.

Le plus souvent, les opérateurs ET ou OU sont réalisés respectivement par les règles « min » et « max ». Il ya alors une certaine affinité avec les règles de la logique classique. Cependant, il existe un grand nombre de règles pour la réalisation des opérateurs ET et OU qui tiennent compte du caractère particulier de la logique floue.

Soit A et B deux ensembles flous définis dans le référentiel U avec les fonctions d'appartenance μ_A et μ_B respectivement :

Définition 1 : (Union)

La fonction d'appartenance $\mu_{A \cup B}$ de l'union $A \cup B$, est définie pour tout $u \in U$ par :

$$\mu_{A \cup B}(u) = \text{Max} \{ \mu_A(u), \mu_B(u) \} \quad (\text{I.4})$$

Dans le domaine de la commande, le + représente souvent la Max :

$$\mu_A(u) + \mu_B(u) = \max \{ \mu_A(u), \mu_B(u) \} \quad (\text{I.5})$$

Définition 2 : (intersection)

La fonction d'appartenance $\mu_{A \cap B}$ de l'intersection $A \cap B$, est définie pour tout $u \in U$ par :

$$\mu_{A \cap B}(u) = \min \{ \mu_A(u), \mu_B(u) \} \quad (\text{I.6})$$

Définition 3 : (Complémentation)

La fonction d'appartenance $\mu_{\bar{A}}$ du complément de l'ensemble A , notée \bar{A} , est définie pour tout $u \in U$ par :

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u) \quad (I.7)$$

Définition 4 : (produit cartésien)

Soit A_1, \dots, A_n des ensembles flous respectivement définis sur U_1, \dots, U_n , le produit cartésien de A_1, \dots, A_n , est un ensemble flou défini sur U_1, \dots, U_n , de fonction d'appartenance :

$$\begin{aligned} \mu_{A_1 \times \dots \times A_n}(u_1, u_2, \dots, u_n) &= \min(\mu_{A_1}(u_1), \dots, \mu_{A_n}(u_n)) \\ &= \mu_{A_1}(u_1) \times \mu_{A_2}(u_2) \times \dots \times \mu_{A_n}(u_n) \end{aligned} \quad (I.8)$$

Définition 5 : (relation floue)

Une relation floue représente le degré de présence, ou d'absence d'une association entre les éléments de deux ou plusieurs ensembles flous. Une relation floue d'ordre n est ensemble flou défini sur $U_1 \times \dots \times U_n$ par l'expression suivante :

$$R_{U_1 \times K \times U_n} = \{(u_1, K, u_n), U_R(u_1, K, u_n) \mid (u_1, K, u_n) \in U_1 \times K \times U_n\} \quad (I.9)$$

Définition 6 : (composition des règles floues)

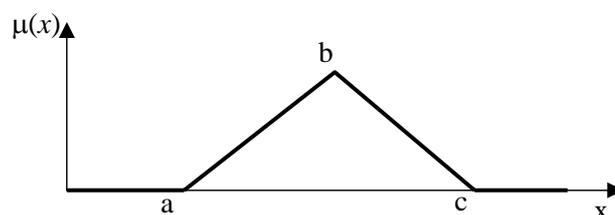
Soit R et S deux relations floues définies respectivement dans $U \times V$ et $V \times W$. La composition de R et S est un ensemble flou, symbolisé par $R \circ S$, de fonction d'appartenance :

$$\mu_{R \circ S}(u, w) = \{[(u, w), \text{SUP}(\mu_R(u, v) * \mu_S(v, w))], u \in U, v \in V, w \in W\} \quad (I.10)$$

I.5.4. Fonction d'appartenance

Afin de permettre un traitement numérique des variables linguistiques dans la prise de décisions floues sur ordinateur, une définition des variables linguistiques à l'aide de fonctions d'appartenance s'impose. Dans ce contexte on associe à chaque valeur de la variable linguistique une fonction d'appartenance désignée par $\mu_A(x)$, qui sera désignée par le degré ou le facteur d'appartenance. Il est à noter que l'ensemble des éléments de x pour lesquels $\mu_A(x) > 0$, est appelé « support de A ». Le plus souvent on utilise pour les fonctions d'appartenance les fonctions suivantes :

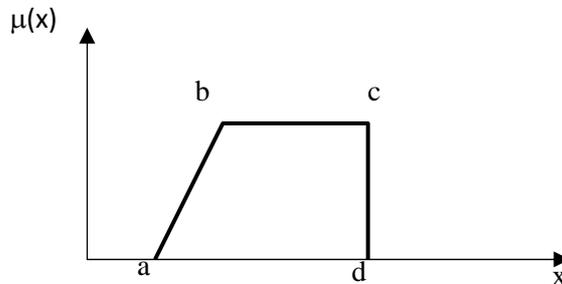
- Fonction triangulaire :



Elle est définie par trois paramètres $\{a, b, c\}$ qui déterminent les coordonnées des trois sommets :

$$\mu(x) = \text{Max} \left(\min \left(\frac{x-a}{b-a}, 1, \frac{c-x}{c-d} \right), 0 \right) \quad (\text{I.11})$$

➤ Fonction trapézoïdale :



Elle est définie par quatre paramètres $\{a, b, c, d\}$:

$$\mu(x) = \text{Max} \left(\min \left(\frac{x-a}{b-a}, 1, \frac{c-x}{c-d} \right), 0 \right) \quad (\text{I.12})$$

➤ Fonction gaussienne :

Elle est définie par deux paramètres $\{\sigma, m\}$:

$$\mu(x) = \exp\left(-\frac{(x-m)^2}{2(\sigma)^2}\right) \quad (\text{I.13})$$

➤ Fonction sinusoïdale :

Elle est définie par deux paramètres $\{a, c\}$:

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (\text{I.14})$$

I.5.5. Propositions et règle floue

Une proposition floue élémentaire est définie à partir d'une variable linguistique $(V; U; T_v)$ par la qualification « V est A », avec A appartient à T_v . Par exemple, « taille de Paul est MOYENNE » est une proposition élémentaire définie à partir de la variable linguistique :

(taille de Paul, {taille}, {..., MOYENNE,...}).

La valeur de vérité d'une proposition élémentaire « V est A » est égale à $\mu_A(v)$ où v correspond à la valeur numérique exacte de V.

Une proposition floue générale est définie à partir de propositions élémentaires et d'opérateurs logiques binaires (et, ou, implique) ou un aire (non). Il existe plusieurs méthodes pour calculer la valeur de vérité de telles propositions. Nous ne citons ici que les plus communément utilisées.

Conjonction : (V_1 est A_1) **et** (V_2 est A_2)

$$\min(\mu_{A_1}(v_1), (\mu_{A_2}(v_2)))$$

(Logique de Zadeh)

$$\max((\mu_{A_1}(v_1) + (\mu_{A_2}(v_2) - 1), 0))$$

(Logique de Lukasiewicz)

$$\mu_{A_1}(v_1) \times \mu_{A_2}(v_2)$$

Disjonction : V_1 est A_1) **ou** (V_2 est A_2)

$$\max(\mu_{A_1}(v_1), (\mu_{A_2}(v_2)))$$

(Logique de Zadeh)

$$\min((\mu_{A_1}(v_1) + (\mu_{A_2}(v_2), 1))$$

(Logique de Lukasiewicz)

$$(\mu_{A_1}(v_1) + (\mu_{A_2}(v_2) - \mu_{A_1}(v_1) \times \mu_{A_2}(v_2))$$

(Logique Probabiliste)

Implication : (V_1 est A_1) **implique** (V_2 est A_2)

$$\min(1 - \mu_{A_1}(v_1) + (\mu_{A_2}(v_2), 1))$$

(Lukasiewicz)

$$\min(\mu_{A_1}(v_1), (\mu_{A_2}(v_2)))$$

(Mamdani)

$$\mu_{A_1}(v_1) \times \mu_{A_2}(v_2)$$

(Larsen)

Complémentation : **non** (V est A)

$$1 - \mu_{A_1}(v)$$

Dans ce qui précède, v , v_1 et v_2 correspondent à des instanciations numériques réelles des variables V , V_1 et V_2 .

Une règle floue est une proposition floue générale utilisant une application entre deux propositions floues quelconques. Par exemple :

$$(V_1 \text{ est } A_1) \text{ et } (V_2 \text{ est } A_2) \text{ implique } (V_3 \text{ est } A_3)$$

Ou sous une forme plus linguistique :

$$\text{Si } (V_1 \text{ est } A_1) \text{ et } (V_2 \text{ est } A_2) \text{ alors } (V_3 \text{ est } A_3)$$

❖ Activation d'une règle floue

Une règle ne peut être activée (i.e. intervenir dans le processus d'inférence) que lorsque la valeur de vérité de la proposition floue constituant sa prémisse est nulle.

I.6. Commande par logique floue

Bien que la logique floue possède un champ d'application extrêmement vaste, nous nous intéressons ici qu'à son utilisation dans le cadre de la commande. En effet, cette méthode permet d'obtenir une loi de commande souvent très efficace sans devoir faire des études théoriques approfondies. L'algorithme de commande consiste en une collection de règles floues appelées base de règle. Dans cette section nous présentons toutes les bases générales de réglage et de commande par la logique floue.

Devant le problème délicat du contrôle et de la commande des systèmes complexes non linéaires et multi variables, les outils utilisés deviennent de plus en plus pointus. Un engouement certain s'est porté depuis quelques années autour de la logique floue, en parallèle des avancées de l'automatique classiques.

La motivation primaire de la logique floue est la possibilité d'exploiter la tolérance de l'imprécision et de l'exactitude [Jamshidi, 93].

L'idée d'appliquer les techniques floues au domaine de contrôle des processus a été proposée la première fois par Zadeh et Change en 1972. Leur démarche reposait sur une approche à la base de modèle en utilisant le concept de fonction flou. Les principes de la logique floue ont été appliqués en 1974 par E.M Mamadani à la construction d'un premier contrôleur flou pour une commande de vitesse d'un moteur [Mamdani, 74]

Mais ce n'est que depuis quelques années que la commande floue à connu, essentiellement au Japon, un essor remarquable. Elle a été appliquée à des problèmes aussi divers que la purification des eaux, la fabrication du ciment, la marche automatisée d'une rame de métro, etc. [Barrat, 93], [Boujahia, 00].

L'attitude des automaticiens à l'égard de la commande floue a d'abord été réservée : après avoir, pendant des décennies, affirmé la nécessité d'identifier le mieux possible un système pour pouvoir construire une commande ayant des performances satisfaisantes, fallait-il faire confiance à cette méthode, qui prétendait remplacer les commandes plus au moins sophistiquées, basées sur l'identification, par des techniques s'appuyant sur le savoir-faire humain plutôt que sur des équations ?

Actuellement, cette attitude a évoluée. On peut penser que la commande floue va peu à peu prendre place dans la panoplie de l'ingénieur contemporain, sans supplanter les méthodes traditionnelles, et quelles constituera un complément précieux dans le cas des systèmes difficilement identifiables ou dont les paramètres subissent des variations brutales.

Il existe beaucoup d'efforts de recherches réussies pour les applications de la théorie de la logique floue aux systèmes à commander. Ces travaux montrent une performance remarquable de ce type de régulateur en comparant avec le régulateur PID conventionnel [Barrat, 93], [Lauchlan, 97].

I.6.1. Principe et réalisation d'un contrôleur flou

I.6.1.1. Procédé général lors de la conception d'une commande par la logique floue

Le procédé à suivre lors de la conception d'une commande par la logique floue est assez différent de celui d'une commande conventionnelle. La figure I.5 en montre les étapes principales. Généralement, lors de la conception d'un réglage ou d'une commande par logique floue nous commençons par une analyse approfondie sur le système à commander, ici il ne s'agit pas d'une recherche d'un modèle mathématique, mais il s'agit d'une exploration complète du comportement du système vis-à-vis de la variation de la commande, cette exploration peut apparaître comme une description du comportement dynamique du système, qui se fait par des variables linguistiques qui peuvent être accompagnées d'une définition par fonction d'appartenance.

L'étape suivante consiste à une recherche d'une stratégie de commande, elle comprend la phase de fuzzification, les inférences et la défuzzification. En suite l'implémentation qui se fait généralement sur un PC, elle est suivie par une opération de test sur une installation réelle qui comporte bien sur le système à commander.

Finalement afin de trouver un comportement convenable, le changement de la stratégie de commande en plusieurs passages est très indispensable.

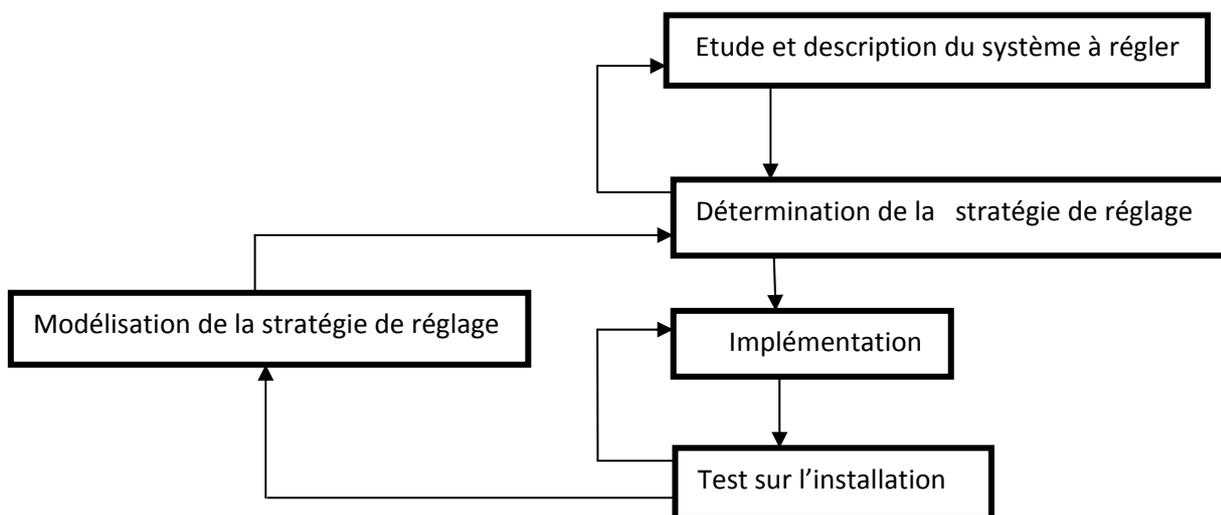


Figure I-5 : Procédé général lors de la conception d'une commande par logique floue

I.6.1.2. Propriétés d'une commande par logique floue

Les propriétés essentielles d'une commande par logique floue peuvent être résumées de la manière suivante :

- Mesure de plusieurs grandeurs : la grandeur à régler et d'autres grandeurs caractérisant le comportement dynamique du système à régler,
- Comportement non linéaire du système,
- Aptitude à commander convenablement surtout des systèmes avec un comportement dynamique compliqué, dont la modélisation est difficile,
- Connaissance approximative des paramètres du système à commander.

La détermination de la stratégie de commande représente l'étape la plus importante, lors d'une conception d'une commande par logique floue. En effet, cette dernière contient la détermination de plusieurs paramètres. L'ensemble de ces derniers est très important, nombre et forme des variables linguistiques des entrées, et mêmes des sorties, choix des différentes logiques, choix de la méthode de défuzzification...

L'analyse de l'influence de ces paramètres n'est évidemment pas une chose facile. Néanmoins, plusieurs études récentes ont tenté en partie de faire varier certains de ces paramètres pour dégager quelques règles. Des études concernant les fonctions d'appartenances ont été menées [Misumoto, 88], [Driankov, 93], [Buhler, 94], [Tso et fung, 97], [Li, 99], [Zhou, 00], [Botzheim, 04], néanmoins la forme et le nombre de ces dernières sont très dépendant de l'application et ne peuvent se dissocier de la partie règles floues, une très grande quantité de travaux ont été réalisés dont le but est d'améliorer et innover cette partie très importante dans un régulateur flou. Parmi ces travaux on peut citer les plus récents tels que, [Wong, 99], [Zapata, 99], [Yeh, 00], [Zhou, 00], [Shiqian Wu, 01], [Chen, 02], [Tao, 02], [Pomares, 04]. Des études ont été également menées concernant les gains de normalisation [Ouatara, 96], [Li, 99], [Yeh, 99].

I.6.2. Les différentes étapes de la commande floue

Un contrôleur flou est un système à base de connaissances particulier, utilisant un raisonnement en profondeur limité, dans une procédure de chaînage ayant des règles (activation des règles par les prémisses). Toutes les règles activables (prémisses caractérisée par un degré d'appartenance non nul) sont activées. On considère en suite une 'moyenne' sur le résultat de ces règles pour engendrer une décision finale. On peut distinguer plusieurs étapes dans le traitement des règles. Un schéma représentatif peut être le suivant :(figure I-16)

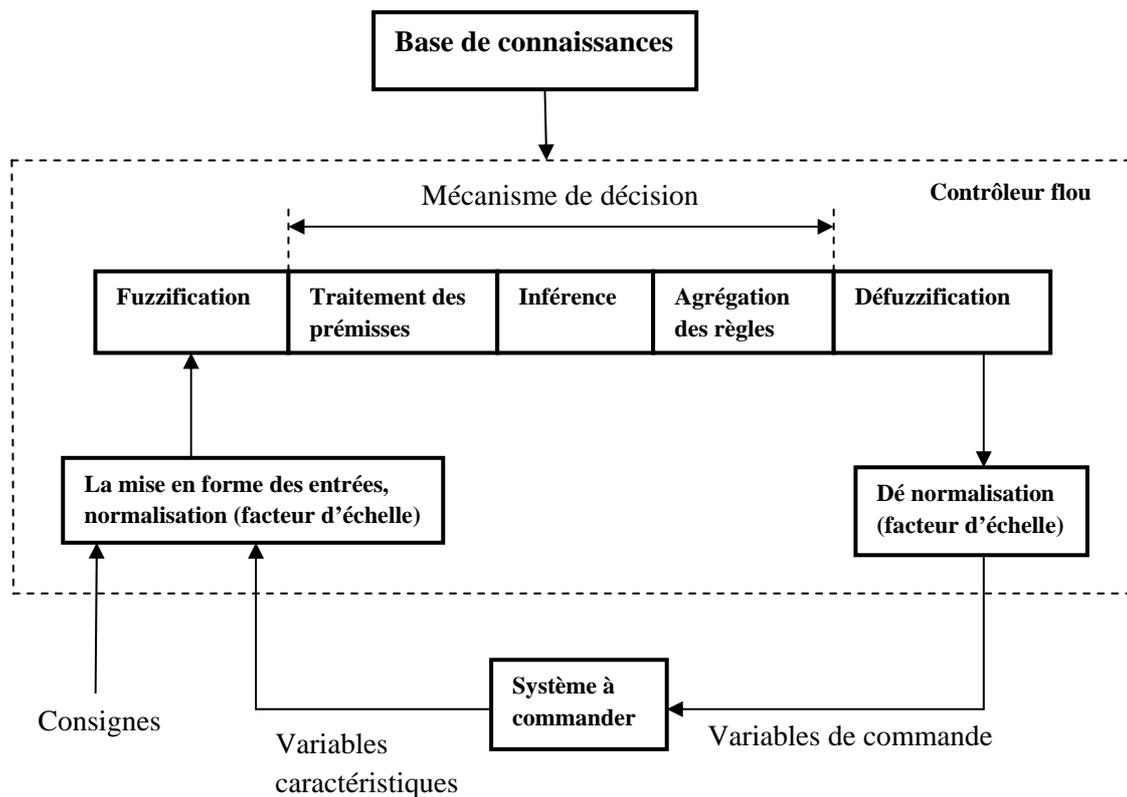


Figure I-6 : Structure de base d'un régulateur flou

Les variables caractéristiques du système à commander et les consignes définissent les variables d'entrées du contrôleur flou. Les variables caractéristiques sont, en général, les grandeurs de sortie de processus et, le cas échéant, d'autres mesures déterminantes pour saisir l'évolution dynamique de processus. Des variables de sortie du contrôleur flou sont les commandes à appliquer au processus.

I.6.2.1. Base de connaissances

La conception d'une base de connaissances représente la phase la plus importante dans la conception des systèmes experts. Elle comprend la base de données et la base des règles floues.

- **La base de données**

Contient la définition des ensembles flous, les facteurs d'échelles pour la normalisation des ensembles de référence et la partition de l'espace flou d'entrée et de sortie.

- **La base de règles floues**

Elle rassemble l'ensemble des règles floues de type «Si-Alors» décrivant en termes linguistiques basés sur la connaissance d'un expert le comportement dynamique du système.

R_1 : Si x_1 est A^1_1 et.....et x_n est A^1_n

Alors u_1 est B^1

Avec : $[x_1 \text{ K } x_n]$: les entrées du régulateur

u_1 : la sortie du régulateur

Chaque régulateur activé donne un sous-ensemble flou de sortie.

I.6.2.2.Mise en forme des entrées, normalisation

Cette première étape permet le traitement des variables d'entrées du contrôleur flou, par exemple, calcul d'erreurs (différences entre grandeurs mesurées et consignes) et de variation d'erreurs. L'utilisation de domaines normalisés (univers de discours compris entre -1et1) nécessite une transformation d'échelle. Celle-ci est réalisée par l'intermédiaire de facteurs d'échelle transformant les grandeurs physiques des entrées en des valeurs normalisées appartenant à l'intervalle [-1, 1].

I.6.2.3.Interface de fuzzification

Transforme les entrées numériques (vecteurs précises) en des ensembles flous. Le choix de l'opérateur de fuzzification dépend de la confiance que l'on accorde aux mesures effectuées.

Ainsi si la mesure x_0 est exacte, le sous-ensemble flou A_x doit être représenté par un fait précis. Par conséquent, on utilise comme opérateur de fuzzification la transformation dite de singleton. La fonction d'appartenance du sous-ensemble flou A_x est alors définie par :

$$\mu_{A_x}(x) = \begin{cases} 1 & \text{si } x = x_0 \\ 0 & \text{si } x \neq x_0 \end{cases} \quad (\text{I.15})$$

La figure I-7 montre l'aspect de cette fonction d'appartenance.

