



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA  
RECHERCHE SCIENTIFIQUE



Université Kasdi Merbah – Ouargla

Faculté des Nouvelles Technologies de l'Information et de la Communication

Département d'Electronique et des Télécommunications

**PROJET DE FIN D'ETUDES**

En vue d'obtention du

**Diplôme de Master en Automatique**

Filière : Génie électrique

Spécialité : Automatique

Présenté par : Abdennour BENNAMIA

Encadré par : M. Azzedine HAMZA

**Thème :**

***Commande à Mode Glissant Adaptative Floue  
Appliquée à un robot à deux degrés de libertés***

Présenté devant le jury composé de :

Nom et Prénom	Grade	Qualité	Université
BENHELLAL BELKHIR	MAA	Président	UKMO
SOURI SAMIRA	MAA	Examineur	UKMO

Année universitaire : 2017/2018

## *Remerciements*

*Je remercie Dieu, le tout puissant, pour m'avoir donné, le courage, la patience, la volonté et la force nécessaire, pour affronter toutes les difficultés et les obstacles, qui se sont hissés au travers de mon chemin, le long de cette année d'études.*

*J'adresse mes sincères remerciements à tous mes enseignants de l'université de Kasdi Merbah Ouargla (UKMO), notamment à mon promoteur Mr. Azzedine HAMZA d'avoir proposé le sujet sur lequel j'ai travaillé, et qui a assuré la direction et l'encadrement du travail présenté dans ce mémoire.*

*Je tiens notamment, à adresser l'expression de ma sincère reconnaissance aux membres du jury d'examen, pour l'intérêt qu'ils ont porté à ce travail.*

*Enfin, mes remerciements vont à tous ceux qui m'ont soutenu de près ou de loin à réussir ce travail.*

*Mes vives salutations*

## *Dédicaces*

*A qui puis-je dédier ce travail :*

*à mes très chers parents, dont le sacrifice, la patience, le soutien, l'aide et les encouragements sont le secret de ma réussite. Sans eux je ne serai pas ce que je suis aujourd'hui.*

*Je dédie ce travail également à mes chers frères et sœurs.*

*A mes amis : Mohammed Ghetas et Laaroussi Boushaba et Ahmed Tekha et à tous les collègues de la classe Master II Auto avec qui j'ai partagé des moments inoubliables.*

*A ceux avec qui je partage de bons souvenirs. A tous ce que j'aime et tous qui m'aiment. A tous ce qui m'ont aidé de près ou de loin dans ce travail.*

*Je dédie ce travail particulièrement à ma très chère épouse et à mes deux anges Sirine et Lejayne.*

*Merci à vous*

### ملخص :

يهدف البحث المقدم في هذه الرسالة إلى دراسة بعض طرق التحكم في ذراع آلية ذات مفصلين (بدرجتين من الحرية). نقدم هنا و بإيجاز طريقة التحكم عن طريق حساب العزم الدوراني وتطبيقها من أجل التحكم في زوايا دوران المفصلين، وبعد ذلك ولنفس الأهداف، نقدم طريقة التحكم بالوضع الانزلاقي. و للتغلب على النقائص المدمجة في النموذج الرياضي و كذلك الإعدادات غير المألوفة التي تعتمد عليها اعتمادا كليا هذه الانواع من طرق التحكم، نقترح دراسة التقنية الهجينة بين التحكم عبر السطح الانزلاقي و النموذج الضبابي المتكيف عبر السطح الانزلاقي حيث يتم استبدال النموذج الضبابي المتكيف للإعدادات غير المؤكدة و غير الخطية للروبوت و المكونة للجزء المعادل من قانون التحكم عبر وضع الانزلاق.

### كلمات البحث:

نمذجة روبوت آلي ذو مفصلين (بدرجتين من الحرية) ،التحكم عن طريق حساب العزم الدوراني، التحكم بالوضع الانزلاقي ذو النموذج الضبابي المتكيف؛

---

### **Résumé :**

Le travail présenté dans ce mémoire, a pour objectif d'étudier certaines méthodes de commande d'un bras manipulateur à deux degrés de libertés. On a étudié rapidement la commande par calcul de couple, et son application en vue de la commande des deux articulations du robot puis, et avec les mêmes objectifs, la commande par mode du glissement. Afin de surpasser les incertitudes de modélisations et les paramètres mal connus qui en dépend totalement ces deux commandes robustes, la technique hybride mode glissant-floue-adaptative est étudiée. Dans cette approche on substitue un modèle flou Takagi-Sugeno adaptatif via la surface du glissement pour modéliser la dynamique non linéaire incertaine fortement couplée du robot manipulateur présente dans la partie équivalente de la commande à mode glissant.