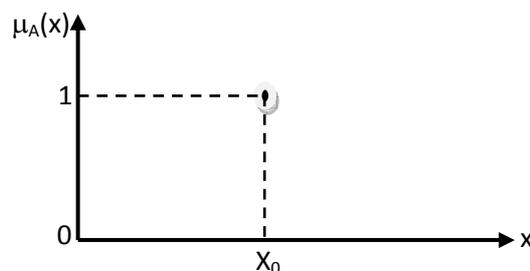


Figure I-7 : Méthode de fuzzification pour une mesure exacte

Par contre, si la mesure de la variable est incertaine, par exemple à cause de bruit, le sous-ensemble flou A_x doit être représenté par un fait imprécis. On utilise alors la méthode de fuzzification qui associe à la variable de la mesure x_0 une fonction d'appartenance telle que, par exemple :

$$\mu_{A_x}(x) = \max \left\{ 0 ; 1 - \frac{|x-x_0|}{\varepsilon} \right\} \quad (\text{I.16})$$

La représentation graphique de cette fonction est représentée sur la figure I.8. Ce sous-ensemble flou comprend donc la mesure x_0 avec une appartenance unité et les valeurs voisines de x_0 avec une appartenance inversement proportionnelle à l'écart avec x_0 .

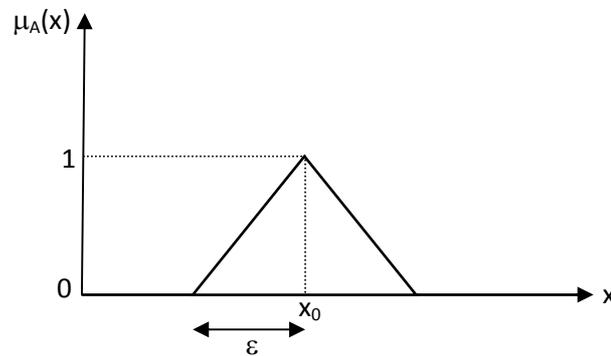


Figure I-8 : Méthode de fuzzification pour une mesure incertaine

La base de triangle (ε) est fonction de l'importance relative des erreurs des mesures. En effet, plus elles sont importantes, plus la mesure de la variable x_0 devient imprécise, et donc, plus le triangle droit s'élargit.

Le choix de la forme des fonctions d'appartenance (triangulaires, trapézoïdales, exponentielles, gaussiennes,...) est arbitraire. Des études comparatives ont montrés, selon les différentes formes de fonctions d'appartenance, des résultats pratiquement similaires en boucle fermée mais les formes triangulaires facilitent la programmation ce qui explique qu'elles soient le plus fréquemment utilisées. Quand au nombre de fonctions d'appartenance, est généralement impair car elles se répartissent autour de zéro (3,5 et 7 sont des valeurs courantes). Un exemple de fonctions d'appartenance triangulaires est donné dans la figure I-19.

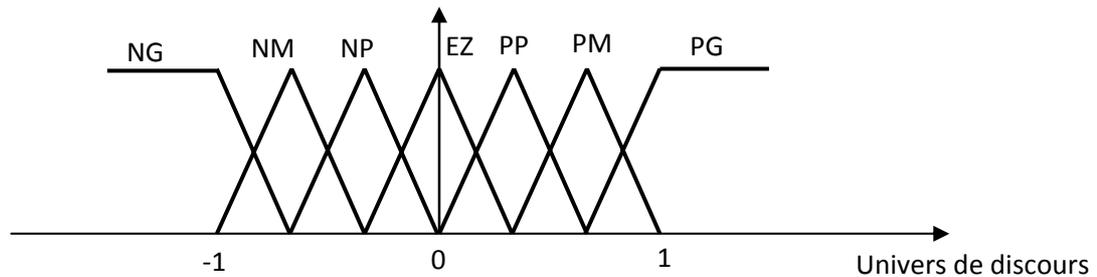


Figure I-9 : Exemple de fonctions d'appartenance triangulaire et trapézoïdale

NG, NM, ..., PG sont des valeurs linguistiques, avec :

- NG : Négatif Grand
- NM : Négatif Moyen
- NP : Négatif Petit
- EZ : Zero
- PP : Petit Positif
- PM : Positif Moyen
- PG : Positif Grand

I.6.2.4. Traitement des prémisses composées

En général, les prémisses des règles vont comporter plusieurs clauses liées par des connecteurs « ET », « OU » et « NON ». Dans la pratique, pour les opérations de conjonction et disjonction, on a souvent recours, parmi les normes et conformes triangulaires, aux opérateurs min et max. Quant à la négation A^c d'un ensemble flou A, elle est caractérisée par $\mu_{A^c}(x) = 1 - \mu_A(x)$.

Des modificateurs linguistiques peuvent aussi être utilisés dans l'écriture de ces règles, par exemple :

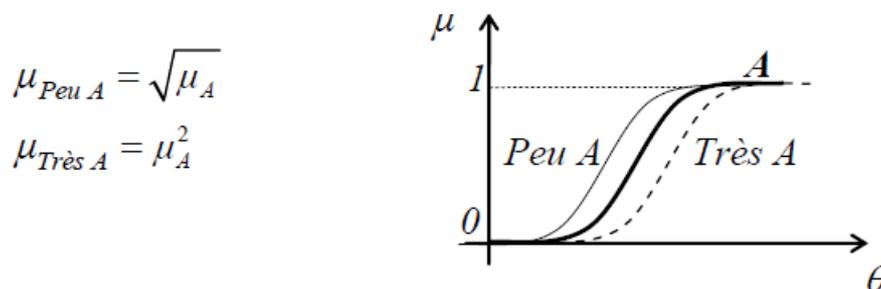


Figure I-10. Exemple de modificateurs linguistiques

I.6.2.5. Moteur d'inférence floue

C'est un mécanisme de décision. Il permet à partir d'un fait observé de la base des règles floues une décision en exploitant le raisonnement approximatif. Dans les inférences de régulateur par

logique floue interviennent les opérateurs ET et OU. L'opérateur ET s'applique aux variables à l'intérieur d'une règle tandis que l'opérateur OU lie les différentes règles.

Il existe plusieurs méthodes pour réaliser ces opérateurs dans une inférence et qui s'appliquent aux fonctions d'appartenance à savoir :

- Méthode d'inférence max-min (Mamdani)
- Méthode d'inférence max-prod (Larsen)
- Méthode d'inférence somme-prod (Sugeno)

❖ Méthode d'inférence max-min

La méthode d'inférence max-min, réalise, au niveau de la condition, l'opérateur OU par la formation de maximum et l'opérateur ET par la formation de minimum.

La figure I-11. Représente graphiquement le principe de la méthode d'inférence max-min.

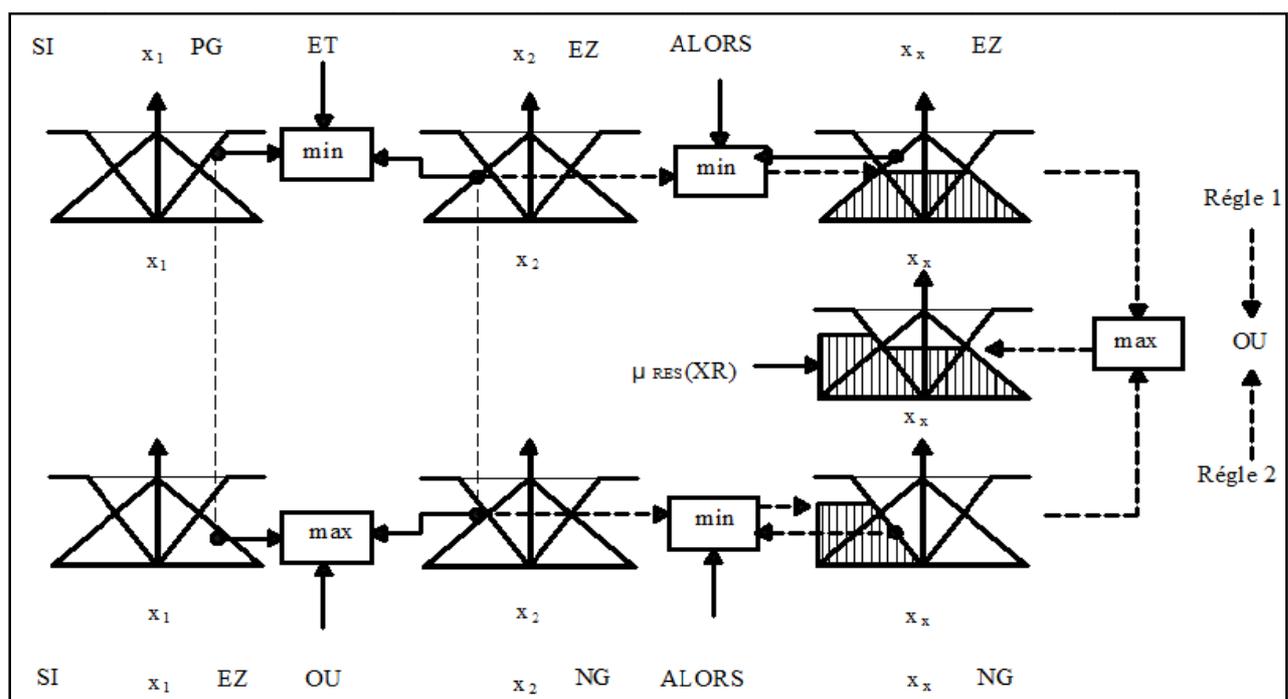


Figure I-11 : Méthode d'inférence max-min pour deux variables d'entrée et deux règles

❖ Méthode d'inférence max-prod

La méthode d'inférence max-prod réalise en général, au niveau de la condition, l'opérateur OU par la formation du maximum et l'opérateur ET par la formation du minimum, par contre, la conclusion dans chaque règle, introduite par ALORS, qui lie la facteur d'appartenance de la

condition avec la fonction d'appartenance de la variable de sortie x_R par l'opérateur ET, est réalisée cette fois ci par la formation du produit (minimum pour la méthode d'inférence max-min).

La représentation graphique du principe de la méthode d'inférence max-prod se trouve sur la figure I-12.

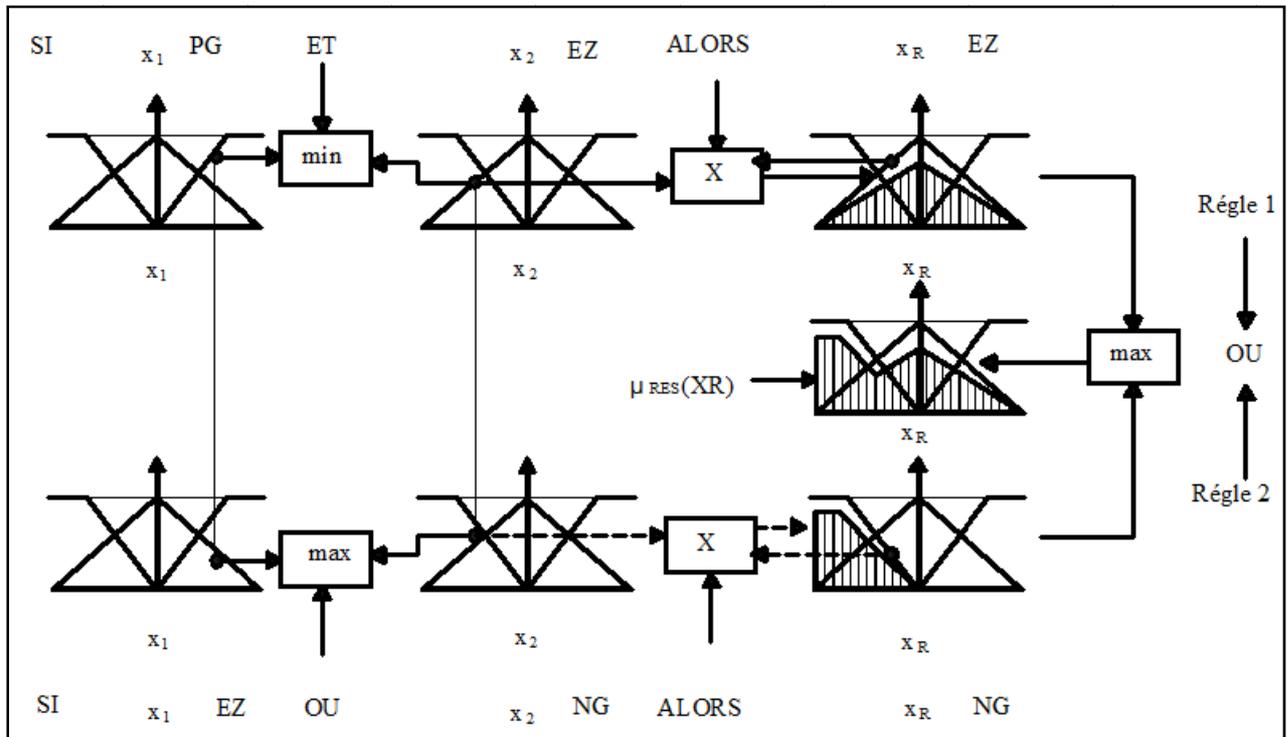


Figure I-12 : Méthode d'inférence max-prod pour deux variables d'entrée et deux règles

❖ **Méthode d'inférence somme-prod**

Par opposition aux méthodes d'inférence précédentes, la méthode d'inférence somme-prod réalise, au niveau de la condition, l'opérateur OU par la formation de somme, plus précisément par la valeur moyenne, tandis que l'opérateur ET est réalisée par la formation du produit. La méthode d'inférence somme-prod est représentée graphiquement à la figure I-13.

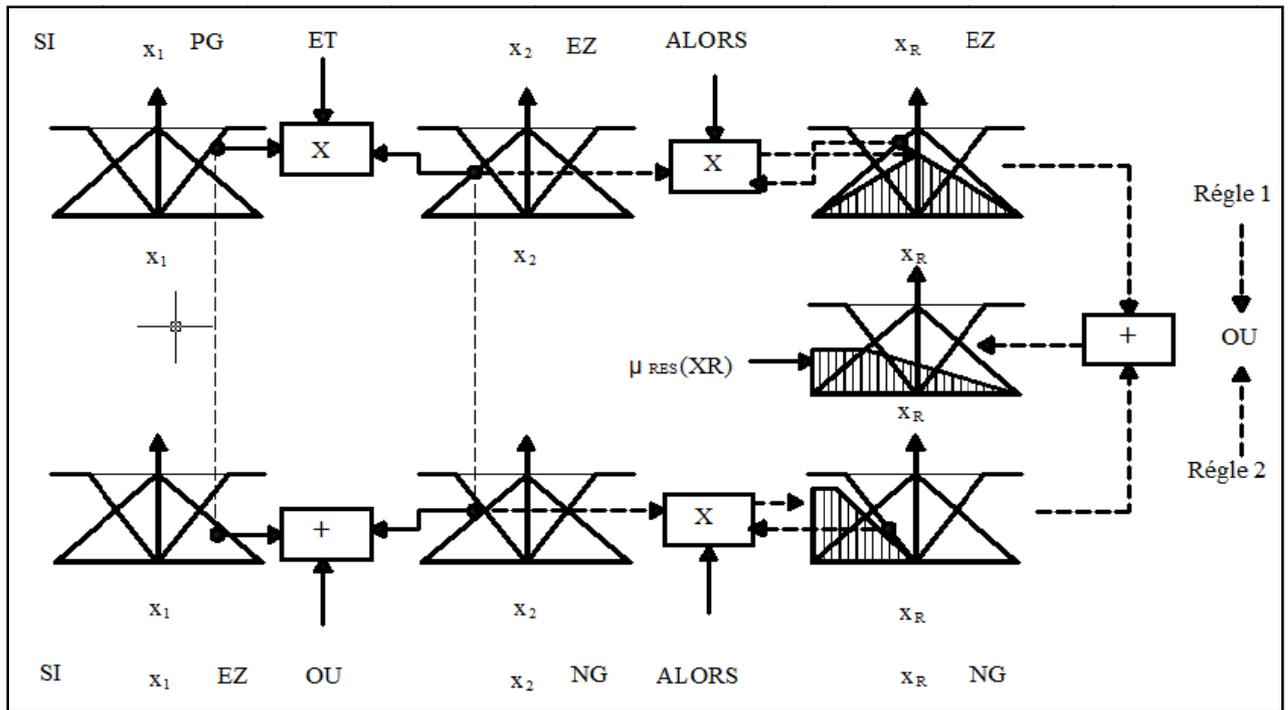


Figure I-13 : Méthode d’inférence somme-prod pour deux variables d’entrée et deux règles

I.6.2.6. Interface de défuzzification

La stratégie de la commande utilisée est décrite par un ensemble de déclaration linguistique ou règles. Puisque le processus exige à ses entrées des valeurs non floues, l’ensemble flou de sortie du régulateur doit être défuzzifié. La défuzzification est l’opération qui consiste à extraire d’un ensemble flou une valeur numérique considérée comme valeur significative. Dans la commande en temps réel, un critère de choix de la commande de défuzzification est la simplicité de calcul.

Récemment [Jean J et al., 00], ils ont proposé une nouvelle méthode de défuzzification permet d’améliorer les inconvénients des méthodes classiques. Même, [Broekhoven et al., 04], ils ont suggéré trois nouvelles méthodes de défuzzification. En ce qui concerne les méthodes classiques, on peut citer les plus utilisées. La figure I.24, illustre ces dernières graphiquement :

- **Méthode du centre de gravité (DEF1)**

C’est la méthode la plus utilisée dans les contrôleurs flous. Elle donne comme valeur le centre de gravité de l’ensemble flou. La sortie défuzzifiée Z est définie par :

$$Z = \frac{\sum_{j=1}^n \mu_Z(z_j)z_j}{\sum_{j=1}^n \mu_Z(z_j)} \tag{I.17}$$

- **Méthode du plus grand maximum (DEF2)**

Cette méthode donne comme résultats la plus grande valeur dont la grandeur a le plus grand degré d'appartenance.

- **Méthode du plus petit maximum (DEF3)**

Elle donne comme résultat la plus petite valeur dont la grandeur a le plus grand degré d'appartenance.

- **Méthode de la moyenne des maxima (DEF4)**

Elle donne comme valeur la moyenne des valeurs dont le degré d'appartenance est maximal.

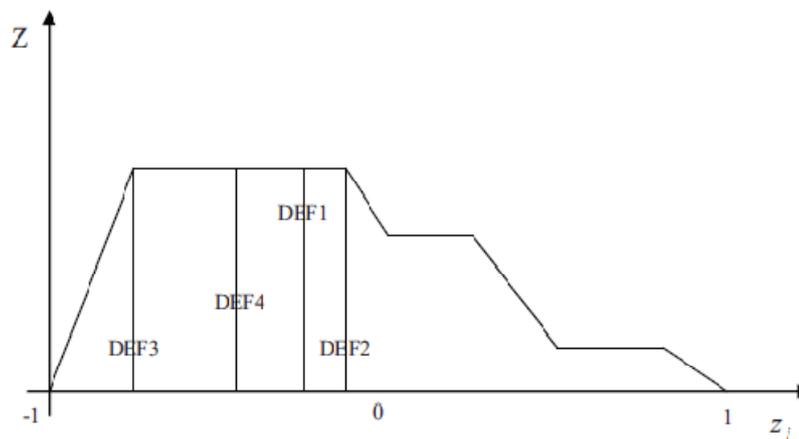


Figure I.14. Différentes méthodes de défuzzification

I.6.2.7. Mise en forme des sorties, dénormalisation

Cette dernière étape transforme les valeurs normalisées des variables de commande en des valeurs appartenant à leur domaine physique respectif.

I.6.3. Structure de base d'un contrôleur flou : analogie structurelle avec les P.I.D

Pour des problèmes de régulation monovariante simples, les entrées du contrôleur flou sont généralement l'erreur e (différence entre la consigne et sortie du processus) et la variation de l'erreur Δe (traduction de la dynamique du système).

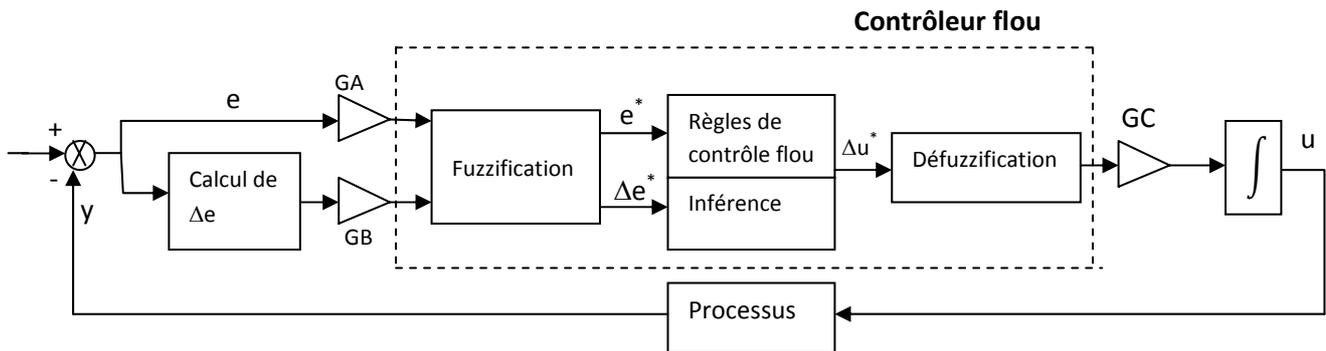


Figure. I-15 : Structure de base d'un contrôleur flou

Ga, GB, GC sont des gains ou facteurs d'échelle.

Selon que la sortie du contrôleur flou concerne la commande ou sa variation. On obtiendra des équivalents structuraux non linéaires des contrôleurs classiques P.D, PI, ou même PID, en changeant le nombre d'entrées des contrôleurs.

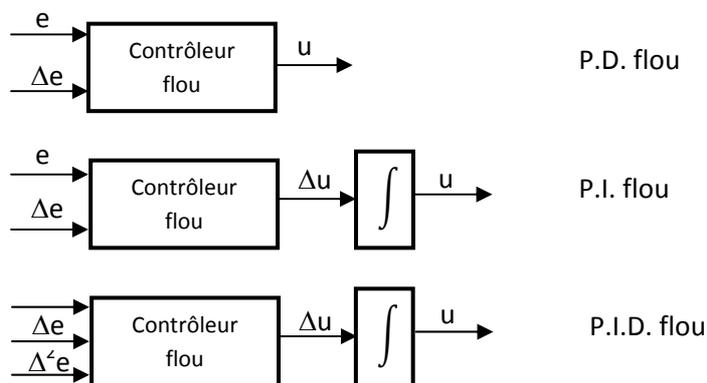


Figure I-6 : Equivalents flous des contrôleurs usuels

I.6.4. Réglage, stabilité et robustesse d'un contrôleur flou

a/ Réglage

Un contrôleur flou possède qui peu de nombreux paramètres de réglages, ce qui peut, priori, effrayer ses utilisateurs potentiels. En effet, contrairement aux contrôleurs classiques (notamment de type PID : 3 constantes à régler), le contrôleur flou possède un nombre bien plus conséquent de paramètres, et offre, par la même, d'avantage de degrés de liberté.

On peut citer parmi les choix et les réglages à faire :

- L'expression des règles,
- La définition des variables et des valeurs linguistiques (sous-ensembles flous pour chaque variable), avec leur fonction d'appartenance associés,
- La méthode d'implication,
- La méthode d'inférence,
- La méthode de défuzzification,
- Les facteurs d'échelle (gains) sur les entrées et les sorties du contrôleur,

Cependant, on constate une certaine insensibilité du résultat au choix des méthodes d'implication, d'inférence et de défuzzification. Pour les autres paramètres, un réglage séquentiel est possible :

- Facteur d'échelle (effet global sur le comportement du système),
- Fonction d'appartenance,
- Règles (effet local sur le comportement du système),

Le réglage par essais des nombreux paramètres étant assez long et fastidieux, diverses techniques d'auto-réglage, d'optimisation et d'apprentissage ont été développées ces dernières années. On peut citer, à titre d'exemple, les techniques de la programmation mathématique, les réseaux de neurones, les algorithmes génétiques.

b/ Stabilité

L'analyse de la stabilité est a priori difficile dans une approche de type "système à base de connaissance" dont l'objectif est d'éviter l'utilisation de méthodes mathématiques du procédé. Une modélisation floue du système bouclé est alors nécessaire pour l'analyse de la stabilité. Toutefois, si une caractérisation mathématique du processus à commander existe, le contrôleur flou étant, en général, un contrôleur non linéaire, certains résultats de la théorie des systèmes non linéaires peuvent être utilisés. On distingue alors deux grandes approches :

- La méthode directe de Lyapunov : cette méthode permet d'affirmer la stabilité asymptotique d'un état d'équilibre s'il existe une fonction de Lyapunov $V(x, t)$ définie positive telle que sa dérivée $\frac{dV(x,t)}{dt}$ soit définie négative.

•L'approche entrée-sortie : cette approche permet d'affirmer la stabilité d'un système si la sortie de celui-ci, en réponse à une entrée bornée, est elle-même bornée.

Nous choisissons ici de distinguer les différentes méthodes d'analyse de la stabilité selon que l'on dispose du modèle mathématique du processus ou pas. La liste des méthodes ici citées est non exhaustive et le lecteur peut se référer à [BO94, MT91] pour en avoir un aperçu plus complet.

c/ Robustesse

La robustesse, bien que maintes fois constatée, ne possède aucune démonstration rigoureuse. Son explication est à chercher dans l'espace non linéaire de cette commande qui peut apparaître comme un contrôleur P.I.D. à gains variables ou pré-programmés, selon les points de fonctionnement. Toutefois, si le modèle du système en boucle fermée est un modèle "flou dynamique" de Sugino, on peut se reporter aux travaux d'automatique classique.

I.6.5. Différents types de régulateurs flous

I.6.5.1. Régulateur de type Mamdani

Dans la plupart des applications rapportées dans la littérature, un contrôleur de ce type est conçu pour réguler, asservir une variable de sortie d'un procédé, soit uniquement à partir de l'erreur e (consigne moins la mesure), soit à partir de l'erreur et sa variation Δe [Foulloy et al., 03]. En 1974, E.H Mamdani avait présenté, pour la première fois, la technique de réglage par la logique floue. Celle-ci consiste à déterminer un ensemble de règles qui maîtrise le comportement dynamique du système à commander. L'obtention de ces règles est facile auprès des experts qui connaissent bien le système. Il avait utilisé des règles à prémisses et conclusions symboliques. L'inférence max-min, et la défuzzification par centre de gravité.

Après, Macvicar et Whelan ont observé que les bases de règles proposées par King et Mamdani étaient incomplètes. Ils ont fait analyse détaillée sur ces bases de règles et proposé une matrice de règles (voir le tableau I.1) qui possède deux entrées, et l'erreur et sa variation, en se basant sur les deux principes suivants [Teng et al., 94], [Boujahia, 00] :

- Si la sortie est égale à la valeur désirée et la variation de l'erreur est nulle, la commande sera maintenue.
- Si la sortie diverge de la valeur désirée, l'action sera dépendante du signe de la valeur de l'erreur et de sa variation.

Si les conditions telles que l'erreur peut être corrigée par elle-même, alors la commande sera maintenue. Dans le cas contraire, la commande sera chargée pour avoir des résultats satisfaisants.

$e \backslash \Delta e$	NB	NS	ZE	PS	PB
NB	NB	NB	NB	NS	ZE
NS	NB	NB	NS	ZE	PS
ZE	ZE	NS	ZE	PS	PB
PS	NS	ZE	PS	PB	PB
PB	ZE	PS	PB	PB	PB

Tableau I.1 : Matrice de Moccicar-Whelan

Avec :

e : l'écart entre la sortie du système et la consigne

Δe : Variation de l'erreur

NB : Négatif grand « Negative Big »

NS : Négatif petit « Negative Small »

ZE : Zéro

PS : Positif petit « Positive Small »

PB : Positif grand « Positive Big »

I.6.5.2. Régulateur de type Sugeno

Dans les régulateurs de ce type, les conclusions des règles ne sont pas symboliques (i.e représentées par des sous-ensembles flous) mais une fonction (numériques) des entrées :

$b^i = f(x_1, K, x_n)$, les prémisses étant symboliques.

Où $f(\cdot)$ est généralement fonction polynomiale.

La sortie du régulateur est donnée par :

$$y = \frac{\sum_{i=1}^n b^i * \alpha_i(x)}{\sum_{j=1}^n \alpha_j} \quad (\text{I.18})$$

Où les α_i sont les valeurs de vérité de chaque règle pour $i = 1 \dots n$.

Notons que la sortie donnée par le régulateur est la variation du signal de commande.

I.6.6. Avantages et inconvénients de la commande par la logique floue

Certainement, la commande par la logique floue comporte un certain nombre d'avantages et d'inconvénients.

Les avantages essentiels sont :

- La non-nécessité d'une modélisation du processus à commander, ainsi que d'une analyse mathématique profonde.
- La possibilité de bénéficier et d'implémenter des connaissances et des expertises humaines sur le système à commander.
- La maîtrise du système à commander avec un comportement complexe (fortement non-linéaire et difficile à modéliser), aussi, cette approche offre la possibilité d'utiliser des processeurs spécialisés (dit processeurs flous), afin d'augmenter la précision et la vitesse de calcul.

Par contre, les inconvénients sont :

- Le manque de méthodes systématiques précises pour la conception et la synthèse d'une commande par la logique floue (par exemple dans le choix des grandeurs à mesurer, dans la détermination des facteurs de normalisation, dans le choix de la stratégie, soit de fuzzification, les inférences et la defuzzification,...)
- L'impossibilité de la démonstration de la stabilité du circuit de commande en toute généralité
- La cohérence des inférences non garantie a priori (apparition de règles d'inférence contradictoire possible)
- La possibilité d'apparition de cycles limites à cause de fonctionnement non-linéaire

I.7. Conclusion

Dans ce chapitre, nous avons rappelé les éléments de base relatifs à la théorie de la logique floue. En effet, des définitions de base sur les variables linguistiques, les sous-ensembles flous, ainsi que les opérateurs flous ont été donnés. L'utilisation de la logique floue dans la commande a été également présentée en décrivant les différents éléments constitutifs d'un régulateur flou. A travers cette étude, nous constatons que deux types de régulateurs flous, celui de Mamdani et celui de Sugeno, sont les plus utilisés. Il est à noter que dans le premier les variables d'entrées et de sorties sont numériques. Cette dernière caractéristique lui donne l'avantage d'être plus adapté lors de l'analyse mathématique du système, en particulier l'analyse de la stabilité.

II.1. Introduction

Pour commander un robot ou pour simuler son comportement on doit disposer de modèle. Plusieurs niveaux de modélisation sont possibles selon les objectifs, les contraintes de la tâche et les performances recherchées.

Les modèles mathématiques nécessaires sont :

- Les modèles géométriques direct et inverse qui expriment la situation de l'organe terminal en fonction des variables articulaires et inversement.
- Les modèles cinématiques direct et inverse qui expriment les vitesses de l'organe terminal en fonction des variables articulaires et inversement.
- Les modèles dynamiques définissant les équations du mouvement du robot qui permettent d'établir les relations entre les couples ou forces exercées par les actionneurs et les positions, vitesses et accélérations des articulations.

II.2. Définitions générales

La robotique est une science pluridisciplinaire qui comprend la mécanique, l'automatique, l'électrotechnique, le traitement de signal, l'informatique, communication.....

Un robot se compose de :

a) Le mécanisme : structure plus au moins proche de celle de bras humain, on dit aussi manipulateur quand il ne s'agit pas d'un robot mobile. Sa motorisation est réalisée par des actionneurs électriques, pneumatiques ou hydrauliques qui transmettent leur mouvement aux articulateurs par des systèmes appropriés.

b) La perception : permet de gérer les relations entre le robot et son environnement. Les organes de perceptions sont des capteurs dits « proprioceptifs » lorsqu'ils mesurent l'état interne du robot (position et vitesses des articulations) ou « extéroceptifs » lorsqu'ils recueillent des informations sur l'environnement (détection de présence, mesure de distance, vision artificielle).

c) La commande : qui synthétise les consignes de l'asservissement pilotant les actionneurs. A partir de la fonction de perception et des ordres de l'utilisateur, elle permet d'engendrer les actions du robot.

d) Interface homme-machine : à travers laquelle l'utilisateur programme les tâches que le robot doit exécuter.

e) **Le poste de travail et les dispositifs périodique** : qui constituent l'environnement dans le quel évolue le robot.

II.2.1. Difficultés

- Complexité de la cinématique
- Le nombre de degré de liberté
- Type d'articulation (prismatique ou rotoïde)
- Type de chaîne (ouverte simple, arborescente ou fermée)

Pour obtenir un bon modèle il faut :

1/ Mettre en oeuvre des procédures efficaces d'identification et de leurs paramètres constitutifs.

2/ Pour qu'une commande puisse être effectivement implantée sur un contrôleur de robot, les modèles doivent être calculés en ligne et donc le nombre d'opération à effectuer doit être minimum

II.2.2. Architecture des robots

Un robot comporte 2 parties essentielles :

Le porteur : structure mécanique articulée constituée des 3 premiers degrés de liberté à partir du bâti. Si P un point de l'extrémité et R_0 un repère lié au bâti, le rôle du porteur est de fixer la position de P dans R_0 .

Les liaisons utilisées sont des liaisons pivots notées R ou prismatiques notées P.

Le poignet : il est destiné à l'orientation de la pince ou de l'outil porté par le robot.

La façon dont les liaisons motorisée sont réparties du bâti au poignet définit trois grandes classes d'architecture :

- Architecture série (ou chaîne cinématique ouverte)
- Architecture parallèle (ou chaîne cinématique multi boucle)
- Architecture mixte (série-parallèle ou parallèle série)

II.3. Modélisation géométrique

II.3.1. Description de la géométrie des robots à structure ouverte simple

Ce paragraphe présente la méthodologie à suivre pour décrire les robots à structure ouverte simple.

Une structure ouverte simple est composée de $n+1$ corps notés C_0, \dots, C_n et de n articulations, le corps C_0 désigne la base du robot et le corps C_n le corps qui porte l'organe terminal et l'articulation j connecte le corps C_j au corps C_{j-1} (figure II-2)

La méthode de description est fondée sur les règles et conventions suivantes :

- les corps sont supposés parfaitement rigides. Ils sont connectés par des articulations considérées comme idéales (pas de jeu mécanique, pas d'élasticité), soit rotoides, soit prismatiques ;
- le repère R_j soit lié au corps C_j ;
- la variable de l'articulation j est notée q_j

Le repère R_j , fixé au corps C_j , est défini de sorte que :

- L'axe Z_j est porté par l'axe de l'articulation j ;
- L'axe X_j est porté par la perpendiculaire commune aux axes Z_j et Z_{j+1} . Si les axes Z_j et Z_{j+1} sont parallèles ou colinéaires, le choix de X_j n'est pas unique : des considérations de symétrie ou de simplicité permettent alors le choix rationnel.

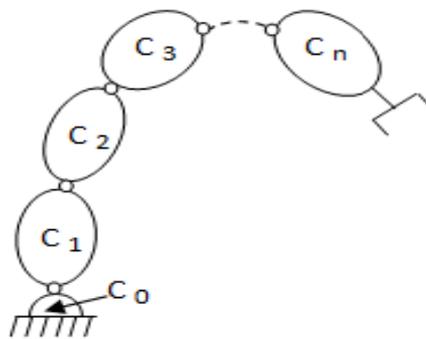


Figure II-1 : Robot à structure ouverte simple

II.3.1.1. Les paramètres de Denavit-hartenberg

Le passage de repère R_{j-1} au repère R_j s'exprime en fonction de quatre paramètres géométriques suivants (figure.II.2) :

- α_j : angle entre les axes Z_{j-1} et Z_j correspondant à une rotation autour de X_{j-1} ;
- d_j : distance entre Z_{j-1} et Z_j le long de X_{j-1} ;
- θ_j : angle entre les axes X_{j-1} et X_j correspondant à une rotation autour de Z_j ;

- r_j distance entre X_{j-1} et X_j le long de Z_j .

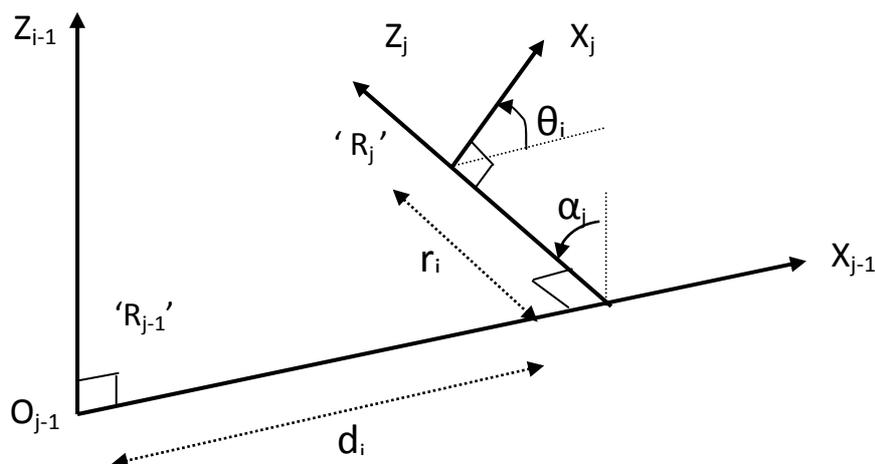


Figure II-2: Paramètres géométriques dans le cas d'une structure ouverte simple

La variable articulaire q_j associée à la $j^{\text{ième}}$ articulation est soit θ_j , soit r_j , selon que cette articulation est de type rotoïde ou prismatique, ce qui se traduit par la relation :

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \quad [\text{II.1}]$$

Avec :

- $\sigma_j = 0$ si l'articulation j est rotoïde ;
- $\sigma_j = 1$ si l'articulation j est prismatique ;
- $\bar{\sigma}_j = 1 - \sigma_j$

Par analogie, on définit le paramètre \bar{q}_j qui sera utilisé par la suite, tel que :

$$\bar{q}_j = \sigma_j \theta_j + \bar{\sigma}_j r_j \quad [\text{II.2}]$$

La matrice de transformation définissant le repère R_j dans le repère R_{j-1} est donné par la relation [II.3].

$${}^{j-1}T_j = \text{Rot}(x, \alpha_j) \text{Trans}(x, d_j) \text{Rot}(z, \theta_j) \text{Trans}(z, r_j)$$

$${}^{j-1}T_j = \begin{bmatrix} C\theta_j & -S\theta_j & 0 & d_j \\ C\alpha_j S\theta_j & C\alpha_j C\theta_j & S\alpha_j & -r_j S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j & r_j C\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [\text{II.3}]$$

On remarque que la matrice de rotation (3×3), ${}^{j-1}A_j$ peut être obtenue par :

$${}^{j-1}A_j = \text{rot}(x, \alpha_j) \text{rot}(z, \theta_j)$$

La matrice de transformation définissant R_{j-1} dans R_j est donnée par :

$${}^jT_{j-1} = \text{Trans}(z, -r_j) \text{Rot}(z, -\theta_j) \text{Trans}(x, d_j) \text{Rot}(x, -\alpha_j)$$

Tel que :

$${}^jT_{j-1} = \begin{bmatrix} & & -d_j C\theta_j \\ & {}^{j-1}A_j^T & d_j S\theta_j \\ 0 & 0 & -r_j \\ 0 & 0 & 1 \end{bmatrix} \quad [\text{II.4}]$$

Remarque :

- pour la définition du repère de référence R_0 , le choix le plus simple consiste à prendre R_0 confondu avec le repère R_1 quand $q_1=0$, ce qui signifie que z_0 est confondu avec z_1 et $O_0 = O_1$ lorsque l'articulation 1 est rotoïde, et z_0 est confondu avec z_1 , et x_0 est parallèle à x_1 lorsque l'articulation 1 est prismatique. Ce choix n'est pas obligatoire, mais il rend les paramètres α_1 , d_1 et q_1 nuls ;
- de même, en définissant l'axe x_n du repère R_n comme étant colinéaire à x_{n-1} lorsque $q_n=0$, on rend le paramètre \bar{q}_n nul ;
- Pour une articulation j prismatique, l'axe z_j est parallèle à l'axe de l'articulation mais la position de cet axe dans l'espace peut être quelconque. On le place donc de telle sorte que d_j ou d_{j+1} soit nul ;
- Lorsque z_j est parallèle à z_{j+1} , on place x_j de telle sorte que r_j ou r_{j+1} soit nul ;
- Cette méthode de description fixe la configuration zéro « géométrique » du robot telle que $q=0$. Cette configuration ne correspond pas forcément à la configuration zéro « codeur ». On peut passer de l'une à l'autre en procédant au changement de variable suivant :

$$q = q_c + q_0 \quad [\text{II.5}]$$

Où q_0 représente le décalage (''offset'') introduit sur les variables codeurs q_c pour obtenir le vecteur des variables articulaires q ;

- lorsqu'une cinématique comporte deux ou plusieurs axes parallèles consécutifs, on peut se ramener à une seule matrice de transformation équivalente faisant intervenir la somme des variables articulaires. Par exemple si $\alpha_{j+1} = 0$, c'est-à-dire si z_j et z_{j+1} sont parallèles, la transformation directe s'écrit :

$$\begin{aligned}
{}^{j-1}T_{j+1} &= {}^{j-1}T_j {}^jT_{j+1} \\
&= \text{Rot}(x, \alpha_j) \text{Trans}(x, d_j) \text{Rot}(z, \theta_j) \text{Trans}(z, r_j) \\
&\quad \text{Tran}(x, d_{j+1}) \text{Rot}(z, \theta_{j+1}) \text{Trans}(z, r_{j+1}) \quad [\text{II.6}] \\
&= \begin{bmatrix} C(\theta_j + \theta_{j+1}) & -S(\theta_j + \theta_{j+1}) & 0 & d_j + d_{j+1} + C\theta_j \\ C\alpha_j S(\theta_j + \theta_{j+1}) & C\alpha_j C(\theta_j + \theta_{j+1}) & -S\alpha_j & d_{j+1} C\alpha_j - (r_j + r_{j+1}) S\alpha_j \\ S\alpha_j S(\theta_j + \theta_{j+1}) & S\alpha_j C(\theta_j + \theta_{j+1}) & S\alpha_j & d_{j+1} S\alpha_j S\theta_j - (r_j + r_{j+1}) C\alpha_j \end{bmatrix}
\end{aligned}$$

Et la transformation inverse a pour expression :

$${}^{j+1}T_{j+1} = \begin{bmatrix} & & & -dC(\theta_j + \theta_{j+1})d_{j+1}C\theta_{j+1} \\ & {}^{j-1}A_{j+1}^T & & d_j S(\theta_j + \theta_{j+1})d_{j+1}S\theta_{j+1} \\ 0 & 0 & 0 & -(r_j + r_{j+1}) \\ & & & 1 \end{bmatrix}$$

On voit apparaitre dans ces expressions des termes en $(\theta_j + \theta_{j+1})$ et en $(r_j + r_{j+1})$, résultats qui sont généralisables à m axes parallèles consécutifs.

II.3.2. Modèle géométrique direct

Le modèle géométrique direct (MGD) est l'ensemble de relations qui permettent d'exprimer la situation de l'organe terminal, c'est-à-dire les coordonnées opérationnelles du robot, en fonction de ces coordonnées articulaires. Dans le cas d'une chaîne ouverte simple, il peut être représenté par la matrice de passage 0T_n :

$${}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) \dots {}^{n-1}T_n(q_n) \quad [\text{II.7}]$$

Le modèle géométrique direct du robot peut aussi être représenté par la relation :

$$X = f(q) \quad [\text{II.8}]$$

Tel que q étant le vecteur des variables articulaires tel que :

$$q = [q_1 \ q_2 \ \dots \ q_n]^T$$

Les coordonnées opérationnelles sont définies par :

$$X = [x_1 \ x_2 \ \dots \ x_m]^T = [P_x \ P_y \ P_z \ n_x \ n_y \ n_z \ a_x \ a_y \ a_z]^T$$

II.3.3. Modèle géométrique inverse

Le modèle géométrique directe d'un robot permet de calculer les coordonnées opérationnelles donnant la situation de l'organe terminal en fonction des coordonnées articulaires. Le problème inverse consiste à calculer les coordonnées articulaires correspondant à une situation donnée de l'organe terminal. Lorsqu'elle existe, la forme

explicite qui donne toutes les solutions possibles (il ya rarement unicité de solution) constitue ce qu'on appelle le modèle géométrique inverse (MGI). Nous nous intéressant alors aux robots à structure ouverte simple.

Soit ${}^fT_E^d$ la matrice de transformation homogène représentant la situation désirée de repère outil R_E par rapport au repère atelier. Dans le cas général, on peut exprimer ${}^fT_E^d$ sous la forme :

$${}^fT_E^d = Z {}^0T_n(q) E \quad [\text{II.9}]$$

Avec :

- Z est la matrice de transformation homogène définissant la situation du robot (repère R_0) dans le repère atelier ;
- 0T_n est la matrice de transformation homogène du repère terminal R_n dans le repère R_0 , fonction du vecteur des variables articulaire q ;
- E est la matrice de transformation homogène définissant le repère outil R_E dans le repère terminal R_n .