### **Mots Clés :**

Modélisation d'un robot manipulateur à deux degrés de liberté, Commande par calcul du couple, Commande adaptative floue à mode glissant;

**Abstract :**

The work presented in this thesis aims to study some control methods of a manipulator arm with two degrees of freedom. We quickly presented present the Torque Control approach and its application for the control of the two joints of the robot then, and with the same objectives, the Sliding Mode Control. In order to overcome the model uncertainties and the poorly known parameters which depend totally these two robust control laws, we proposed to study the hybrid approach Adaptive Fuzzy Sliding Mode Control. In this approach we will substitute an Adaptive Takagi-Sugeno Fuzzy Model via the sliding surface to model the strongly coupled uncertain nonlinear dynamics of the robot that constitute the equivalent part of the sliding mode control law.

**Keywords :**

Modeling of a 2-R robot, Torque Control, Adaptive Fuzzy Sliding Mode Control;

## Liste des figures

Chapitre I. - Modélisation du robot manipulateur à deux degrés de liberté .....	4
Figure I.1 - Robot à chaîne ouverte simple .....	5
Figure I.2 - Représentation de Denavit-Hartenberg .....	6
Figure I.3 - Bras manipulateur à deux degrés de liberté (2DDL) .....	13
Figure I.4 - Simulation boucle ouverte : $Teta\_1$ .....	19
Figure I.5 - Simulation boucle ouverte : $Teta\_2$ .....	19
Chapitre II. - Commande par calcul du couple .....	21
Figure II.1 - Système de commande par calcul de couple .....	22
Figure II.2 - Commande par calcul de couple avec PD .....	27
Figure II.3 - Commande par calcul de couple: $Teta\_1$ et l'erreur $e\_1$ .....	29
Figure II.4 - Commande par calcul de couple: $Teta\_2$ et l'erreur $e\_2$ .....	30
Figure II.5 - Forme des perturbations externes .....	29
Figure II.6 - Commande par calcul de couple: $Teta\_1$ et l'erreur $e\_1$ avec perturbations .....	29
Figure II.7 - Commande par calcul de couple: $Teta\_2$ et l'erreur $e\_2$ avec perturbations .....	30
Figure II.8 - Commande par calcul de couple: $Teta\_1$ et l'erreur $e\_1$ avec perturbations et gains PD importants .....	30
Figure II.9 - Commande par calcul de couple: $Teta\_2$ et l'erreur $e\_2$ avec perturbations et gains PD importants .....	30
Chapitre III. - Commande Adaptative Floue A Mode Glissant .....	32
Figure III.1 - Fonction $sign()$ de la commande discrète .....	37
Figure III.2 - Fonction Saturation $sat()$ .....	39
Figure III.3 - Simulation pour $\lambda_1 = 10$ et $K_1 = 10$ .....	39
Figure III.4 - Simulation pour $\lambda_2 = 10$ et $K_2 = 10$ .....	40
Figure III.5 - Variables linguistiques et Fonction d'appartenance .....	44
Figure III.6 - Structure de base d'un régulateur flou .....	45
Figure III.7 - Fonction d'appartenance au sous ensemble flou Négatif : $A=-10$ et $C=-0.3$ .....	49
Figure III.8 - Fonction d'appartenance au sous ensemble flou Zero : $\delta=0.3$ et $C=0$ .....	49
Figure III.9 - Fonction d'appartenance au sous ensemble flou Positif : $A=10$ et $C=0.3$ .....	49
Figure III.10 - Commande AFSMC .....	51
Figure III.11 - Simulation AFSMC : $Teta\_1$ pour $\lambda_1=10, K_1=100, r_1=10$ .....	51
Figure III.12 - Simulation AFSMC : $Teta\_2$ pour $\lambda_2=10, K_2=70, r_2=10$ .....	51