Lorsque $n \geq 6$, on peut écrire la relation suivante en regroupant dans le membre de droite tous les termes connus :

$${}^0T_n(q) = Z^{-1} {}^fT_E^d E^{-1} \quad [\text{II.10}]$$

• Principe

Considérons un robot manipulateur dont la matrice de transformation homogène a pour expression :

$${}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) \dots {}^{n-1}T_n(q_n) \quad [\text{II.11}]$$

Soit U_0 la situation désirée telle que :

$$U_0 = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & n_x & a_x & 0 \\ s_y & n_y & a_y & 0 \\ s_z & n_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [\text{II.12}]$$

II.4.Modélisation cinématique (différentielle)

II.4.1.Modèle cinématique direct

Le modèle cinématique direct d'un robot manipulateur décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires. Il est noté

$$\dot{x} = J(q) \dot{q} \quad [\text{II.13}]$$

Où $J(q)$ désigne la matrice Jacobéenne ($m \times n$) du mécanisme, égale à : $\frac{\partial X}{\partial q}$ et fonction de la configuration articulaire q . La même matrice Jacobienne intervient dans le calcul du modèle différentiel directe qui donne la variation élémentaire dX des coordonnées opérationnelles en fonction des variations élémentaires des coordonnées articulaires dq , soit :

$$dX = J(q) dq \quad [\text{II.14}]$$

Intérêt de la matrice Jacobéenne :

- Elle est la base du modèle différentiel inverse, permettant de calculer une solution locale des variables articulaires q connaissant les coordonnées opérationnelles X .
- En statique, on utilise le Jacobien pour établir la relation liant les efforts exercés par l'organe terminale sur l'environnement aux forces et couples des actionneurs.
- Elle facilite le calcul des singularités et de la dimension de l'espace opérationnel accessible du robot.

- **Calcul de la Jacobienne par dérivation du MGD**

Le calcul de la matrice Jacobéenne peut se faire en dérivant le MGD, $X=f(q)$ à partir de la relation suivante :

$$J_{ij} = \frac{\partial f_i(q)}{\partial q_j} \quad \text{avec : } i=1 \dots m ; j=1 \dots m ; \quad [\text{II.15}]$$

Où J_{ij} est l'élément (i,j) de la matrice Jacobienne. Cette méthode est facile à mettre en œuvre pour les robots à deux ou trois degrés de liberté. Il existe une autre méthode basée sur le calcul de la jacobienne de base qui est pratique pour les robots ayant plus de degrés de liberté.

II.4.2. Modèle cinématique inverse

L'objectif du modèle cinématique inverse est de calculer, à partir d'une configuration q donnée, les vitesses articulaires \dot{q} qui assurent au repère terminal une vitesse opérationnelles \dot{X} imposée. Cette définition est analogue à celle du modèle différentiel inverse : ce dernier permet de déterminer la différentielle articulaire des coordonnées opérationnelles dX spécifiée. Pour obtenir le modèle cinématique inverse, on inverse le modèle cinématique direct en résolvant un système d'équations linéaires. La mise en œuvre peut être faite de façon analytique ou numérique :

- La solution analytique a pour avantage de diminuer considérablement le nombre d'opérations, mais on doit traiter séparément tous les cas singuliers ;

- Les méthodes numériques sont plus générales, la plus répandue étant fondée sur la notion de pseudo-inverse : les algorithmes traitent de façon unifiée les cas réguliers, singuliers et redondants. Elles nécessitent un temps de calcul relativement important.

- **Forme générale du modelé cinématique**

Quelle que soit la méthode utilisée pour décrire les coordonnées opérationnelles, le modelé cinématique direct peut être mis sous la forme :

$$\dot{X} = \begin{bmatrix} \Omega_p & O_3 \\ O_3 & \Omega_r \end{bmatrix} \begin{bmatrix} O A_i & O_3 \\ O_3 & O A_i \end{bmatrix} \begin{bmatrix} I_3 & -i\hat{L}_{j,n} \\ O_3 & I_3 \end{bmatrix} {}^i J_{n,j} \dot{q} \quad [\text{II.16}]$$

Ou sous forme considérée:

$$\dot{X} = {}^0 J_x \dot{q} \quad [\text{II.17}]$$

Etant donné la simplicité des éléments de ${}^i J_{n,j}$ comparés à ceux de ${}^0 J_x$, il est préférable de chercher une solution analytique à partir de l'expression précédente. Celle-ci s'écrit encore :

$${}^i \dot{X}_{n,j} = {}^i J_{n,j} \dot{q}$$

avec :

$${}^i \dot{X}_{n,j} = \begin{bmatrix} I_3 & i\hat{L}_{j,n} \\ O_3 & I_3 \end{bmatrix} \begin{bmatrix} iA_0 & O_3 \\ O_3 & iA_0 \end{bmatrix} \begin{bmatrix} \Omega_p^{-1} & O \\ O_3 & \Omega_r^+ \end{bmatrix} \dot{X} \quad [\text{II.18}]$$

Si on utilise les coordonnées cartésiennes pour décrire la position, alors $\Omega_p^{-1} = I_3$. L'inverse à gauche Ω_r^+ .

Pour alléger l'écriture, nous considérons la forme allégée :

$$\dot{X} = J \dot{q} \quad [\text{II.19}]$$

Calcul de la jacobienne

Dans le cas régulier, la matrice jacobienne est carrée d'ordre n et son déterminant est non nul. Deux méthodes de calcul peuvent être mises en oeuvre.

- **Première méthode**

On calcul J^{-1} , la matrice inverse de J, qui permet de déterminer les vitesses articulaires \dot{q}
 Grace à la relation: $\dot{q} = J^{-1} \dot{X}$ [II.20]

Lorsque la matrice J a la forme suivante:

$$J = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix} \quad [\text{II.21}]$$

Les matrices A et C étant carrées inversibles, il est facile de montrer que l'inverse de cette matrice s'écrit :

$$J^{-1} = \begin{bmatrix} A^{-1} & 0 \\ C^{-1}BA^{-1} & C^{-1} \end{bmatrix} \quad [\text{II.22}]$$

La résolution de problème se ramène donc à l'inversion, beaucoup plus simple, de deux matrices de dimension moindre. Lorsque le robot- manipulateur possède six degrés de liberté et un poignet de type rotule, la forme générale de J est celle de la relation [II.21], A et C étant de dimension (3×3) .

- **Seconde méthode**

Dans cette méthode, on tient compte d'une éventuelle forme particulière de la matrice J permettant de réduire le nombre d'inconnus à prendre en compte simultanément. Cette méthode donne, dans la plupart des cas, des solutions nécessitant moins d'opérations.

Considérons la structure suivante :

$$\begin{bmatrix} \dot{X}_a \\ \dot{X}_b \end{bmatrix} = \begin{bmatrix} A & O_3 \\ B & C \end{bmatrix} \begin{bmatrix} \dot{q}_a \\ \dot{q}_b \end{bmatrix} \quad [\text{II.23}]$$

A et C étant des matrices carrées de dimension (3×3) , inversibles en dehors des positions singulières.

La solution \dot{q} est donnée par :

$$\begin{cases} \dot{q}_a = A^{-1} \dot{X}_a \\ \dot{q}_b = C^{-1} [\dot{X}_b - B\dot{q}] \end{cases} \quad [\text{II.24}]$$

II.5. Modélisation dynamique

Le modèle dynamique est la relation entre les couples (et/ou forces) appliqués aux actionneurs et les positions, vitesses et accélérations articulaires. On représente le modèle dynamique par une relation de la forme :

$$\Gamma = f(q, \dot{q}, \ddot{q}, \ddot{i}_e) \quad [\text{II.25}]$$

Avec :

- Γ : Vecteur des couples/forces des actionneurs, selon que l'articulation est rotoïde ou prismatique. Dans la suite, on écrira tout simplement couples ;
- q : vecteur de positions articulaires ;
- \dot{q} : vecteur des vitesses articulaires ;
- \ddot{q} : vecteur des accélérations articulaires ;
- \ddot{i}_e : vecteur représentant l'effort extérieur (forces et moments) qu'exerce le robot sur l'environnement.

On convient d'appeler modèle dynamique inverse, ou tout simplement modèle dynamique, la relation de la forme [II.25].

Le modèle dynamique direct est celui qui exprime les accélérations articulaires en fonction des positions, vitesses et couples des articulations. Il est alors représenté par la relation :

$$\ddot{\mathbf{q}} = \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, \Gamma, \ddot{\mathbf{e}}) \quad [\text{II.26}]$$

Parmi les applications du modèle dynamique, on peut citer :

- la simulation, qui utilise le modèle dynamique direct ;
- le dimensionnement des actionneurs ;
- l'identification des paramètres inertiels et des paramètres de frottement du robot;
- la commande, qui utilise le modèle dynamique inverse ;

Plusieurs formalismes ont été utilisés pour obtenir le modèle dynamique des robots. Les formalismes les plus souvent utilisés sont :

- a) Le formalisme de Lagrange
- b) Le formalisme de Newton-Euler

Remarque :

Les notations des variables de modèle dynamique sont illustrées dans (l'annexe A)

II.5.1. Formalisme de Lagrange

Le but de ce paragraphe est d'étudier la forme générale du modèle dynamique, mettre en évidence les différents termes qui y interviennent de déduire les propriétés caractéristique de ces termes. La méthode présentée n'est pas celle qui donne le modèle le plus performant du point vue du nombre d'opérations, mais c'est le modèle le plus simple compte tenu de ces objectifs dans le premier temps, nous considérons un robot idéal sans frottement, sans élasticité et ne subissant ou n'exerçant aucun effort extérieur.

Ces facteurs seront pris en compte par la suite. Le formalisme de Lagrange décrit les équations du mouvement en terme de travail et d'énergie du système, ce qui se traduit, lorsque l'effort extérieur sur l'organe terminal est supposé nul, par l'équation suivante :

$$\Gamma_i = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q} \quad i=1, \dots, n \quad [\text{II.27}]$$

Avec:

- L : lagrangien du système égal à $E - U$;
- E : énergie cinétique totale du système ;
- U : énergie potentielle totale du système,

II.5.1.1. Forme générale des équations dynamiques

L'énergie cinétique du système est une fonction quadratique des vitesses articulaires :

$$E = \frac{1}{2} \dot{q}^T M q \quad [II.28]$$

Où M est la matrice ($n \times n$) de l'énergie cinétique, d'élément générique M_{ij} , appelée aussi matrice d'inertie du robot, qui est symétrique et définie positive. Ses éléments sont fonction des variables articulaires q .

Energie potentielle étant fonction des variables articulaires q , le couple Γ peut se mettre, à partir des équations [II.28] et [II.29], sous la forme :

$$\Gamma = M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) \quad [II.29]$$

avec :

- $C(q, \dot{q}) \dot{q}$: vecteur de dimension ($n \times 1$) représentant les couples/forces de Coriolis et des forces centrifuges, tel que :

$$C \dot{q} = \dot{M} \dot{q} - \frac{\partial E}{\partial q} \quad [II.30]$$

- $G = [G_1 \dots G_n]^T$: vecteur des couples /forces de gravité.

Plusieurs formes sont possibles pour la matrice C . On peut par exemple calculer ses éléments à partir du symbole de « Cristophell » $c_{i,jk}$ tel que :

$$\begin{cases} C_{i,j} = \sum_{k=1}^n c_{i,jk} \dot{q}_k \\ C_{i,jk} = \frac{1}{2} \left[\frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{jk}}{\partial q_i} \right] \end{cases} \quad [II.31]$$

Les éléments du vecteur G se calculent en écrivant que :

$$G_i = \frac{\partial U}{\partial q_i} \quad [II.32]$$

Les éléments du vecteur M, C, G sont en fonction des paramètres géométriques et inertiels du mécanisme. Les équations dynamiques d'un système mécanique articulé forment donc un système de n équations différentielles du second ordre. Couplés et non linéaires.

Pour calculer les éléments M, C et G , il faut tout d'abord calculer les énergies cinétique et potentielle de tout les corps du robot. On procède ensuite comme suite :

- L'élément M_{ij} est égal au coefficient de $(\dot{q}_i^2/2)$ dans l'expression de l'énergie cinétique, tant dis que l'élément M_{ij} , si $i \neq j$, est égal au coefficient de $\dot{q}_i \dot{q}_j$;
- Le calcul de C se fait selon la relation [II.31] ;
- Le calcul de G se fait selon la relation [II.32]

II.5.1.2. Calcul de l'énergie cinétique et potentielle

L'énergie cinétique totale du système est donnée par la relation :

$$E = \sum_{j=1}^n E_j \quad [\text{II.33}]$$

Où E_j désigne l'énergie cinétique du corps C_j , qui s'exprime par :

$$E_j = \frac{1}{2} (\omega_j^T I_{G_j} \omega_j + M_j V_{G_j}^T V_{G_j}) \quad [\text{II.34}]$$

D'autre part l'énergie potentielle s'écrit :

$$U = \sum_{j=1}^n U_j = \sum_{j=1}^n -M_j g^T (L_{0,j} + S_j) \quad [\text{II.35}]$$

L'énergie cinétique et potentielle étant linéaire par rapport aux paramètres inertiels, le modèle dynamique l'est également.

II.5.2. Formalisme de Newton-Euler

Les équations de Newton-Euler expriment le torseur dynamique en G_j des efforts extérieurs sur un corps j par les équations :

$$F_j = M_j \dot{V}_{G_j} \quad [\text{II.36}]$$

$$M_{G_j} = I_{G_j} \dot{\omega}_j + \omega_j \times (I_{G_j} \omega_j) \quad [\text{II.37}]$$

La méthode de Luh, Walker et Paul confédérées comme une avancée importante vers la possibilité de calculer le modèle dynamique des robots en ligne, utilise ces équations et est fondée sur une double récurrence.

La récurrence avant, de la base du robot vers l'effecteur, calcule successivement les vitesses et accélérations des corps, puis le torseur dynamique.

La récurrence arrière, l'effecteur arrière, de l'effecteur vers la base, permet le calcul des couples des actionneurs en exprimant pour chaque corps le bilan des efforts. Cette méthode permet d'obtenir directement le modèle dynamique inverse sans avoir à calculer explicitement les matrices M , C et G . Les paramètres inertiels utilisés sont M_j , S_j et I_{G_j} , le modèle ainsi obtenu n'est pas linéaire par rapport aux paramètres inertiels.

II.5.2.1. Equations de Newton-Euler linéaires par rapport aux paramètres inertiels

L'algorithme de Newton-Euler suivant est fondée sur la double récurrence de Luh mais exprimant le torseur dynamique des efforts extérieur en O_j plutôt qu'en G_j , utilisant les paramètres inertiels de bases.

$$F_j = M_j \dot{V}_j + \dot{\omega}_j \times MS_j + \omega_j \times (\omega_j \times MS_j) \quad [\text{II.38}]$$

$$M_j = J_j \dot{\omega}_j + \omega_j \times (J_j \omega_j) + MS_j \times \dot{V}_j \quad [\text{II.39}]$$

En utilisant les notations des torseurs, on obtient :

$$F_j = J_j \dot{V}_j + \begin{bmatrix} \omega_j \times (\omega_j \times MS_j) \\ \omega_j \times (J_j \omega_j) \end{bmatrix} \quad \text{avec :} \quad F_j = \begin{bmatrix} F_j \\ M_j \end{bmatrix}, \quad \dot{V} = \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix}$$

II.5.2.2. Forme parasitique des équations Newton-Euler

Pour utiliser pratiquement l'algorithme de Newton-Euler exposé ci-dessus. Il faut projeter dans un même repère les vecteurs en tenseurs qui apparaissent dans une même équation. Les équations de récurrences sont données par les expressions suivantes :

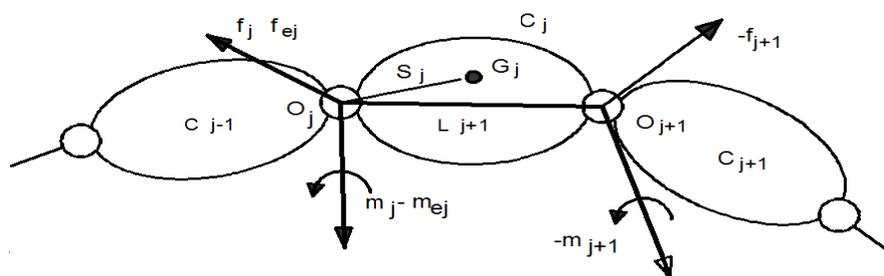


Figure II-3 : Bilan des efforts au centre de gravité

a) récurrence avant, pour $j = 1, \dots, n$;

$${}^j\omega_{j-1} = {}^jA_{j-1} {}^{j-1}\omega_{j-1}$$

$${}^j\omega_j = {}^j\omega_{j-1} + \bar{\sigma}_j \dot{q}_j {}^j a_j$$

$${}^j\dot{\omega}_j = {}^jA_{j-1} {}^{j-1}\dot{\omega}_{j-1} + \bar{\sigma}_j (\ddot{q}_j {}^j a_j + {}^j\omega_{j-1} \times \dot{q}_j {}^j a_j)$$

$${}^j\dot{V}_j = {}^jA_{j-1} ({}^{j-1}\dot{V}_{j-1} + {}^{j-1}U_{j-1} {}^{j-1}P_j) + \sigma_j (\ddot{q}_j {}^j a_j + 2 {}^j\omega_{j-1} \times \dot{q}_j {}^j a_j)$$

$${}^jF_j = M_j {}^j\dot{V}_j + {}^jU_j {}^jMS_j$$

$${}^jM_j = {}^jJ_j {}^j\dot{\omega}_j + {}^j\omega_j \times ({}^jJ_j {}^j\omega_j) + {}^jMS_j \times {}^j\dot{V}_j$$

$$\text{Avec : } \omega_0 = 0, \dot{\omega}_0 = 0, \dot{V}_0 = -g \text{ et } {}^jU_j = {}^j\dot{\omega}_j + {}^j\omega_j \times {}^j\omega_j$$

L'introduction de la matrice jU_j , ainsi que l'utilisation de certains de ses éléments dans le calcul de jM_j , permet un gain de $21n$ multiplications et $6n$ additions dans le cas d'un robot général.

b) pour récurrence arrière, $j = n, \dots, 1$

$${}^j f_j = {}^j F_j + {}^j f_{j+1} + {}^j f_{e_j}$$

$${}^{j-1} f_j = {}^{j-1} A {}^j f_j$$

$${}^j m_j = {}^j M_j + {}^j A_{j+1} {}^{j+1} m_{j+1} + {}^j P_{j+1} \times {}^j f_{j+1} + {}^j m_{e_j}$$

$$\Gamma_j = (\sigma_j {}^j f_j + \bar{\sigma}_j {}^j m_j)^T {}^j a_j + F_{s_j} \text{Singn}(\dot{q}_j) + F_{v_j} \dot{q}_j + I_{a_j} \ddot{q}_j$$

II.6. Aperçu sur la modélisation dynamique des robots manipulateurs à chaîne ouverte simple

Un robot manipulateur est défini comme une structure mécanique articulée, rigide à chaîne ouverte simple. En utilisant le principe de modélisation par le formalisme de Lagrange, le modèle dynamique est décrit par le système à n équations différentielles du second ordre, non linéaires et couplées suivant :

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + F_v \dot{q} + G(q) = \Gamma \quad [\text{II.40}]$$

Où $q, \dot{q}, \ddot{q} \in R^n$ sont respectivement les vecteurs des positions, vitesses et accélérations articulaires. $\Gamma \in R^n$ est le vecteur des couples des actionneurs. $M(q) \in R^{n \times n}$ est la matrice d'inertie du robot. Elle est symétrique et définie positive. $C(q, \dot{q}) \dot{q} \in R^n$ est le vecteur des termes centrifuges et de Coriolis. $F_v \in R^{n \times n}$ est la matrice des coefficients des frottements visqueux. $G(q) \in R^n$ est le vecteur des forces de gravités.

Quatre propriétés structurelles peuvent être mentionnées pour chacun des termes de l'équation différentielle (II.41) du robot :

$$\mathbf{P1} : \varphi_1 I_n \leq M(q) \leq \varphi_2 I_n \text{ avec } \varphi_1 \text{ et } \varphi_2 \text{ deux constantes strictement positives.} \quad [\text{II.41}]$$

$$\mathbf{P2} : \dot{M}(q) - 2C(q, \dot{q}) \text{ est une matrice antisymétrique, c à d :}$$

$$x^T [\dot{M}(q) - 2C(q, \dot{q})]x = 0, \quad \forall x \in R^n \quad [\text{II.42}]$$

$$\mathbf{P3} : \|G(q)\| \leq g_{\max} \quad [\text{II.43}]$$

$$\mathbf{P4} : \|F_v \dot{q} + F_s(q)\| \leq \nu_1 \|\dot{q}\| + \nu_2 \text{ où } F_s(q) \text{ est le vecteur des frottements secs,} \quad [\text{II.44}]$$

$$\text{avec } \nu_1, \nu_2 \succ 0$$

II.6.1. Modélisation du robot à deux degré de liberté

Nous développons dans cette partie le formalisme de Lagrange pour le calcul du modèle dynamique d'un robot manipulateur et ceci revient à calculer la matrice d'inertie $M(q)$, la matrice de Coriolis $C(q, \dot{q})$ et la matrice de Gravité $G(q)$.

L'équation du modèle dynamique sous une forme matricielle d'un robot rigide en tenant compte des actionneurs ainsi que les frottements secs et visqueux, est donnée par (II.40). en considérant que les frottements sont négligeables $F_v \approx 0$.

$$\Rightarrow \Gamma(q) = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \quad [\text{II.45}]$$

II.6.1.1 Calcul des matrices de passages

Les matrices de transformation homogène du premier et du deuxième segment par rapport au repère non inertiel et du deuxième par rapport au repère inertiel, sont respectivement données par :

Passage de $R_0 \rightarrow R_1$:

$${}^0T_1 = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & 0 \\ \sin(q_1) & \cos(q_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [\text{II.46}]$$

Passage de $R_1 \rightarrow R_2$:

$${}^1T_2 = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & l_1 \\ \sin(q_2) & \cos(q_2) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [\text{II.47}]$$

Passage de $R_0 \rightarrow R_2$:

$${}^0T_2 = \begin{bmatrix} \cos(q_1 + q_2) & -\sin(q_1 + q_2) & 0 & l_1 \cos(q_1) \\ \sin(q_1 + q_2) & \cos(q_1 + q_2) & 0 & l_1 \sin(q_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [\text{II.48}]$$

II.6.1. 2 Calcul des tenseurs d'inertie

La matrice de pseudo-inertie I_1 du corps C_1 en O_1 est donnée par (II.49):

$$I_i = \begin{bmatrix} \int x^2 dm & \int yx dm & \int zx dm & \int x dm \\ \int xy dm & \int y^2 dm & \int zy dm & \int y dm \\ \int xz dm & \int yz dm & \int z^2 dm & \int z dm \\ \int x dm & \int y dm & \int z dm & \int dm \end{bmatrix} \quad [\text{II.49}]$$

$$I_i = \begin{bmatrix} \int x^2 dm & 0 & 0 & m_1 s_1 \\ 0 & \int y^2 dm & 0 & 0 \\ 0 & 0 & \int z^2 dm & 0 \\ m_1 s_1 & 0 & 0 & m_1 \end{bmatrix} \quad [\text{II.50}]$$

La matrice de pseudo-inertie I_2 du corps C_2 en O_2 est donnée par (II.49) :

$$I_2 = \begin{bmatrix} \int x^2 dm & 0 & 0 & m_2 s_2 \\ 0 & \int y^2 dm & 0 & 0 \\ 0 & 0 & \int z^2 dm & 0 \\ m_2 s_2 & 0 & 0 & m_2 \end{bmatrix} \quad [\text{II.51}]$$

II.6.1.3 Calcul de la contribution du corps C_1 à la matrice d'inertie

$$m_{jk}(q) = \sum_{i=1}^n \text{trace} \left[\frac{\partial {}^oT_i}{\partial q_j} I_i \frac{\partial {}^oT_i^T}{\partial q_k} \right] \quad [\text{II.52}]$$

Les éléments $m_{11}(q)$, $m_{21}(q)$ et $m_{22}(q)$ sont données respectivement par (II.52) :

$$m_{11}(q) = J_1(O_1) \quad [\text{II.53.1}]$$

$$m_{21}(q) = 0 \quad [\text{II.53.2}]$$

$$m_{22}(q) = 0 \quad [\text{II.53.3}]$$

Où $J_1(O_1)$ est l'inertie par rapport à l'axe (O_1, \vec{Z}_1) de C_1 .