## Liste des tableaux

Chapitre III. - Commande Adaptative Floue A Mode Glissant .....	42
Tableau III-1 - Fonctions d'appartenance .....	42
Tableau III-2 - Table de décision de Mac Vicar-Whelan .....	46
Tableau III-3 - Méthodes de défuzzification .....	47

## Table des matières

Remerciement .....	i
Dédicace.....	ii
Résumé.....	iii
Liste des figures et tableaux.....	v
Table des matières.....	vi
Introduction générale.....	1
Chapitre I. - Modélisation du robot manipulateur à deux degrés de liberté .....	4
I.1. Introduction.....	4
I.2. Structure mécanique et géométrie des robots.....	5
I.2.1. Structure mécanique.....	5
I.2.2. Structure géométrique .....	6
I.3. Modèle géométrique.....	7
I.3.1. Modèle géométrique direct.....	7
I.3.2. Modèle géométrique inverse .....	7
I.4. Modèle cinématique.....	8
I.4.1. Modèle cinématique direct.....	8
I.4.2. Modèle cinématique inverse.....	9
I.5. Modèle dynamique.....	10
I.5.1. Modèle dynamique inverse .....	11
I.5.2. Modèle dynamique direct.....	18
I.6. Simulation .....	19
I.7. Conclusion.....	20
Chapitre II. - Commande par calcul du couple.....	21
II.1. Introduction.....	21
II.2. Commande par calcul de couple .....	21
II.3. Dérivation de la boucle interne (Feed-Forward).....	23
II.4. Conception de la boucle externe (PD).....	26
II.4.1. Choix des gains PD : .....	26
II.4.2. Calcul de couple avec PD.....	27
II.5. Simulations.....	28
II.6. Conclusion.....	31

Chapitre III. - Commande Adaptative Floue A Mode Glissant .....	32
III.1. Introduction .....	32
III.2. Synthèse de la commande robuste par mode glissant .....	33
III.2.1. Le choix de la surface de glissement.....	33
III.2.2. La Condition d'accès au mode glissant.....	34
III.2.3. Synthèse des lois de commande du mode glissant .....	35
III.3. Application au robot à deux degré de liberté .....	37
III.4. Simulations et résultats.....	39
III.5. Conclusion : .....	40
III.6. Synthèse d'un système flou pour calculer la commande à mode glissant.....	41
III.6.1. Rappelle théorique sur la logique floue.....	41
III.6.2. Synthèse d'un système adaptatif flou pour la commande SMC .....	49
III.7. Conclusion.....	52
Conclusion générale.....	53
Bibliographie.....	a



# *Introduction générale*

## Chapitre I Introduction générale

Au cours de la dernière décennie, une attention accrue a été accordée à la commande en vue du suivi de référence (poursuite ou Tracking) pour les bras manipulateurs. La commande en poursuite est nécessaire pour assurer que chaque articulation suit une trajectoire désirée. Beaucoup de recherches ont traité le problème de poursuite étaient basés sur la théorie VSS (Système à Structures Variables) (Wang, Rad, & Chan, 2001) (Yoo & Ham, 2000), sur la théorie adaptative (AZOUI, 2009), sur la logique floue (Wang, Rad, & Chan, 2001) (Wang L. , 1993) (Yoo & Ham, 2000) et le réseau de neurones (Lewis, Kai, & Yesildirek, 1995)(MAKHLOUFI, 2015). Les chercheurs devaient faire face à de nombreuses incertitudes dans les dynamiques des robots (KHALIL, 2002) (KHALIL, 2012) (DOMBRE, 2007), en particulier les incertitudes structurées, telles que les paramètres de charge utile, et non structurées telles que les frottements et les perturbations.

C'est difficile d'obtenir la performance souhaitée lorsque l'algorithme de commande est seulement basé sur le modèle mathématique dynamique du robot. Afin de surpasser ce problème, Slotine (Yoo & Ham, 2000) a proposé l'algorithme de contrôle adaptatif dans lequel les paramètres du robot et de la charge utile ont été mis à jour en utilisant la propriété paramétrable linéaire de la dynamique du robot. Certains ont suggéré les méthodes utilisant des réseaux des neurones avec des stratégies d'apprentissage (Lewis, Kai, & Yesildirek, 1995)(MAKHLOUFI, 2015) et d'autres ont suggéré un contrôleur qui doit dériver la forme linéaire de la dynamique du robot en utilisant un jeu de paramètres correctement sélectionné et régler les paramètres (AALIM & AL-SAIF, 2014). Parce qu'en général, nous devons considérer dix paramètres sur chaque lien du robot manipulateur, ce n'est pas facile d'ajuster les paramètres et de linéariser l'équation dynamique du robot. La structure du réseau neuronal ou flou et les lois d'adaptation devaient être trouvées par des méthodes des d'essais et erreurs.

Afin de surmonter ces difficultés, certains chercheurs ont proposé un système de commande qui utilise une modélisation à base de logique floue comme un compensateur pour toute incertitude (Yoo & Ham, 2000) (Wang, Rad, & Chan, 2001) (Wang L. , 1993). L'algorithme de commande proposé n'a pas besoin de dériver la matrice régressive et régler les paramètres. De nombreuses applications du théorème d'approximation universel ont prouvé qu'un système à base de logique floue Takagi-Sugeno (T-S) est capable de modéliser et se rapprocher uniformément de n'importe quel système non linéaire. Dans ces travaux,

leurs auteurs tentent de résoudre quelques inconvénients de la théorie conventionnelle de commande, des difficultés de la modélisation et de l'analyse pour le système complexe.

(Wang, Rad, & Chan, 2001) ont proposé un algorithme de contrôle flou adaptatif basé sur la dynamique des erreurs sous forme d'une surface de glissement. Cet même équipe a proposé une commande floue en mode glissant et a prouvé la stabilité du système en boucle fermé. (Yoo & Ham, 2000) ont utilisé les systèmes à base de logique flou (FLS) comme compensateurs des fonctions non linéaires inconnues.

Dans ce document, nous synthétisons un système flou FLS et l'utiliser pour compenser les incertitudes sur la dynamique non linéaire couplée d'un robot manipulateur à deux degrés de liberté (2DDL) : un système MIMO fortement couplé avec deux entrée de commande qui sont les couples au niveau des deux articulations, et deux sorties qui sont les angles de rotation des deux portions du bras. Pour réduire l'erreur entre l'incertitude réelle et la sortie du compensateur, nous appliquons une simple et robuste loi d'adaptations étudié en (Wang, Rad, & Chan, 2001). En parallèle nous avons appliqués un autre type de commande robuste au robot à deux degrés de liberté (2DDL) ; à savoir : la commande par calcul de couple.

Après cette introduction générale, ce document est organisé comme suit :

- Le premier chapitre traite la modélisation du robot manipulateur à deux degrés de liberté. Il présente les différents modèles : géométrique, cinématique et dynamique des robots. A la fin les modèles dynamiques direct et indirect sont établis pour un robot manipulateur à deux degrés de liberté en vue de les utiliser pour appliquer nos différents types de commande robustes et non linéaires.
- Le deuxième chapitre présente la commande par calcul de couple ; basée sur la théorie de FeedBack Linearization, cette commande transforme le problème de commande d'un système non linéaire à un problème de commande d'un système linéaire avec un état augmenté. A la fin les résultats de simulation avec la présence des perturbations externes sont commentés et une conclusion est tirée.

- Le troisième chapitre est divisé en deux parties :
  - Une première partie qui applique une commande à mode du glissement conventionnelle avec le modèle dynamique du robot en présence des incertitudes et des perturbations externes.
  - Une deuxième partie qui remplace le modèle dynamique du robot qui fait partie de la commande à mode glissant par un compensateur flou tel que développé en (Wang, Rad, & Chan, 2001).

Nous terminons ce travail par une conclusion sur les résultats de simulations des différents types de commande sur le bras manipulateur à deux degrés de liberté.