I.6.1.4 Calcul de la contribution du corps C_2 à la matrice d'inertie

Les éléments $m_{11}(q)$, $m_{21}(q)$ et $m_{22}(q)$ sont données respectivement par (II.52) :

$$m_{11}(q) = m_2 I_1^2 + J_2(O_2) + 2 m_2 I_1 s_2 \cos(q_2) \quad [\text{II.54.1}]$$

$$m_{21}(q) = J_2(O_2) + m_2 I_1 s_2 \cos(q_2) \quad [\text{II.54.2}]$$

$$m_{22}(q) = J_2(O_2) \quad [\text{II.54.3}]$$

Où $J_2(O_2)$ est l'inertie par rapport à l'axe (O_2, \vec{Z}_2) de C_2 .

II.6.1.5 Calcul de la matrice d'inertie du robot

En exploitant les relations (II.53) et (II.54), nous déterminons l'expression de la matrice d'inertie, elle est donnée par :

$$M(q) = \begin{bmatrix} J_1(O_1) + J_2(O_2) + m_2 l_1^2 + 2m_2 l_1 s_2 \cos(q_2) & J_2(O_2) + m_2 l_1 s_2 \cos(q_2) \\ J_2(O_2) + m_2 l_1 s_2 \cos(q_2) & J_2(O_2) \end{bmatrix} \quad [\text{II.55}]$$

II.6.1.6 Calcul de la matrice de Coriolis du robot

L'utilisation des symboles de Christoffel $C_{kj,i}(q, \dot{q})$ permet d'écrire :

$$C_{ij}(q, \dot{q}) = \frac{1}{2} \left\{ \frac{\partial m_{ij}(q)}{\partial q_k} + \frac{\partial m_{ki}(q)}{\partial q_j} - \frac{\partial m_{jk}(q)}{\partial q_i} \right\} \dot{q}_k = C_{kj,i}(q, \dot{q}) \dot{q}_k \quad [\text{II.56}]$$

Se basant sur les relations (II.55) et (II.56) nous calculons les coefficients dynamiques $C_{ij}(q, \dot{q})$ ($i, j=1,2$) de la matrice $C(q, \dot{q}) \in \mathfrak{R}^{2 \times 2}$. Ces coefficients sont définis par les relations suivantes :

$$\begin{aligned} C_{1,1}(q, \dot{q}) &= \frac{1}{2} \left[\dot{q}^T \frac{\partial \dot{m}_{11}(q)}{\partial q} + \left(\frac{\partial m_{11}(q)}{\partial q_1} - \frac{\partial m_{11}(q)}{\partial q_1} \right) \dot{q}_1 + \left(\frac{\partial m_{12}(q)}{\partial q_1} - \frac{\partial m_{12}(q)}{\partial q_1} \right) \dot{q}_2 \right] \\ &= -m_2 l_1 s_2 \dot{q}_2 \sin(q_2) \end{aligned} \quad [\text{II.57.1}]$$

$$C_{1,2}(q, \dot{q}) = \frac{1}{2} \left[\dot{q}^T \frac{\partial m_{12}(q)}{\partial q} + \left(\frac{\partial m_{11}(q)}{\partial q_2} - \frac{\partial m_{21}(q)}{\partial q_1} \right) \dot{q}_1 + \left(\frac{\partial m_{12}(q)}{\partial q_2} - \frac{\partial m_{22}(q)}{\partial q_1} \right) \dot{q}_2 \right]$$

$$= -m_2 l_1 s_2 (\dot{q}_1 + \dot{q}_2) \sin(q_2) \quad [\text{II.57.2}]$$

$$C_{2,1}(q, \dot{q}) = \frac{1}{2} \left[\dot{q}^T \frac{\partial m_{21}(q)}{\partial q} + \left(\frac{\partial m_{21}(q)}{\partial q_2} - \frac{\partial m_{11}(q)}{\partial q_1} \right) \dot{q}_1 + \left(\frac{\partial m_{22}(q)}{\partial q_2} - \frac{\partial m_{12}(q)}{\partial q_1} \right) \dot{q}_2 \right]$$

$$= m_2 l_1 s_2 \dot{q}_1 \sin(q_2) \quad [\text{II.57.3}]$$

$$C_{2,2}(q, \dot{q}) = \frac{1}{2} \left[\dot{q}^T \frac{\partial m_{22}(q)}{\partial q} + \left(\frac{\partial m_{21}(q)}{\partial q_2} - \frac{\partial m_{21}(q)}{\partial q_1} \right) \dot{q}_1 + \left(\frac{\partial m_{22}(q)}{\partial q_2} - \frac{\partial m_{22}(q)}{\partial q_1} \right) \dot{q}_2 \right]$$

$$= 0 \quad [\text{II.57.4}]$$

$$\Rightarrow C(q, \dot{q}) = \begin{bmatrix} -m_2 l_1 s_2 \dot{q}_2 \sin(q_2) & -m_2 l_1 s_2 (\dot{q}_1 + \dot{q}_2) \sin(q_2) \\ m_2 l_1 s_2 \dot{q}_1 \sin(q_2) & 0 \end{bmatrix} \quad [\text{II.57}]$$

II.6.1.7 Calcul de l'énergie potentielle totale

L'énergie potentielle d'un robot manipulateur est donnée par

$$U = \sum_{j=1}^n U_j \quad [\text{II.58}]$$

Où U_j désigne l'énergie potentielle d'un corps C_j qui est définie par :

$$U_j = -m_j {}^0 \mathbf{g}^T {}^0 \mathbf{p}_j \quad [\text{II.59}]$$

Avec :

${}^0 \mathbf{g}$: vecteur $\in \mathcal{R}^3$ de l'accélération de la gravité exprimée dans le repère R_0 de la base du robot.

${}^0 \mathbf{P}_j$: vecteur $\in \mathcal{R}^3$ des coordonnées du centre de masse du corps C_j par rapport au repère R_0

Etant donné que le vecteur ${}^0 \mathbf{P}_j$ est fonction des positions articulaires q , l'énergie potentielle U dépend alors de la variable articulaire q .

L'énergie potentielle de chaque C_j ($j=1,2$) s'obtient à partir de relation (58) (59) :

$$U_1 = -m_1 {}^0 \mathbf{g}^T {}^0 \mathbf{p}_1 = -m_1 s_1 g \cos(q_1) \quad [\text{II.60}]$$

$$U_2 = -m_2 {}^0 \mathbf{g}^T {}^0 \mathbf{p}_2 = -m_2 l_1 g \cos(q_1) - m_2 s_2 g \cos(q_1 + q_2)$$

D'où l'expression de l'énergie potentielle totale :

$$U = -g[(m_1 s_1 + m_2 l_1) \cos(q_1) + m_2 s_2 \cos(q_1 + q_2)] \quad [\text{II.61}]$$

II.6.1. 8 Calcul du vecteur des forces de gravité

Les vecteurs des termes de gravité se déduit à partir de l'énergie potentielle totale (g) en utilisant le fait que $i^{\text{ème}}$ coordonnée du vecteur $G(q)$ est donnée par $\frac{\partial U}{\partial q_i}$

Les éléments du vecteur $G(q) \in \mathbb{R}^2$ sont définie par :

$$G_1(q) = \frac{\partial U}{\partial q_1} = g(m_1 s_1 + m_2 l_1) \sin(q_1) + m_2 s_2 \sin(q_1 + q_2) \quad [\text{II.62}]$$

$$G_2(q) = \frac{\partial U}{\partial q_2} = g [m_2 s_2 \sin(q_1 + q_2)]$$

II. 6.2. Modèle dynamique du robot à 2 ddl

L'équation du modèle dynamique sous une forme matricielle de ce robot :

$$M(q) = \begin{bmatrix} 8.77 + 1.02 \cos(q_2) & 0.76 + 0.51 \cos(q_2) \\ 0.76 + 0.51 \cos(q_2) & 0.62 \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} -0.5 \sin(q_2) \dot{q}_2 & -0.5 \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\ 0.5 \sin(q_2) \dot{q}_1 & 0 \end{bmatrix}$$

$$G(q) = 10 \begin{bmatrix} 7.6 \sin(q_1) + 0.63 \sin(q_1 + q_2) \\ 0.63 \sin(q_1 + q_2) \end{bmatrix}$$

II.7. Quelques lois de commande classique des robots manipulateurs

La commande des bras manipulateurs a fait l'objet de nombreux travaux, parmi les principales approches utilisées, la commande classique du PID, la commande de poursuite de trajectoire, la commande adaptative, la commande robuste à structure variable (mode glissant), et la commande point à point développée dans la suite de ce mémoire.

Le problème de la commande d'un robot manipulateur peut être formulé comme la détermination de l'évolution des forces généralisées (forces ou couples) que les actionneurs doivent exercer pour garantir l'exécution de la tâche tout en satisfaisant certains critères de performance.

II.7.1. Loi de commande proportionnelle dérivée avec compensation de l'effet de gravité (point à point)

Cette loi de commande permet de positionner les différentes articulations du robot manipulateur autour d'une position désirée constante, de plus elle permet de lever la contrainte relative à l'utilisation de grand gain dans d'autres lois de commande, et ceci par la compensation de l'effet du vecteur de gravité.

Considérant le modèle dynamique sans frottement et sans perturbation décrit par l'équation suivante :

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \quad [\text{II.63}]$$

Ce modèle décrit un système à n équations différentielles du second ordre non linéaires et couplées. n étant le nombre d'articulations. La loi de commande est définie par :

$$\Gamma = G(q) - k_p \tilde{q} - k_v \dot{\tilde{q}} \quad [\text{II.64}]$$

Où k_p et k_v sont respectivement les matrices des gains proportionnels et dérivés de dimension $(n \times n)$, ces deux matrices sont diagonales et définies positives.

$\tilde{q} = q - q_d$ représente l'écart de position, avec q (mesurée) est la position angulaire, et q_d la position désirée. $\dot{\tilde{q}} = \dot{q} - \dot{q}_d = \dot{q}$ est l'écart en vitesse.

Le calcul de la dynamique en boucle fermée nous donne la relation suivante :

$$\ddot{\tilde{q}} = -M(q) \left[(C(q, \dot{q}) + k_v) \dot{\tilde{q}} + k_p \tilde{q} \right] \quad [\text{II.65}]$$

Avec $\ddot{\tilde{q}} = \ddot{q} - \ddot{q}_d = \ddot{q}$ l'écart en accélération.

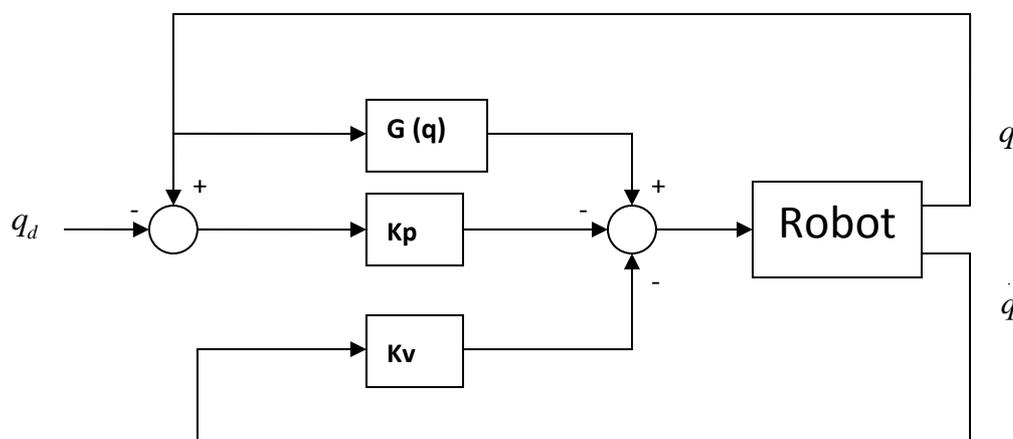


Figure II-4 : Schéma bloc d'une commande point à point

II.7.2. Commande en poursuite de trajectoire

II.7.2.1. Générateur de trajectoire

Afin d'étudier les lois de commandes il est nécessaire d'étudier les trajectoires de référence utilisées en robotique, ce qui nous permet d'analyser la validité d'une loi de commande.

D'une manière générale le problème de la génération de mouvement est de calculer pour chaque articulation les trajectoires de référence en position, vitesse et accélération qui sont fonction du temps et qui assurent le passage du robot par une trajectoire désirée d'une configuration initiale à une configuration finale désirée.

On peut distinguer les classes de mouvement suivantes :

- Le mouvement entre deux points avec trajectoire libre.
- Le mouvement entre deux points via des points intermédiaires, spécifiés notamment pour éviter les obstacles, avec trajectoire libre.
- Le mouvement entre deux points avec trajectoire.
- Le mouvement entre deux points via des points intermédiaires avec trajectoire spécifiée.

Considérons un robot à n degré de liberté où:

$q^i = [q_1^i \dots q_n^i]^T$ représente le vecteur des positions initiales, et $q^f = [q_1^f \dots q_n^f]^T$ le vecteur des positions finales.

Le mouvement pour passer de q^i à q^f s'écrit :

$$q_d(t) = q^i + r(t)D \quad [\text{II.66}]$$

avec $D = q^f - q^i$

$r(t)$ est le polynôme d'interpolation.

Plusieurs fonctions permettent de satisfaire le passage de q^i à $t = 0$ vers q^f à $t = t_f$, tel que l'interpolation polynomiale et l'interpolation bang-bang.

II.7.2.1.1. Interpolation polynomiale

Il existe plusieurs modes d'interpolation polynomiale parmi lesquels l'interpolation linéaire, l'interpolation par des polynômes de degrés trois et de degré cinq.

a- Interpolation linéaire

Il s'agit de l'interpolation la plus simple, où le mouvement de chaque articulation est décrit par une équation linéaire en temps. L'équation du mouvement s'écrit :

$$q_d(t) = q^i + r(t)D \quad \text{avec } r(t) = \frac{t}{t_f} \quad [\text{II.67}]$$

Ainsi $\dot{q}_d(t) = \dot{r}(t)D$ avec $\dot{r}(t) = \frac{1}{t_f}$ [II.68]

et $\ddot{q}_d(t) = \ddot{r}(t)D$ avec $\ddot{r}(t) = 0$ [II.69]

Les trajectoires pour ce type d'interpolation sont continues en position mais discontinues en vitesse tel qu'il est montré sur la figure III-3 pour une articulation quelconque j . C'est la raison pour laquelle ce type de mouvement est inacceptable sur les robots réels.

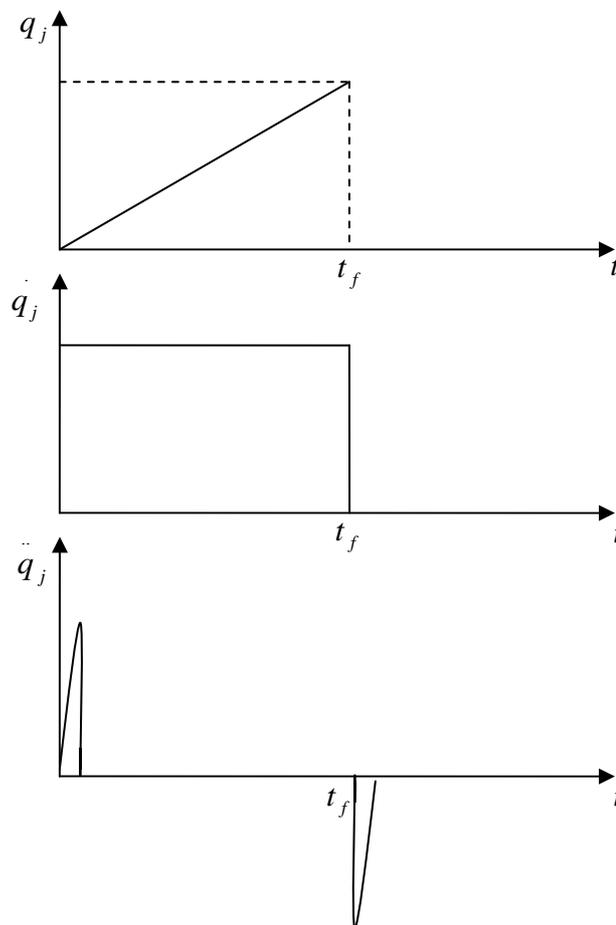


Figure II-5 : Polynôme d'interpolation linéaire.

b- Interpolation polynomiale de degré 3

Si l'on impose une vitesse nulle aux points de départ et d'arrivée, on ajoute deux contraintes aux deux contraintes de position. Le degré minimal du polynôme qui satisfait ces

quatre contraintes est de degré trois et a pour forme générale:

$$q_d(t) = q^i + \left[3\left(\frac{t}{t_f}\right)^2 - 2\left(\frac{t}{t_f}\right)^3 \right] D \quad [\text{II.70}]$$

$$\text{ainsi} \quad \dot{q}_d(t) = \left[6\left(\frac{t}{t_f}\right) - 6\left(\frac{t}{t_f}\right)^2 \right] D \quad [\text{II.71}]$$

$$\text{et} \quad \ddot{q}_d(t) = \left[\frac{6}{t_f} - 12\left(\frac{t}{t_f}\right) \right] D \quad [\text{II.72}]$$

La figure II-7 montre l'évolution des positions vitesses et accélérations pour n'importe quelle articulation j . Ce type d'interpolation assure la continuité des trajectoires en positions et vitesses, mais pas celle des accélérations.

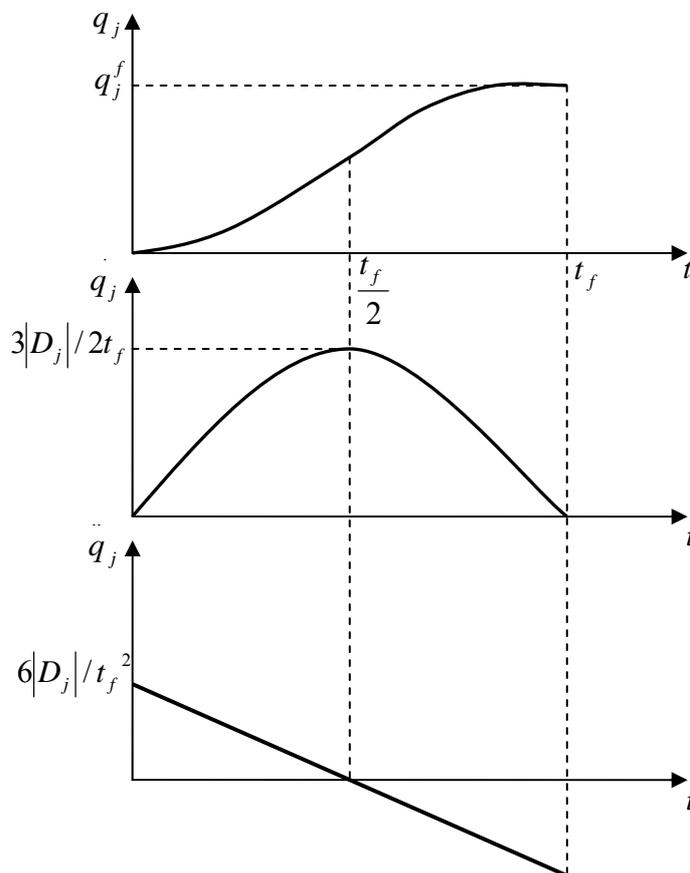


Figure II-6 : Polynôme d'interpolation de degré 3

La vitesse est maximale lorsque $t = t_f / 2$, Elle vaut donc :

$$\left| \dot{q}_{f \max} \right| = \frac{3|D_j|}{2t_f} \quad \text{Avec} \quad |D_j| = |q_j^f - q_j^i| \quad [\text{II.73}]$$

L'accélération est maximale à $t = 0$ et $t = t_f$, elle a pour valeur:

$$\left| \ddot{q}_{f \max} \right| = \frac{6|D_j|}{t_f^2} \quad [\text{II.74}]$$

c- Interpolation polynomiale de degré 5

L'interpolation polynomiale de degré cinq assure des trajectoires continues en positions, vitesses et accélérations. Le polynôme est obtenu en employant les conditions aux limites suivantes :

$$q(0) = q_i, \quad q(t_f) = q_f, \quad \dot{q}(0) = 0, \quad \dot{q}(t_f) = 0, \quad \ddot{q}(0) = 0, \quad \ddot{q}(t_f) = 0$$

Pour satisfaire les six contraintes, le polynôme d'interpolation doit être de degré cinq :

$$r(t) = 10\left(\frac{t}{t_f}\right)^3 - 15\left(\frac{t}{t_f}\right)^4 + 6\left(\frac{t}{t_f}\right)^5 \quad [\text{II.75}]$$

Les trajectoires des positions, vitesses et accélérations pour l'articulation j sont présentées à la figure II-8. Les vitesses et accélérations maximales ont pour expressions :

$$\left| \dot{q}_{f \max} \right| = \frac{15|D_j|}{8t_f} \quad [\text{II.76}]$$

$$\left| \ddot{q}_{f \max} \right| = \frac{10|D_j|}{\sqrt{3}t_f^2} \quad [\text{II.77}]$$

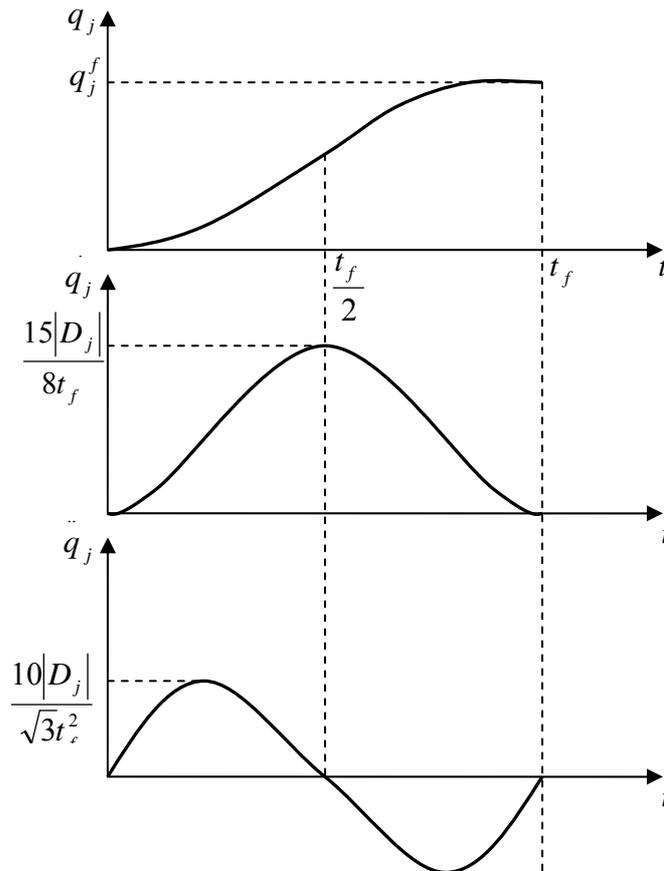


Figure II-7 : Polynôme d'interpolation de degré 5

II.7.2.1.2. Interpolation Bang-Bang

Le mouvement est matérialisé par une phase d'accélération jusqu'à $t_f/2$ et de décélération de $t_f/2$ à t_f

$$\begin{cases} q_d(t) = q^i + 2\left(\frac{t}{t_f}\right)^2 D & 0 \leq t \leq t_f/2 \\ q_d(t) = q^i + [-1 + 4\left(\frac{t}{t_f}\right) - 2\left(\frac{t}{t_f}\right)^2] D & \text{ailleurs} \end{cases} \quad [\text{II.78}]$$

La figure II.8 : montre l'allure des positions, vitesses et accélérations.