*Chapitre I :*  
*Modélisation du robot*  
*manipulateur à 2DDL*

## **Chapitre I. - Modélisation du robot manipulateur à deux degrés de liberté**

### **I.1. Introduction**

Pour développer une stratégie de commande performante pour un robot manipulateur, il est impératif de connaître la cinématique et la dynamique du robot considéré. Et comme les robots manipulateurs sont des systèmes mécaniques poly articulés (MAKHLOUFI, 2015), fortement non linéaires, fortement couplés, d'une dynamique qui peut être mal définis, ils ont des paramètres variables et sont soumis à des perturbations externes. Pour pouvoir faire exécuter une action où une tâche au robot manipulateur, il est nécessaire de modéliser ce système multi variable pour positionner les différents axes. La modélisation de ces systèmes relève très souvent d'une approximation des phénomènes physiques mis en jeu (MAKHLOUFI, 2015), c'est à partir de cette représentation approximative que l'on souhaite construire une commande pour le système réel, cette commande doit être alors robuste dans le sens où elle devra assurer une faible sensibilité aux incertitudes des paramètres, à leurs variations et aux perturbations externes.

Les erreurs de modélisation, les incertitudes sur l'estimation des paramètres physiques ainsi que les différentes perturbations externes influent beaucoup sur la qualité du contrôle. Tous ces facteurs doivent être pris en compte lors de la modélisation et de l'élaboration de la loi de commande pour préserver la qualité et la précision des tâches auxquelles est destiné le robot.

En effet la majorité des tâches confiées aux robots sont délicates et exigent une très grande précision sous des trajectoires précisés et rapides. Dans ce cas le type de commande nécessaire est la commande par découplage non linéaire, cette méthode qui est considérée comme la solution théorique idéale pour la commande de ce type de robot.

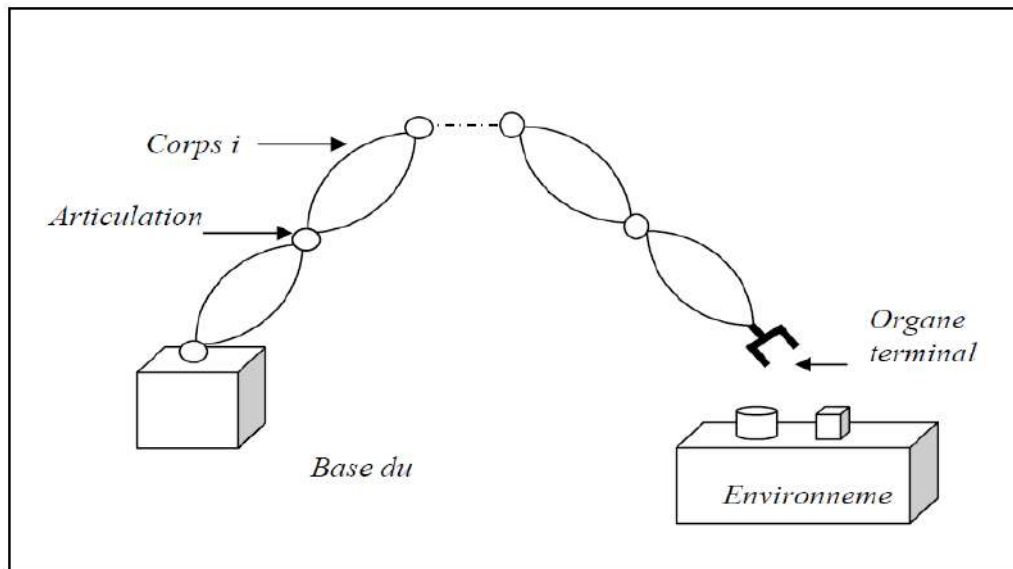
Dans ce chapitre, nous examinerons divers concepts de la théorie de contrôle utilisés dans la commande des robots. A la fin, on élaborera le modèle dynamique d'un bras manipulateur à deux degrés de liberté (2DDL) à travers la mise en forme des équations mathématiques régissant le mouvement de ce bras. Ce dernier sera utilisé dans nos simulations ainsi que l'application des commandes.

## I.2. Structure mécanique et géométrique des robots

### I.2.1. Structure mécanique

La structure mécanique du robot manipulateur peut être divisée en deux parties distinctes comme le montre la **Figure I.1** (AZOUI, 2009) :

- **Organe terminal** : les tâches qui sont dévolues aux robots sont très variées. Pour chaque opération ou travail spécifique, l'organe terminal prend un aspect particulier. Il peut être une pince, une caméra, une pompe de peinture.
- **Élément porteur** : il est composé d'un ensemble de corps souples ou rigides liés par des articulations, servant à déplacer l'organe terminal d'une configuration à une autre.



**Figure I.1 - Robot à chaîne ouverte simple**

Avant de décrire les relations géométriques entre les différents corps du robot, on définit les notions suivantes (AZOUI, 2009) :

- **Degré de liberté** : nombre de paramètres utilisés pour spécifier la configuration d'un élément de la chaîne cinématique par rapport à un autre.
- **Degré de mobilité** : toute articulation est caractérisée par son degré de mobilité  $m$ , c'est-à-dire le nombre de degrés de liberté entre deux corps successifs de la chaîne cinématique ( $0 \leq m \leq 6$ ).

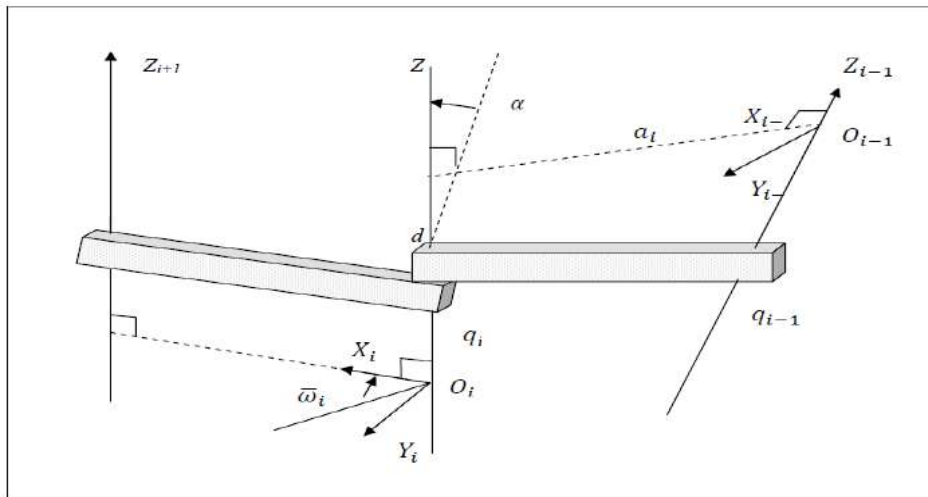
- **Espace articulaire** : représente l'état des corps composant le robot en fonction des variables articulaires, sa dimension  $n$  est égale au nombre de degrés de liberté du robot.
- **Espace opérationnel** : décrit la position et l'orientation de l'organe terminal du robot par rapport à un repère de référence, sa dimension  $m \leq 6$ .

### I.2.2. Structure géométrique

Parmi les diverses méthodes utilisées pour déterminer la position et l'orientation de l'organe terminal par rapport au repère de référence, la plus répandue est celle de Denavit Hartenberg (KHALIL, 2012) représentée dans la **Figure I.2**.

Pour exprimer le passage du repère  $R_{i-1}$  au repère  $R_i$ , on définit les paramètres géométriques suivants :

- Translation selon  $X_{i-1}$  d'une distance  $a_i$ .
- Rotation autour de  $X_{i-1}$  d'un angle  $\alpha_i$ .
- Translation selon  $Z_i$  d'une distance  $d_i$ .
- Rotation autour de  $Z_i$  d'un angle  $\omega_i$ .



**Figure I.2 - Représentation de Denavit-Hartenberg**

La matrice de transformation homogène est donnée comme suit (KHALIL, 2012) :

$$T_i^{i-1} = \text{Trans}(X_{i-1}, a_{i-1}) \text{Rot}(X_{i-1}, \alpha_{i-1}) \text{Trans}(Z_i, d_i) \text{Rot}(Z_i, \omega_i) \quad \text{I.1}$$

Si on désigne par  $T_n^0$  la matrice de transformation reliant le repère  $R_n$  au repère  $R_0$  alors :

$$T_n^0 = T_1^0 T_2^1 T_3^2 \dots T_n^{n-1} \quad \text{I.2}$$



### I.3. Modèle géométrique

Il permet de déterminer la configuration (position, orientation) de l'organe terminal d'un robot en fonction de la configuration de ses liaisons. La modélisation des robots de façon systématique et automatique exige une méthode adéquate pour la description de leur morphologie.