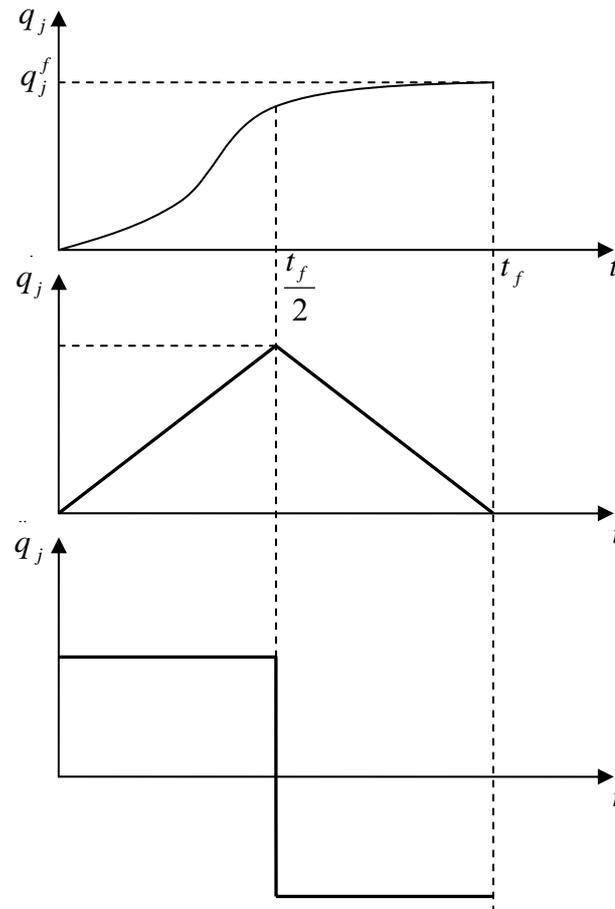


Figure II-8 : Interpolation Bang-Bang.

II.7.3. Loi de commande du couple calculé (computed torque control)

Cette loi de commande ressemble à la classe des lois de commande par découplage non linéaire. Elle consiste à poursuivre une trajectoire prédéfinie par $q_d(t)$, $\dot{q}_d(t)$ et $\ddot{q}_d(t)$ et permettant d'obtenir en boucle fermée un comportement linéaire et découplé sous certaines conditions à définir. Ce type de loi de commande sera proche de la loi de commande par découplage non linéaire mais avec un objectif différent.

Partant du même modèle dynamique donné par l'équation (III.10), la loi de commande est donnée par la relation suivante :

$$\Gamma = M(q)[\ddot{q}_d + k_p \tilde{q} + k_v \dot{\tilde{q}}] + C(q, \dot{q})\dot{q} + G(q) \quad [\text{II.79}]$$

Où k_p et k_v sont des matrices symétriques et définies positives; et $\tilde{q} = q_d - q$ désignant l'erreur de poursuite en position. Ainsi la dynamique en boucle fermée du robot est obtenue en remplaçant le couple du modèle dynamique [II.45] par la loi de commande [II.79] :

$$\ddot{q}_d + k_p \tilde{q} + k_v \dot{\tilde{q}} = 0 \quad [\text{II.80}]$$

Le choix d'une matrice diagonale pour k_p et k_v nous assure un découplage et ainsi le système est linéaire découplé par feedback. Le schéma bloc de la loi de commande du couple calculé est donné par la figure II-9.

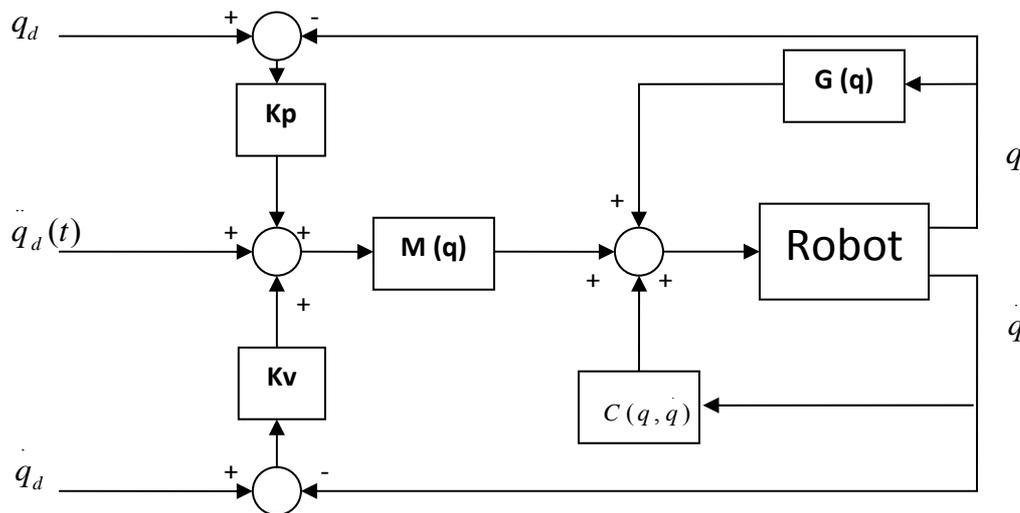


Figure II-9 : Schéma bloc de la commande du couple calculé

II.8. Exemples de simulation

Les simulations ont été réalisées par MATLAB qui est considéré comme l'un des outils les plus puissants dans le calcul mathématique. Plusieurs simulations ont été effectuées avec le modèle dynamique du robot décrit par l'équation [II.45] avec deux types de lois de commande, la loi de commande point à point et la loi de commande du couple calculé. Nous avons deux objectifs différents : l'objectif de régulation pour la première et de poursuite de trajectoire pour la deuxième. Nous utilisons le modèle du robot manipulateur à deux degrés de liberté.

II.8.1. Bras manipulateur à 2ddl

Le premier système contrôlé dans cette étude est un bras manipulateur plan à deux degrés de liberté (voir figure II-10), dont le modèle de l'équation [II.45] est donné par les trois matrices $M(q)$, $C(q, \dot{q})$ et $G(q)$.

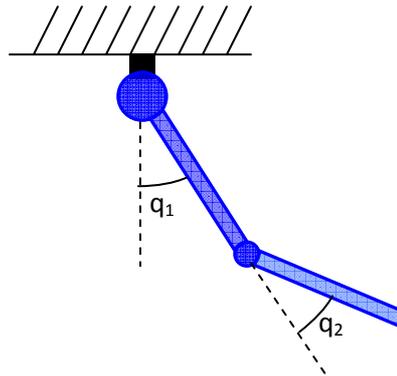


Figure II-10 : Robot manipulateur à deux degrés de liberté

II.8.1.1. Application de la Loi de commande point à point

Les figures (II.11, II-12, II-13, II-14, II-15, II-16) présentent les résultats de simulation, pour des positions désirées $q_1=1.5$ rad et $q_2=1.6$ rad, les matrices des paramètres k_p et k_v sont données par :

$$k_p = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix}, k_v = \begin{bmatrix} 120 & 0 \\ 0 & 100 \end{bmatrix}.$$

Nous remarquons d'après les figures II-11 et II-12 que les deux articulations 1 et 2 atteignent les valeurs désirées 1.5 rad et 1.6 rad respectivement. Nous remarquons aussi que le temps de réponse des deux articulations diffère, puisque la première articulation atteint la valeur désirée plus rapidement que la deuxième.

Les erreurs sur les positions 1 et 2 ont une convergence exponentielle, tel qu'il est montré dans les figures II-15 et II-16.

II.8.1.2. Application de la loi de commande du couple calculé

Le comportement de notre système après l'application de cette loi de commande, est illustré par les figures (II-17, II-18, II-19, II-20) pour les matrices des gains k_p et k_v suivantes :

$$k_p = \begin{bmatrix} 650 & 0 \\ 0 & 600 \end{bmatrix}, k_v = \begin{bmatrix} 85 & 0 \\ 0 & 80 \end{bmatrix}.$$

La génération de trajectoire a été réalisée par une interpolation polynomiale de degré cinq.

Pour ce type de commande en poursuite de trajectoire, les résultats de simulation montrent qu'il y a une bonne poursuite des trajectoires, avec un temps de réponse identique pour les deux articulations.

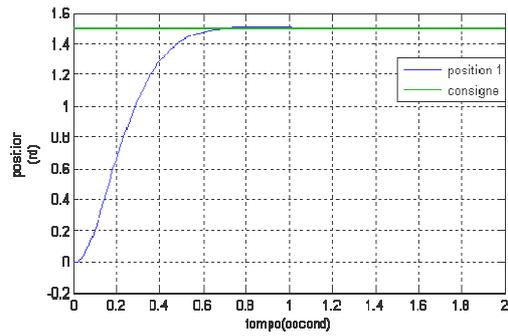


Figure II-11: Allure de la position 1 avec la commande point à point.

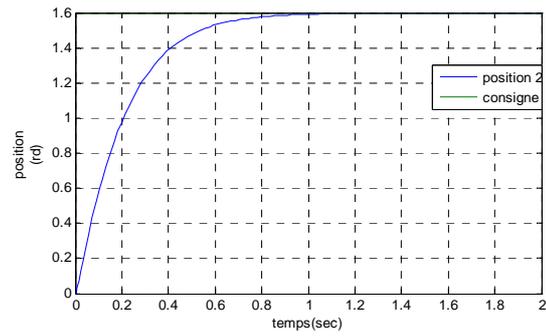


Figure II-12 : Allure de la position 2 avec la commande point à point.

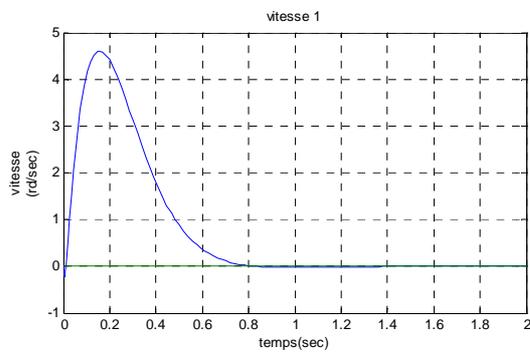


Figure II-13: Allure de la vitesse 1 avec la commande point à point.

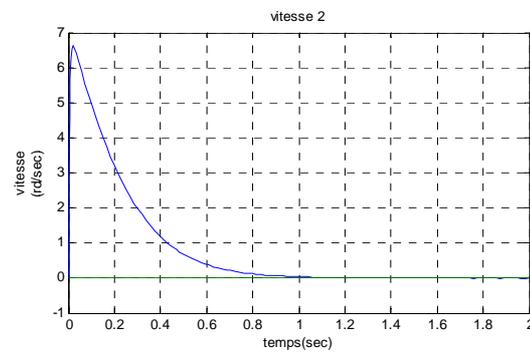


Figure II-14 : Allure de la vitesse 2 avec la commande point à point.

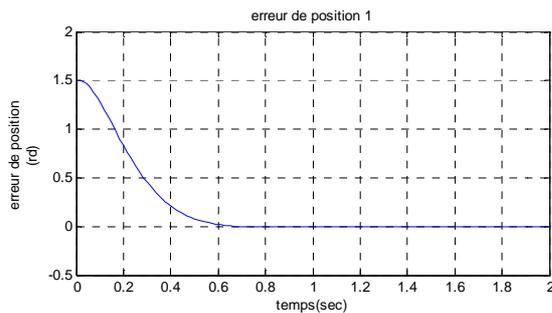


Figure II-15 : Allure de l'erreur en position 1 avec la commande point à point.

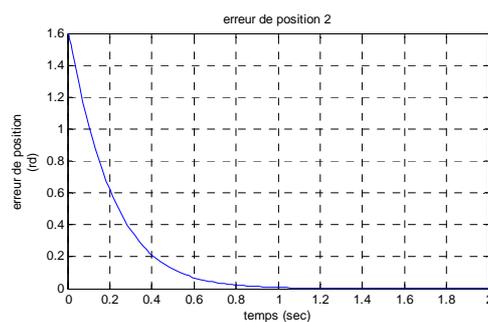


Figure II-16 : Allure de l'erreur en position 2 avec la commande point à point.

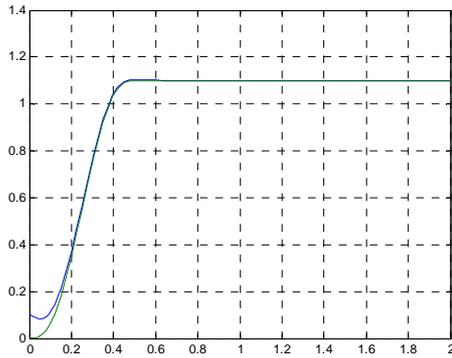


Figure II-17 : Allure de la position 1 avec la commande du couple calculé.

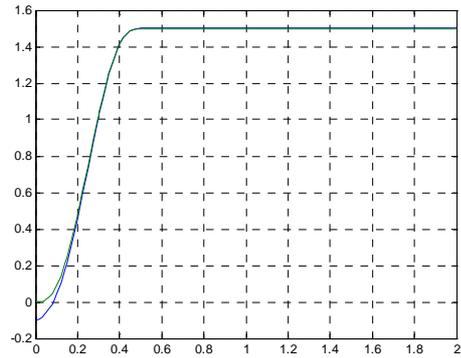


Figure II-18: Allure de la position 2 avec la commande du couple calculé.

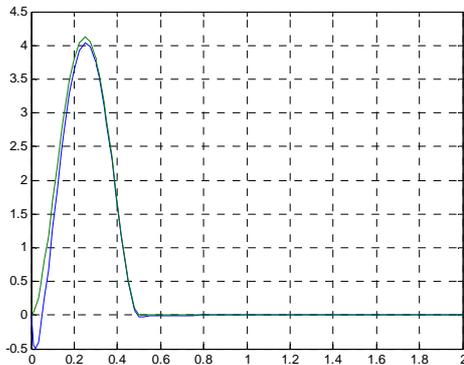


Figure II-19 : Allure de la vitesse 1 avec la commande du couple calculé.

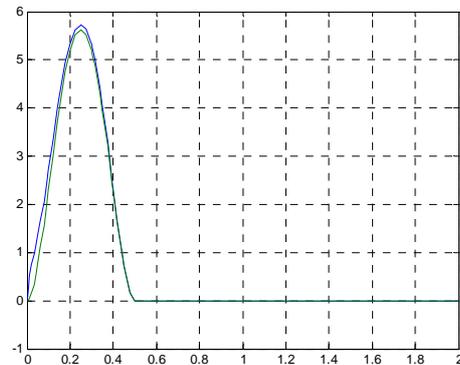


Figure II-20: Allure de la vitesse 2 avec la commande du couple calculé.

II.9.Conclusion

Dans cette partie de notre travail nous avons mis en évidence la modélisation des bras manipulateurs à chaîne ouverte simple et l'application de quelques lois de commande classiques à ce type de système, à savoir la commande proportionnel dérivée avec compensation de l'effet de gravité (point à point) avec un objectif de régulation, et la loi de commande Loi de commande du couple calculé (computed torque control) ayant comme objectif la poursuite de trajectoire.

L'objectif principal de notre travail est d'arriver à concevoir un contrôleur flou pour la commande d'un bras manipulateur à deux degré de liberté, pour but de comparer ces résultats

à celles de la commande classique conventionnelles, et par faute de temps malheureusement on n'a pas pu trouver de bons résultats, pour cela nous avons essayé d'appliquer cette technique pour la commande d'un autre type de système non-linéaire à savoir le pendule inversé.

III.1. Introduction

Dans un système basé sur la logique floue les variables d'entrées et les variables de sorties sont reliées par des règles qui interviennent au niveau de l'inférence de la défuzzification. Dans ce chapitre on va introduire un contrôleur flou pour la commande d'un pendule inversé avec des essais de simulation sur le logiciel MATLAB

III.2. Présentation du pendule inversé

Le pendule inversé est un système non linéaire et instable utilisé pour tester des nouvelles techniques de contrôle.

Une représentation du pendule inversé est donnée par la figure III-1; un chariot surmonté d'une tige se déplace horizontalement sur une courroie actionnée par un moteur à courant continu.

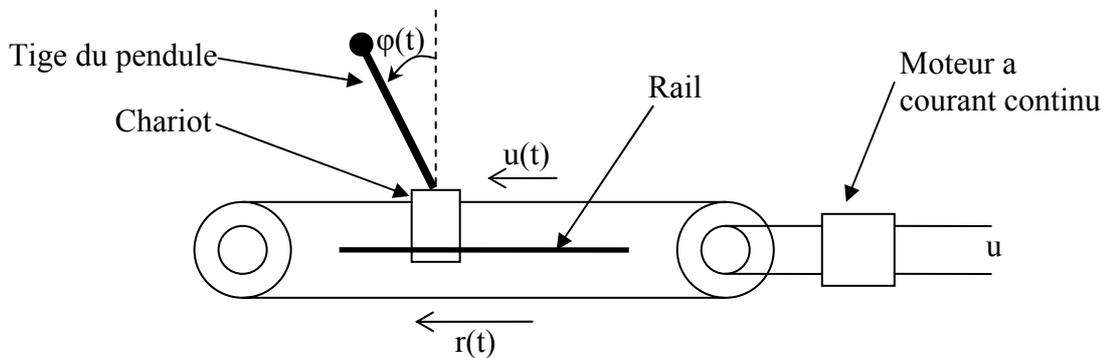


Figure III-1 : Description schématique du pendule inversé

Les variables principales sont :

- la position angulaire de la tige φ .
- La vitesse angulaire de la tige $\dot{\varphi}$.
- La position horizontale du chariot r .
- La vitesse de déplacement du chariot \dot{r} .

Soit le vecteur d'état :

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} r \\ \dot{r} \\ \varphi \\ \dot{\varphi} \end{bmatrix} \quad (\text{III.1})$$

Les équations différentielles $\dot{x} = f(x, u)$ modélisant le pendule inversé sont de la forme :

$$\begin{cases} \dot{x}_1 = x_3 \\ \dot{x}_2 = x_4 \\ \dot{x}_3 = \frac{a_{32} \sin(x_2) \cos(x_2) + a_{33}x_3 + a_{34}x_4 \cos(x_2) + a_{35}x_4^2 \sin(x_2) + b_3 u}{1 - a_{45}(\sin(x_2))^2} \\ \dot{x}_4 = \frac{a_{42} \sin(x_2) + a_{43}x_3 \cos(x_2) + a_{44}x_4 + a_{45}x_4^2 \sin(x_2) \cos(x_2) + b_4 \cos(x_2).u}{1 - a_{45}(\sin(x_2))^2} \end{cases} \quad (III.2)$$

Les coefficients a_{ij} et b_i pour $i = 3,4$ et $j = 2, \dots, 5$ sont donnés dans l'annexe B.

III.3. La commande par logique floue du pendule inversé

Dans la commande floue, plusieurs approches peuvent être utilisées. Ces approches se distinguent selon les entrées, qui sont généralement l'erreur e (différence entre la consigne et la sortie du processus) et la variation de l'erreur Δe ; ainsi que la sortie du contrôleur.

Selon que la sortie du contrôleur flou concerne la commande ou sa variation, nous obtiendrons des équivalents structuraux non linéaires des contrôleurs classiques PD, PI, ou même PID.

La figure III-2 montre un contrôleur flou de type PI, dans ce cas la sortie du contrôleur flou est considérée comme un incrément de commande.

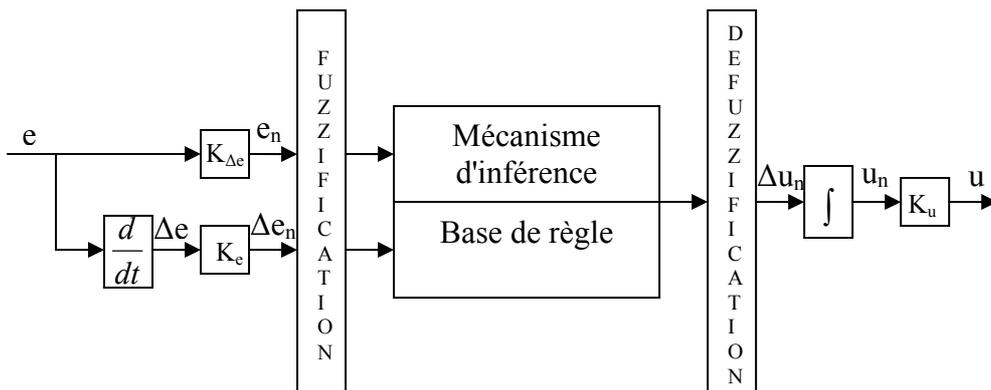


Figure III-2 : Schéma de principe d'un contrôleur flou de type PI

Par contre, si la sortie du contrôleur est directement appliquée au processus, le contrôleur est appelé contrôleur flou de type PD. Dans notre travail, nous avons utilisé ce type de

contrôleur flou dans la commande du pendule inversé. La structure de ce type de régulateur est représentée dans la figure III-3.

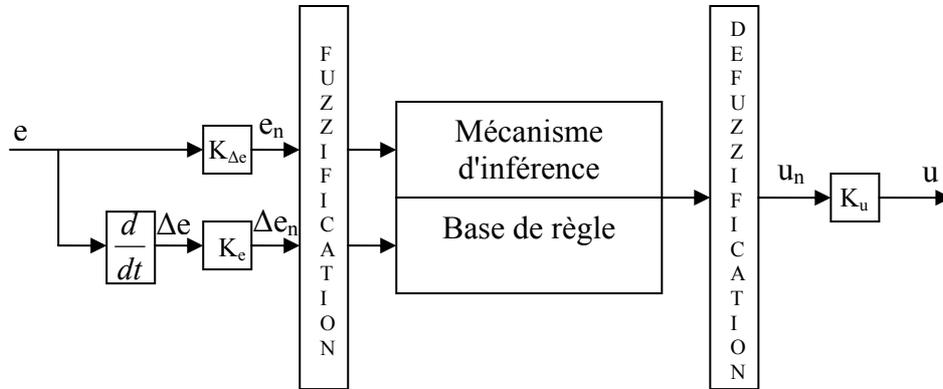


Figure III-3 : Schéma de principe d'un contrôleur flou de type PD

Le contrôleur flou de type PID peut être obtenu en combinant les deux contrôleurs flous de type PI et PD comme il est indiqué dans la figure III-4.

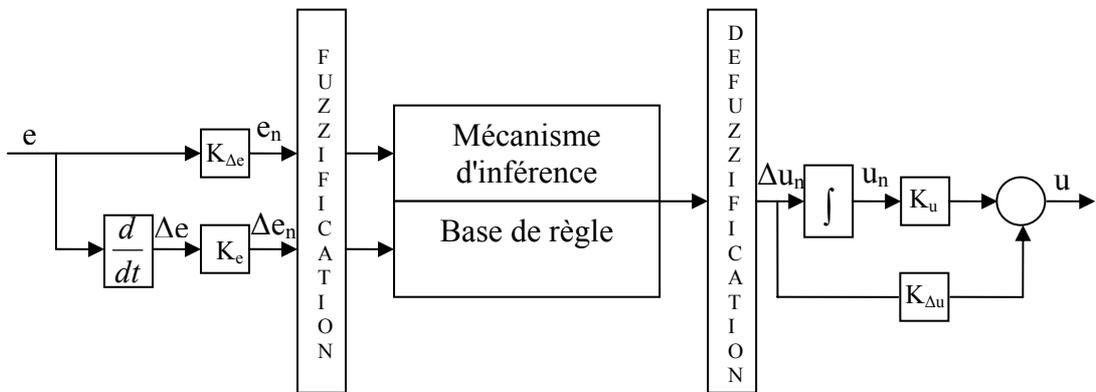


Figure III-4: Schéma de principe d'un contrôleur flou de type PID

Les gains K_e et $K_{\Delta e}$ sont nommés facteurs d'échelle. Ils servent à transformer les valeurs physiques des entrées dans un domaine normalisé $[-1 \ 1]$. De plus, la dénormalisation change la valeur normalisée du signal de commande à son domaine physique respecté à l'aide des deux facteurs d'échelle K_u et $K_{\Delta u}$. Par conséquent, les entrées du contrôleur flou e_n et Δe_n sont normalisées par l'utilisation des expressions suivantes:

$$\begin{cases} e_n = K_e \cdot e \\ \Delta e_n = K_{\Delta e} \cdot \Delta e \end{cases} \quad (III.3)$$

De la même façon, la sortie u_n du contrôleur est dénormalisée à u en utilisant la relation suivante:

$$u = K_u \cdot u_n + K_{\Delta u} \cdot \Delta u_n \quad (\text{III.4})$$

III.4. Conception du Contrôleur flou pour la commande du pendule inversé

L'objectif de la stratégie de régulation est d'une part, de maintenir la tige du pendule en position verticale haute, d'autre part, d'amener le chariot vers une position donnée rc .

La conception d'un contrôleur flou passe toujours par quatre stades que nous allons détailler successivement.

1^{ère} étape: choix des entrées et sorties.

Les variables d'entrée du contrôleur flou sont la position et une grandeur proportionnelle à la vitesse angulaire de la tige, φ et $\Delta\varphi$, l'erreur en position e donnée par $e = r - rc$ et une grandeur proportionnelle à la vitesse de déplacement du chariot $\Delta e = \Delta r$. La variable de sortie du contrôleur flou est une tension u permettant de commander un moteur à courant continu.

Pour diminuer le nombre de variables en entrée du contrôleur flou, nous réduisons la dimension du problème. Cette réduction peut être obtenue par une fusion mathématique des variables d'entrée avant de les fournir en entrée du contrôleur flou. Elle est réalisée comme suit:

$$\begin{cases} X\varphi = a \cdot \varphi + b \cdot \Delta\varphi \\ Xe = c \cdot e + d \cdot \Delta e \end{cases} \quad (a, b, c, d > 0) \quad (\text{III.5})$$

Ainsi, si Xe est nul, cela signifie que le chariot a atteint sa position de référence ($e = 0$ et $\Delta e = 0$), ou encore qu'il se déplace vers celle-ci ($ce = -d\Delta e$). Le raisonnement est identique pour $X\varphi$. Si $X\varphi$ est nul, la position angulaire du pendule est stabilisée ou s'approche de zéro. Par conséquent, la stabilisation de $X\varphi$ et Xe permet d'amener le pendule vers une position de référence et assure le maintien de la tige du pendule en position verticale haute. Plus la valeur absolue de $X\varphi$, respectivement Xe , est importante, plus la position horizontale, respectivement, angulaire, du pendule est critique. Ces deux variables, $X\varphi$ et Xe , représentent respectivement la position angulaire critique et la position horizontale critique du pendule.

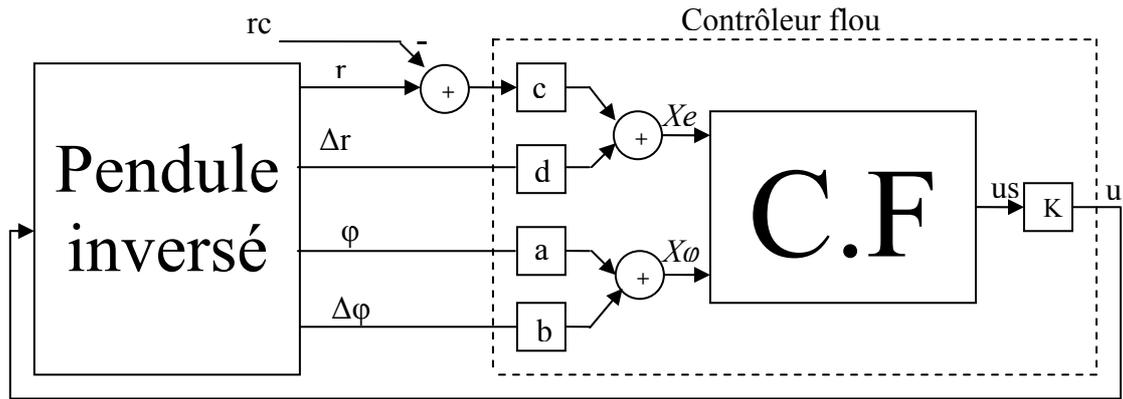


Figure III-5 : Contrôleur flou pour le pendule inversé

2^{ème} étape: définition des fonctions d'appartenance

La première étape de conception a permis de cerner au mieux les caractéristiques linguistiques des variables. Il faut maintenant définir complètement les sous-ensembles flous, c'est à dire expliciter leurs fonctions d'appartenance. Une fois encore, l'intuition et l'expérience auront leur rôle à jouer. Quelques principes ressortent de la pratique ou des essais: choix de fonctions triangulaires ou trapézoïdales, chevauchement de l'espace des sous-ensembles voisins, degré maximal d'appartenance.

Nous envisagerons pour toutes les variables d'entrée et de sortie trois sous ensembles flous dont les fonctions d'appartenance sont de formes triangulaire. La figure III-6 montre le choix de la forme des fonctions d'appartenance pour les deux entrées et la sortie, répartis sur l'univers de discours normalisé [-1,1].

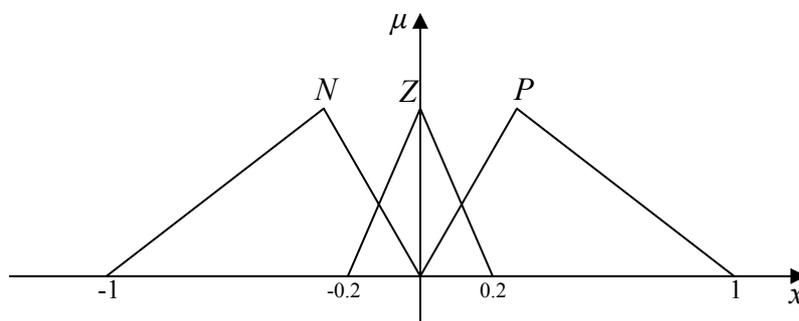


Figure III-6 : Partition floue

Où les abréviations dans la figure signifient:

- N : Négatif;
- Z : Zéro;
- P : positif.

3^{ème} étape: détermination du jeu de règles :

Cette étape concerne l'élaboration de la base de règle du contrôleur. C'est de nouveau à un expert et à sa connaissance du problème que l'on se fiera le plus souvent. Dans le cadre de la régulation, nous utilisons fréquemment l'erreur et sa variation. À partir de ces deux entrées, traduites sous forme de variables floues, il est possible de déterminer les règles dans le domaine temporel et on peut construire une matrice comprenant toutes les possibilités linguistiques de ces règles.

L'analyse temporelle, qui doit conduire à établir les règles du contrôleur flou, peut par exemple consister à considérer la réponse à un échelon d'un processus à piloter en fonction des objectifs qui seront fixés en boucle fermé.

La forme générale de la réponse est représentée par la figure III-7, selon l'amplitude de e et le signe de Δe , cette réponse est divisée en quatre régions.

- ①: $e > 0$ et $\Delta e < 0$
- ②: $e < 0$ et $\Delta e < 0$
- ③: $e < 0$ et $\Delta e > 0$
- ④: $e > 0$ et $\Delta e > 0$

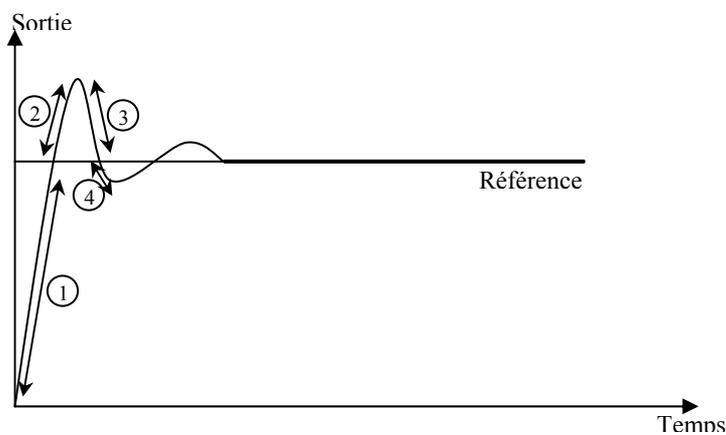


Figure III-7 : Ecriture du jeu de règles grâce à une analyse temporelle

Le problème de commande du pendule est loin d'être trivial, car deux variables sont à réguler mais l'on ne dispose que d'une seule action de commande. La dynamique de φ étant plus rapide que celle de r , la stratégie adoptée est la suivante : nous chercherons dans un premier temps à équilibrer le pendule (position verticale haute), puis à l'amener progressivement vers sa position de référence en le déséquilibrant du bon côté, c'est à dire vers celui qui entraînera une valeur de commande telle que le pendule se déplacera vers r_c .

Nous pouvons alors écrire les règles suivantes afin de commander le pendule à la fois horizontalement et verticalement.

$X_e \backslash X_\phi$	N	Z	P
N	N	N	P
Z	N	Z	P
P	N	P	P

Tableau III-1: Base de règle du contrôleur flou

Le tableau III-1 peut être traduit de la manière suivante:

1. Si X_ϕ est Négatif et X_e est Négatif Alors u est Négatif.
- .
- .
6. Si X_ϕ est Zéro et X_e est positif Alors u est positif.
- .
- .
9. Si X_ϕ est Positif et X_e est positif Alors u est positif.

Ces règles traduisent bien que l'objectif prioritaire est la stabilisation verticale du pendule, X_e n'est considéré que lorsque X_ϕ est nul.

4^{ème} étape: choix d'une méthode de défuzzification, du mécanisme d'inférence, ainsi des valeurs des facteurs d'échelle.

La méthode de défuzzification utilisée dans notre travail est la méthode du centre de gravité.

Pour le mécanisme d'inférence, nous avons opté pour le mécanisme d'inférence MAX-MIN.

Le choix des facteurs d'échelle résulte après différents essais de simulation afin d'avoir les meilleurs résultats.

III.4.1 Résultats de simulation

- Schéma bloc de simulation de pendule sur MATLAB est montré par la figure III-8.

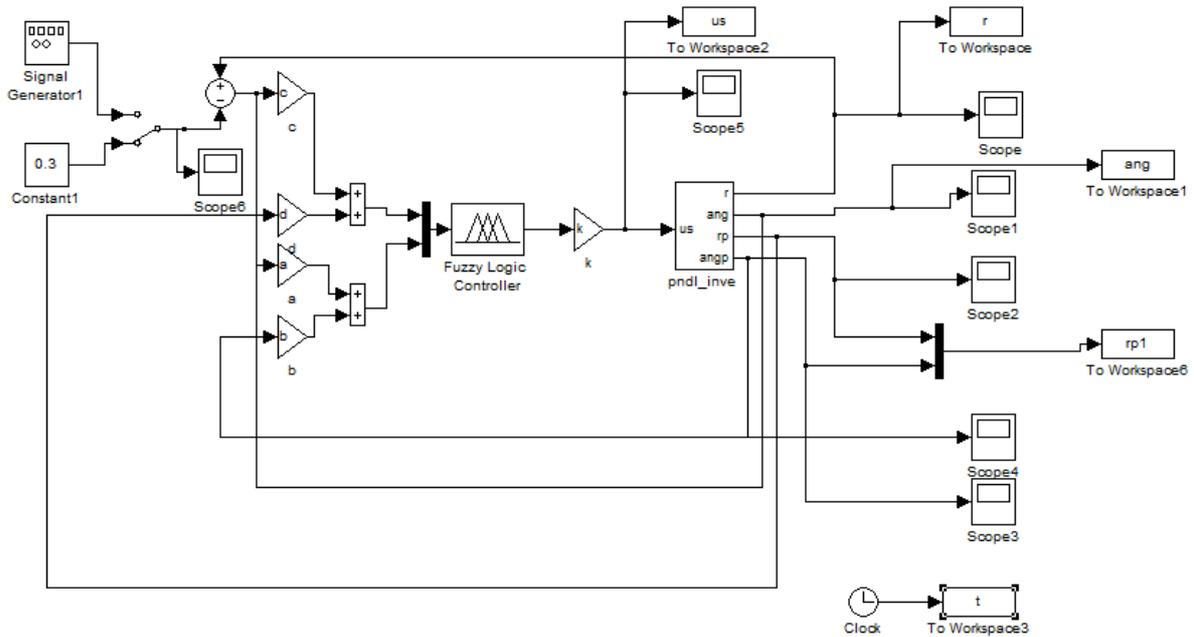


Figure.III-8 : Schéma bloc de simulation sur MATLAB.

Les limitations physiques du pendule inversé ici considéré sont $r \in [-0.5, +0.5]$ mètres et $u \in [-10, +10]$ volts.

Pour évaluer les performances du réglage par logique floue, nous avons effectué différentes simulations pour une consigne $rc = 0.3m$, ainsi que pour une inversion de consigne jusqu'à $rc = -0.3m$, tel qu'il est montré dans les figures (III-9, III-10, III-11, III-12, III-13, III-14, III-15).

Les performances obtenues avec ce contrôleur flou contenant seulement neuf règles sont tout à fait satisfaisantes. De plus les règles sont très lisibles et facilement interprétables par un non spécialiste en automatique.

D'après la figure III-9, nous pouvons remarquer que la réponse de notre système rejoint sa référence avec un temps de réponse de 4s. Nous avons remarqué aussi qu'il existe un retard qui est dû au fait que notre système doit tout d'abord positionner la tige du pendule à sa position verticale. La tige du pendule se déstabilise puis elle rejoint sa position verticale tel qu'il est montré dans la figure III-10. La tension de commande atteint 8 volts pour permettre au chariot d'atteindre sa position désirée, en gardant la tige dans la position verticale.

Pour une variation de consigne de 0.3m à -0.3m la réponse suit toujours la référence, tel qu'il est montré par la figure III-13. Nous remarquons aussi qu'il y a un léger dépassement. Ceci est dû au changement brusque de consigne de 0.6m. À chaque fois qu'il y a changement

de consigne, nous avons une variation de la position angulaire d'environ 0.01rd, mais la tige arrive à se stabilisé dans sa position verticale. La tension de commande reste entre 8volts et 9 volts.

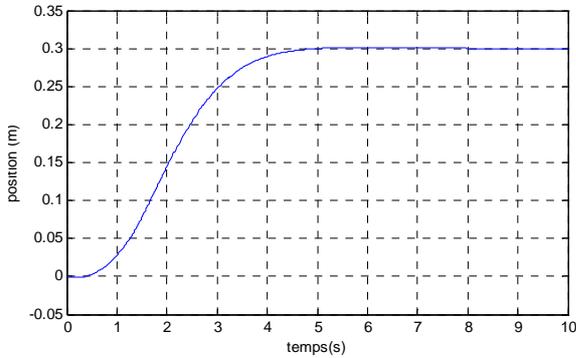


Figure III-9 : Réponse de la position à une consigne de 0.3m

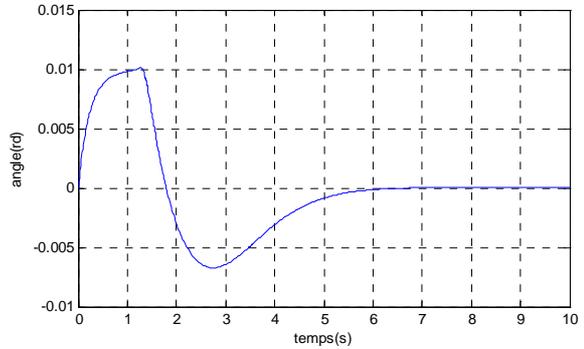


Figure III-10 : La position angulaire du pendule

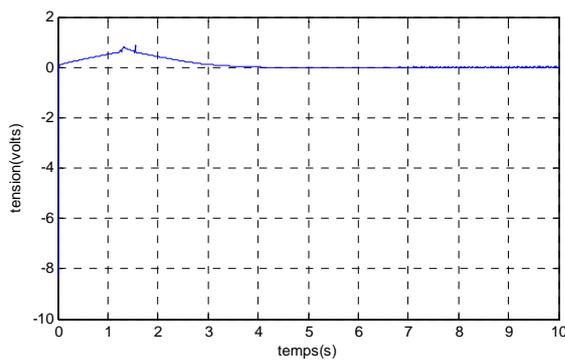


Figure III-11 : La tension de commande

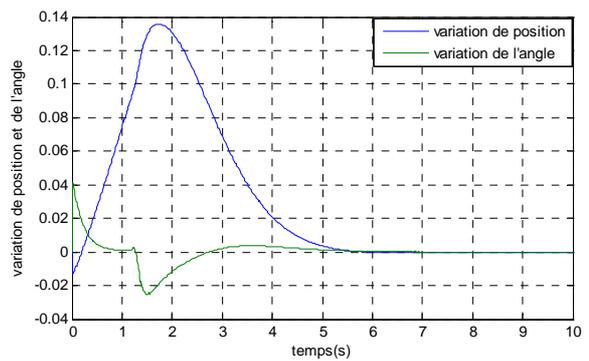


Figure III-12 : Les variations de la position et de l'angle

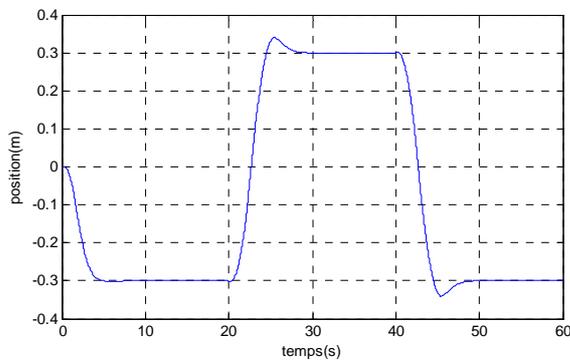


Figure III-13 : Réponse de la position pour une variation de consigne de 0.3m jusqu'à -0.3m

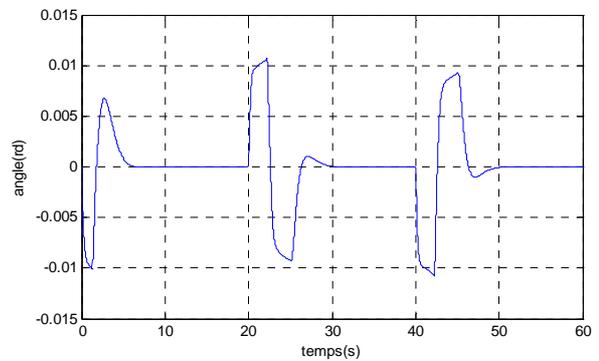


Figure III-14 : Réponse de la position angulaire pour une variation de consigne de 0.3m jusqu'à -0.3m

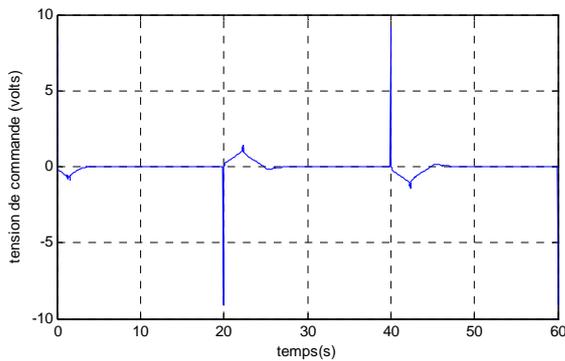


Figure IV-15 : Tension de commande pour une variation de consigne de 0.3m jusqu'à -0.3m

III.5. Conclusion

L'application de la commande floue sur des systèmes non-linéaires, tel que l'exemple du pendule à bien montrée la place de cette technique pour ce type du système qui sont difficilement modélisable et surtout à commander par les techniques de commande conventionnelles classique. Les résultats de simulation illustre encore une fois la montée exponentielle de cette méthode et la place qui occupe aujourd'hui.