#### I.3.1. Modèle géométrique direct

Le modèle géométrique direct permet de déterminer la position et l'orientation de l'organe terminal du manipulateur par rapport à un repère de référence en fonction des variables articulaires, le modèle s'écrit (KHALIL, 2012):

$$X = f(q) \quad \text{I.3}$$

Ou :

$q = [q_1 \ q_2 \ \dots \ q_n]^T \in R^n$  : Vecteur des variables articulaires.

$X = [x_1 \ x_2 \ \dots \ x_m]^T \in R^m$  : Vecteur des variables opérationnelles,  $m \leq n$ .

Par exemple, si le manipulateur se déplace dans l'espace on pose  $m=6$  (3 coordonnées pour la position et 3 coordonnées pour la rotation). S'il se déplace dans un plan on pose  $m=2$  et si en plus on est concerné par la rotation on pose  $m=3$ .

La position de l'organe terminal peut être définie par des coordonnées cartésiennes, cylindriques ou sphériques. Le choix d'une structure particulière est guidé par les caractéristiques du robot, ainsi que par celle de la tâche à réaliser.

#### I.3.2. Modèle géométrique inverse

Le modèle géométrique inverse permet de déterminer le vecteur des variables articulaires à partir du vecteur de coordonnées opérationnelles, le modèle s'écrit :

$$q = f^{-1}(X) \quad \text{I.4}$$

Parmi les méthodes utilisées pour déterminer le modèle géométrique inverse on cite :

- **Les méthodes géométriques** : permettent de déterminer le vecteur  $q$  par utilisation des transformations géométriques en prenant avantage de la structure particulière du manipulateur considéré.
- **Les méthodes algébriques** : permettent de déterminer le vecteur  $q$  en effectuant des transformations algébriques sur l'équation (I.4). Parmi les méthodes utilisées on cite la méthode de Paul (AZOUI, 2009) qui consiste à multiplier successivement les deux membres du modèle géométrique direct par les matrices homogènes  $T_{i-1}^i$  avec ( $i=1\dots n-1$ ) permettant ainsi d'isoler et d'identifier les variables articulaires l'une après l'autre.

Lors de la résolution du problème géométrique inverse on rencontre pratiquement les situations suivantes (AZOUI, 2009) :

- Solutions en nombre fini, lorsqu'elles peuvent être calculées sans ambiguïté.
- Aucune solution possible lorsque la position désirée ne peut être atteinte par le manipulateur.
- Plusieurs solutions possibles lorsque le manipulateur est redondant ou lorsqu'il passe par une configuration singulière.

#### I.4. Modèle cinématique

##### I.4.1. Modèle cinématique direct

Le modèle cinématique direct permet de déterminer la vitesse de l'organe terminal dans l'espace opérationnel en fonction de la vitesse des variables articulaires.

Le modèle est décrit par l'équation :

$$\dot{X} = J(q)\dot{q} \quad \text{I.I.5}$$

Où  $J(q) \in R^{m \times n}$  est la matrice Jacobienne. L'intérêt de la matrice jacobienne est multiple (KHALIL, 2012) :

- elle est à 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 terminal 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.

L'une des méthodes utilisées pour le calcul de la matrice jacobéenne est la dérivation du modèle géométrique direct :

$$J(q) = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \dots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \dots & \frac{\partial f_m}{\partial q_n} \end{bmatrix}$$

Pour une configuration articulaire  $q$  donnée, le rang  $r$  de la matrice jacobéenne  $J_n$ , (notée  $J$ ), correspond au nombre de degrés de liberté du repère associé à l'organe terminal. Il définit la dimension de l'espace opérationnel accessible dans cette configuration. Le nombre de degrés de liberté  $M$  de l'espace opérationnel d'un robot, est le rang maximal  $r_{max}$  que prend la matrice jacobéenne dans toutes les configurations possibles. Deux cas sont à examiner (KHALIL, 2012):

- si  $M$  égale au nombre de degrés de liberté  $N$  du robot (égal à  $n$  dans le cas des robots en chaîne simple ou à structure arborescente), le robot est non redondant : il possède juste le nombre d'articulations lui permettant de donner le nombre  $M$  de degrés de liberté à son organe terminal.
- si  $N > M$ , le robot est redondant d'ordre  $(N-M)$ . Il dispose de plus d'articulations qu'il n'en faut pour donner le nombre  $M$  de degrés de liberté à son organe terminal.