VI.1.Introduction à la réalité virtuelle

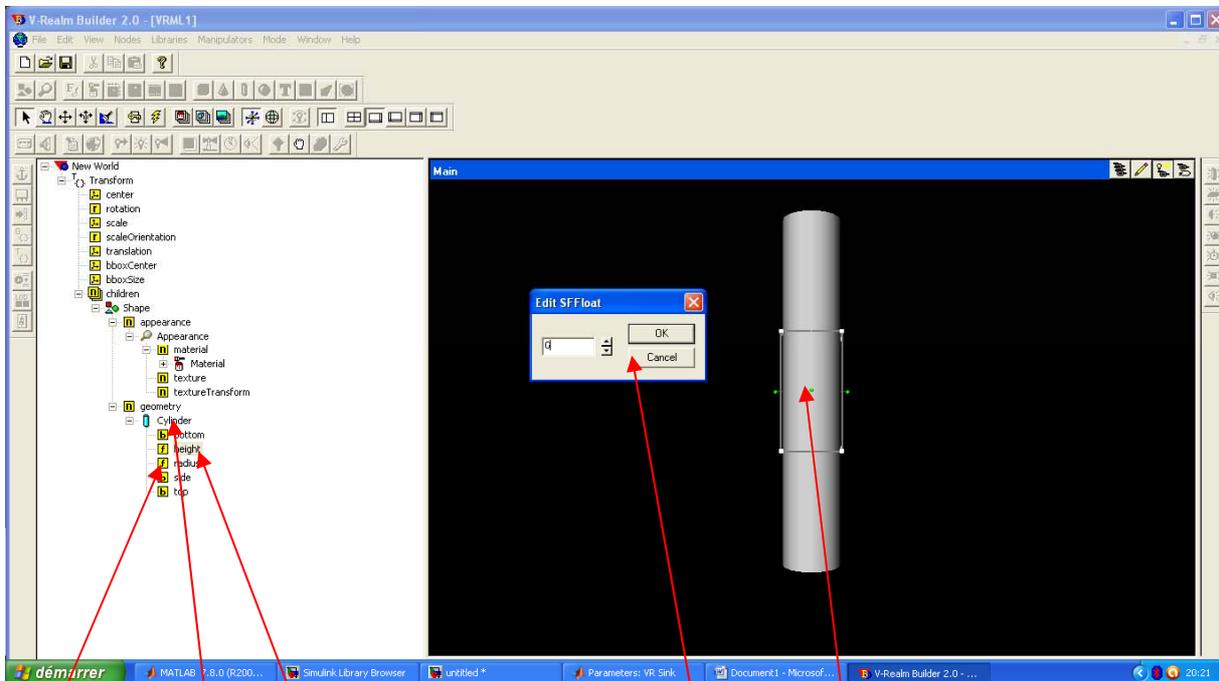
La réalité virtuelle permet de visualiser le comportement des systèmes physiques, par une simple simulation. Dans ce contexte, nous allons visualiser de façon concrète le comportement du bras manipulateur à 2ddl suite à une commande quelconque.

La conception du bras manipulateur est réalisée à l'aide du "Virtual Reality Toolbox" du SIMULINK-MATLAB.

VI.2. Démonstration sur un bras manipulateur à 2ddl

Les étapes à suivre pour la conception du bras manipulateur sont dans les figures (VI-1 et VI-2).

La figure VI-1, représente un cylindre qui formera les tiges du bras.



Choix de diamètre

Choix de la longueur

Choix des paramètres de dimension

La forme finale de cylindre qui représentera la première tige du bras

Sélection de cylindre

Figure.VI-1 : Construction des deux tiges du bras manipulateur

La sphère dans notre cas représente les articulations du bras manipulateur comme le montre la **Figure.VI-2**

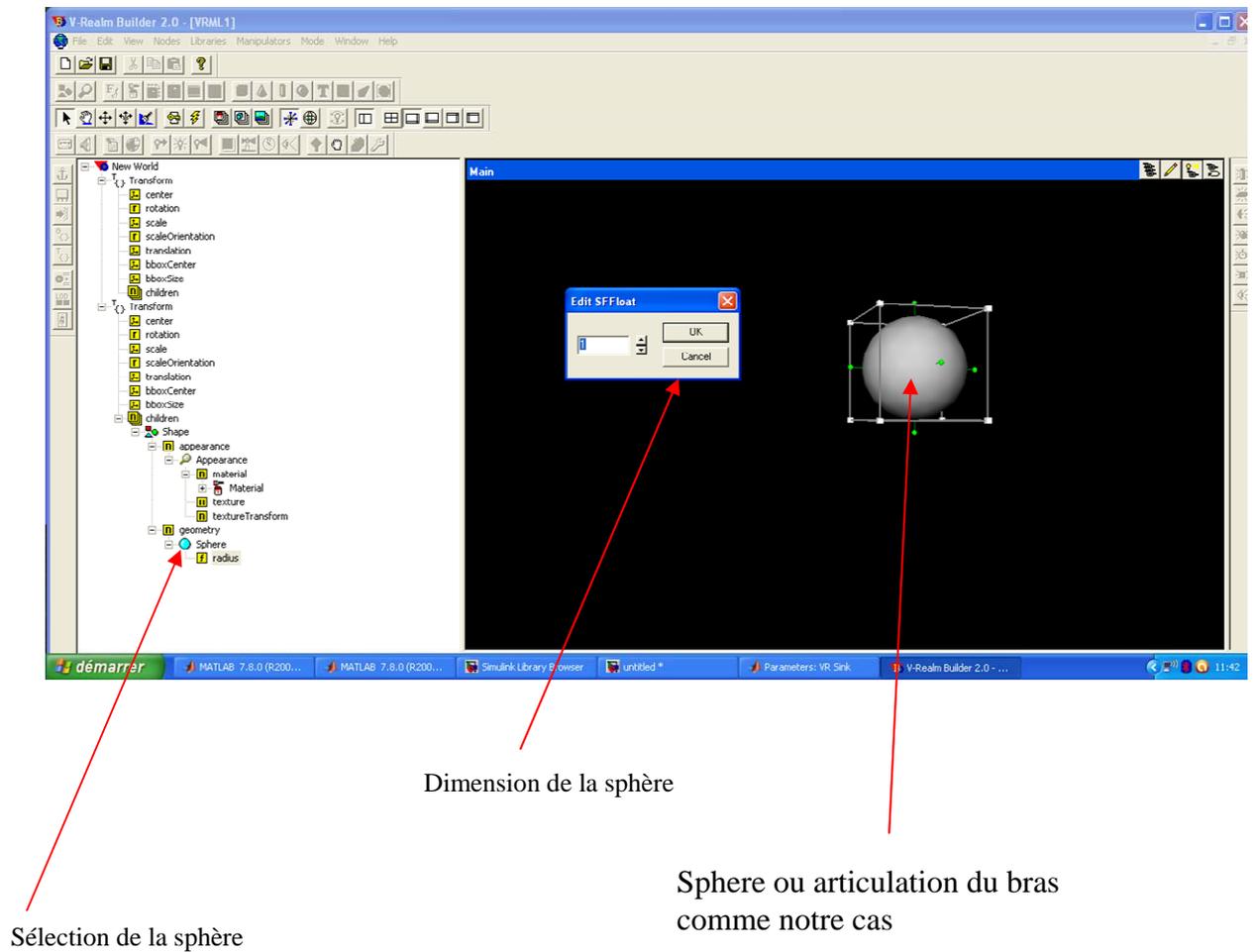


Figure.VI-2 : construction des articulations du bras manipulateur

La fenêtre "V-Realm Builder" permet de faire l'assemblage des parties du bras manipulateur, ces parties sont disponibles dans la barre d'outil, ceci est montré par les figure IV-3 :

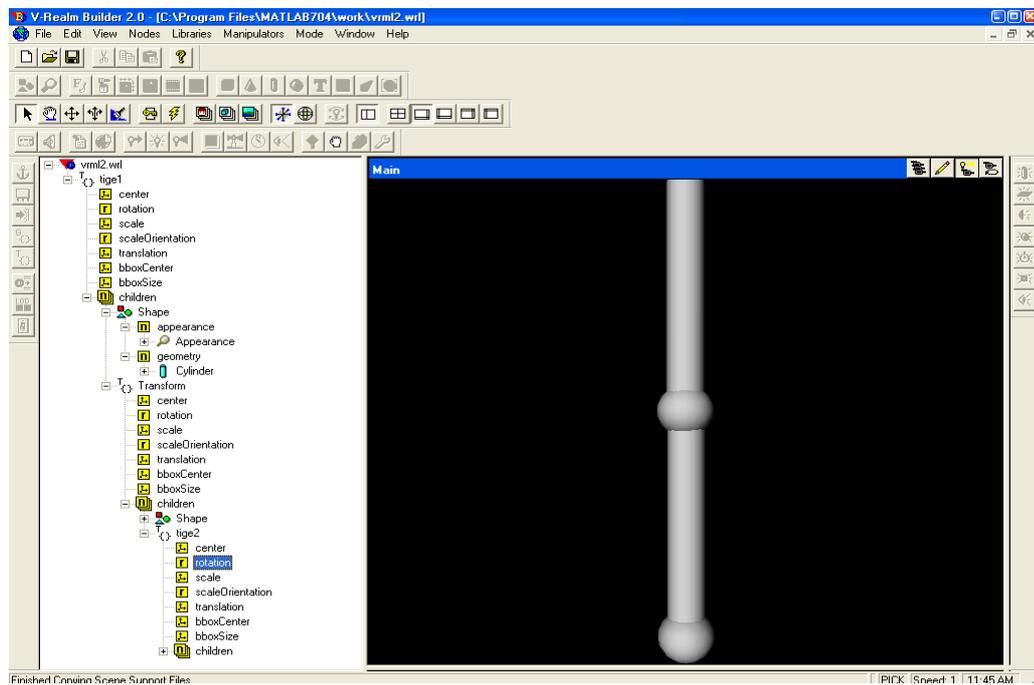


Figure.VI-3 : Assemblage du bras manipulateur

Les blocs SIMULINK utilisés pour cette application sont illustrés par la figure VI-2 :

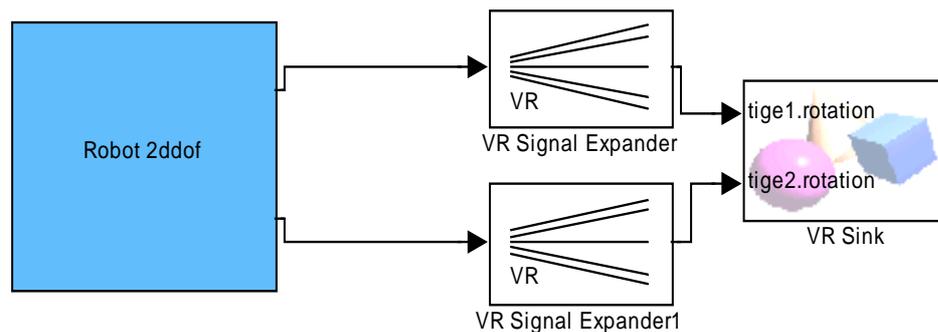


Figure.VI-4 : Blocs SIMULINK de la réalité virtuelle

Par double clic sur le bloc **VR Sink**, la fenêtre des paramètres apparaisse. Cette fenêtre permet de spécifier les paramètres de l'application, ainsi que d'accéder à une application déjà enregistrée:

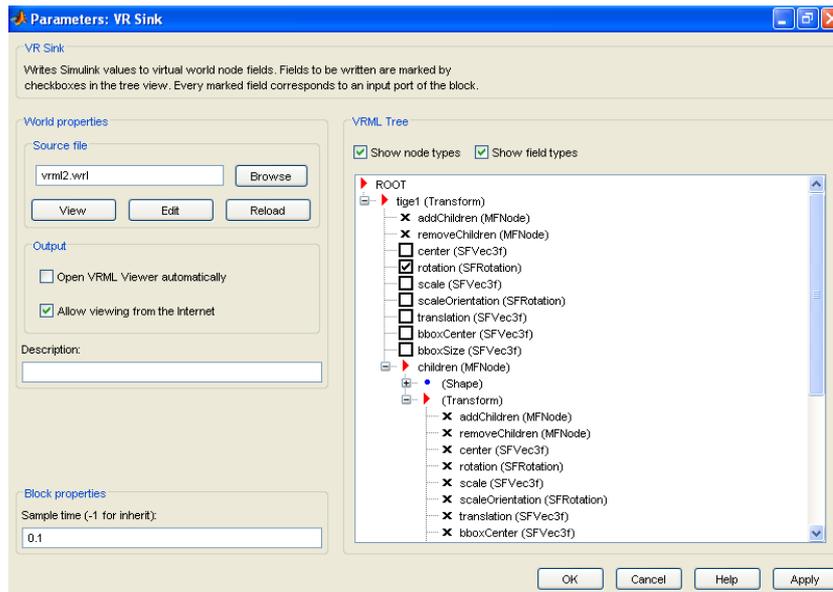


Figure.VI-5 : La fenêtre VR Sink

En lançant la simulation sur le schéma fonctionnel du Simulink, nous pouvons visualiser le comportement du bras manipulateur, et le suivi des références à l'aide d'une fenêtre qui apparaîtra juste après le lancement de la simulation :

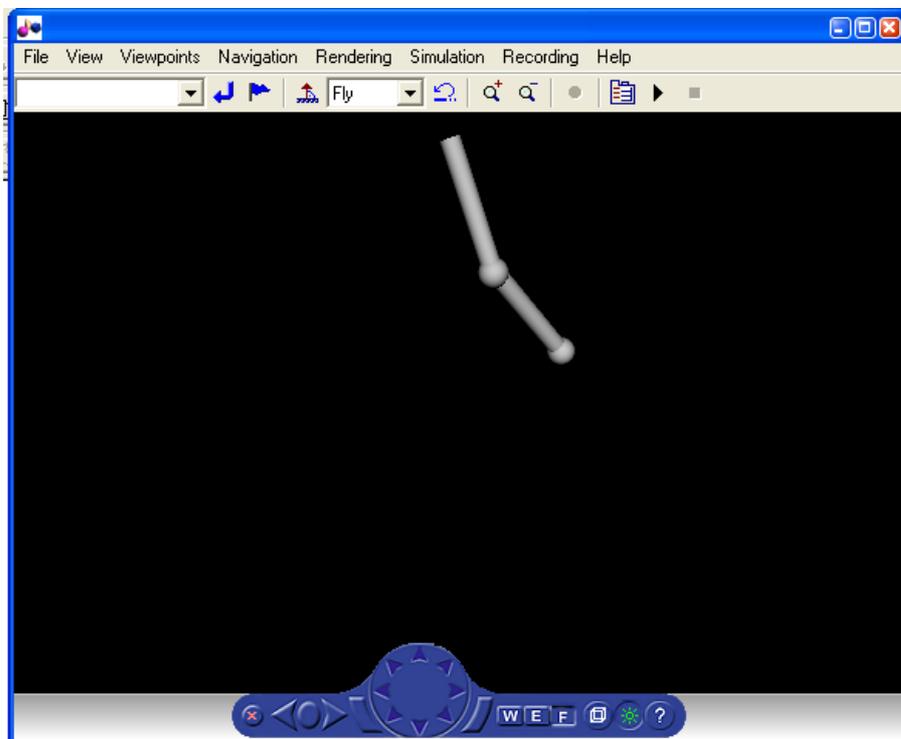


Figure-6 : Visualisation du comportement du bras manipulateur

VI.3.Conclusion

La réalité virtuelle nous permet de visualiser un peu de réel sur le comportement du système à commander, pour cela cette technologie fournit des raccourcis à la conception finale du système.

Conclusion générale

Notre objectif a eu pour but de la conception d'un contrôleur flou pour la commande d'un pendule inversé, en abordant parallèlement la commande classique sur un robot manipulateur à deux degré de liberté.

Le premier chapitre à été consacré pour la représentation des principes fondamentaux de la théorie de la logique floue, en citant une petite comparaison entre la logique classique et la logique floue, ainsi que les différents types de régulateurs flous pour la commande.

Dans le deuxième chapitre, un recueil assez complet sur les structures géométriques des robots manipulateurs à chaîne ouverte simple, ainsi que la présentation des différentes méthodes à utiliser pour la modélisation de ce type de système non-linéaire, et une application de quelques lois de commande classique à un bras manipulateur à deux degré de liberté et ce en procédant avec le SIMULINK du logiciel MATLAB, qui nous a permis de visualiser les résultats de ces commandes classiques.

Dans le troisième chapitre nous avons fait la conception d'un contrôleur flou dont le but est de commander un système non-linéaire à savoir le pendule inversé, des essais effectués avec deux types de consignes, des résultats satisfaisants confirment la rapidité et la robustesse de ce type de contrôleur par rapport au contrôleur classique qui s'adapte difficilement au changement des paramètres.

En fin, le quatrième chapitre est consacré à la réalité virtuelle en montrant une petite animation du bras manipulateur à deux degré de liberté.

Le contrôleur flou à montré son efficacité pour la commande, pour cela on envisage dans le future de pouvoir continuer ce travail, en réalisant d'autres types de ce contrôleur pour valider encore plus de bon résultats.

Bibliographie

- [1] M^r ETENNE DOMBRE, WISSAMA KHALIL ‘ Modélisation et commande des robots 1^{ère} édition HERMES SCIENCE PUBLICATION PARIS, 1988.
- [2] M^r WISSAMA KHALIL, M^r ETENNE DOMBRE, ‘ Modélisation identification et commande des robots 2^{ème} édition HERMES SCIENCE PUBLICATION PARIS, 1988.
- [3] M^r J.P LALLEMAND, S.ZEGHLOUL ‘ Robotique aspects fondamentaux modélisation mécanique que CAO robotique-commande ‘ Université de Poitiers, CNRS Laboratoire de mécanique des solides, 1994.
- [4] M^r Veronique LACROSE ‘ Réduction de la Complicité des Contrôleurs Flous : Application à la Commande Multi variable ’ Docteur en Automatique et l’Informatique Industrielle.
- [5] M^r MOHAMMED FAYCAL KHALIL- “Observateur non linéaire application à la commande des robots manipulateurs” Magister en Electronique, USTO ,1999.
- [6] M^r CHENAF MOHAMMED “Modélisation et commande adaptative des robots rigide et rapide ” Magister en Electronique, USTO, 1999.
- [7] M^{me} GELENA GODJEVAC “Idée nette sur la logique floue” presse POLYTECHNIQUE ET UNIVERSITAIRES ROMANDES ,1999.
- [8] M^r blaha“ contrôleur flou pour la navigation autonome d’un robot mobile et mise au point du SVIT” PFE en Electronique, USTO, 2004.
- [9] M^r HANSRUEDI BUHLER“ réglage par logique floue” PRESSE POLYTECHNIQUE ET UNIVERSITAIRES ROMANDES.1994.

Annexe A :**Notations des variables de modèle dynamique**

Les principales notations sont les suivantes :

- \mathbf{a}_j vecteur unitaire suivant l'axe \mathbf{z}_j ;
- \mathbf{F}_j résultante des forces extérieures sur le corps C_j ;
- \mathbf{f}_{ej} résultante du torseur dynamique exercé sur le corps C_j sur l'environnement ;
- \mathbf{F}_{sj} paramètre de frottement sec de l'articulation j ;
- \mathbf{F}_{vj} paramètre de frottement visqueux de l'articulation j ;
- \mathbf{g} accélération de la pesanteur ;
- \mathbf{G}_j centre de gravité du corps C_j ;
- \mathbf{I}_{Gj} tenseur d'inertie du corps C_j par rapport à un repère parallèle à \mathbf{R}_j et d'origine \mathbf{G}_j ;
- \mathbf{I}_{aj} moment d'inertie de l'actionneur j et de son réducteur ressenti par l'articulation ;
- \mathbf{J}_j matrice d'inertie globale du corps j de dimension (6x6) ;
- \mathbf{L}_j vecteur liant l'origine \mathbf{R}_j , antécédent de l'origine \mathbf{R}_j , et l'origine du repère \mathbf{R}_j ;
- \mathbf{M}_j masse de corps C_j ;
- \mathbf{MS}_j premier moment d'inertie du corps C_j auteur de l'origine du repère \mathbf{R}_j ;
- \mathbf{M}_{Gj} moment des efforts extérieurs exercé sur le corps C_j autour de \mathbf{G}_j ;
- \mathbf{M}_j moment des efforts extérieurs exercé sur le corps C_j autour de \mathbf{O}_j ;
- \mathbf{m}_j moment de torseur dynamique exercé sur le corps C_j par son antécédent ;
- \mathbf{m}_{ej} moment de torseur dynamique sur le corps C_j sur l'environnement autour de \mathbf{O}_j ;
- \mathbf{S}_j vecteur ayant pour origine \mathbf{O}_j et pour extrémité le centre de masse du corps C_j ;
- \mathbf{V}_j vitesse de point \mathbf{O}_j ;
- $\dot{\mathbf{V}}_j$ accélération du point \mathbf{O}_j ;
- \mathbf{V}_{Gj} vitesse du centre de gravité du corps C_j ;
- $\dot{\mathbf{V}}_{Gj}$ accélération du centre de gravité du corps C_j ;

ω_j vitesse de rotation du corps C_j ;

$\dot{\omega}_j$ accélération de rotation du corps C_j ;

V_j vecteur de dimension (6x1) regroupant les composantes des éléments du torseur cinématique du corps j ;

f_j résultante du torseur dynamique exercé sur le corps C_j par son antécédent et par l'actionneur j ;

jJ_j tenseur d'inertie du corps C_j par rapport au repère R_j , qui s'exprime par :

$${}^jJ_j = \begin{bmatrix} \int (y^2 + z^2) dm & - \int xy dm & - \int xz dm \\ - \int xy dm & \int (x^2 + z^2) dm & - \int yz dm \\ - \int xz dm & - \int yz dm & \int (x^2 + y^2) dm \end{bmatrix}$$

Annexe B

Coefficients du modèle du pendule inversé :

$a_{32} = -1.93$, $a_{33} = -1.99$, $a_{34} = 0.009$, $a_{35} = 0.063$,
 $a_{42} = 36.99$, $a_{43} = 6.26$, $a_{44} = -0.174$, $a_{45} = -0.1972$,
 $b_1 = 0.5128$, $b_2 = -1.61$.