Que ce soit dans l'un ou dans l'autre cas, pour certaines configurations articulaires, il se peut que le rang  $r$  soit inférieur à  $M$  : on dit que le robot possède une singularité d'ordre  $(M-r)$ . Il perd alors localement la possibilité d'engendrer une vitesse le long ou autour de certaines directions.

Lorsque la matrice  $J$  est carrée, les singularités d'ordre un sont solution de  $\det(J)=0$  où  $\det(J)$  désigne le déterminant de la matrice jacobéenne  $J$  du robot. Elles sont données par  $\det(JJ^T)=0$  dans le cas redondant.

#### I.4.2. Modèle cinématique inverse

Le modèle cinématique inverse permet de déterminer la vitesse des variables articulaires en fonction de la vitesse des variables opérationnelles. Pour les robots manipulateurs non redondants ( $n=m$ ), le modèle s'écrit :

$$\dot{q} = J^{-1}(q)\dot{X} \quad \text{I.6}$$

La solution de l'équation (I.6) existe si  $J$  est de rang plein, cela est valable tant que le robot manipulateur ne passe pas par une configuration singulière. Pour les robots manipulateurs redondants, le modèle cinématique inverse admet plusieurs solutions possibles. Le choix d'une parmi plusieurs est guidé par l'optimisation d'une fonction objective.

Les méthodes employées pour résoudre un tel problème peuvent être classées en deux groupes (MAKHLOUFI, 2015):

- Méthodes d'optimisation cinématique.
- Méthodes d'optimisation dynamique.

Pour obtenir le modèle cinématique inverse, on inverse le modèle cinématique direct en résolvant un système d'équations linéaires. Il existe plusieurs méthodes pour la détermination du modèle cinématique inverse (AZOUI, 2009) (DOMBRE, 2007) en considérant trois cas, selon que le manipulateur est régulier, en configuration singulière ou redondant. Ces méthodes sont soit analytiques, soit numériques :

- **La solution analytique:** les méthodes analytiques conviennent pour résoudre le problème inverse lorsque la matrice jacobienne associée au mécanisme est régulière. Leur mise en œuvre est simplifiée lorsque le poignet du mécanisme possède trois articulations rotoïde d'axes concourants : le problème se réduit alors à l'inversion de deux matrices régulières d'ordre trois. Elle a l'avantage de diminuer considérablement le nombre d'opérations, mais on doit traiter séparément tous les cas singuliers (AZOUI, 2009).
- **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.

### I.5. Modèle dynamique

Le modèle dynamique est la relation entre les couples et/ou forces appliquées aux actionnaires et les positions, vitesses et accélérations. On représente le modèle dynamique par une relation de la forme :

$$\tau = f(q, \dot{q}, \ddot{q}, f_e) \quad \text{I.7}$$

Avec :

$\Gamma$  : Vecteurs des couples/forces des actionnaires, selon que l'articulation soit rotoïde ou prismatique. (dans la suite en écrira tout simplement couple).

$q$  : Vecteur des positions articulaires.

$\dot{q}$  : Vecteur des vitesses articulaires.

$\ddot{q}$  : Vecteur des accélérations articulaires.

$f_e$  : Vecteur représentant les efforts extérieur (fores et moments) qu'exerce le robot manipulateur sur l'environnement.

Parmi les applications du modèle dynamique, on peut citer :

- la simulation, qui utilise le modèle dynamique direct ;
- le dimensionnement des actionneurs (DOMBRE, 2007) ;
- l'identification des paramètres inertiels et des paramètres de frottement du robot (KHALIL, 2002);
- la commande, qui utilise le modèle dynamique inverse (KHALIL, 2002).

### I.5.1. Modèle dynamique inverse

Le modèle dynamique inverse exprime les couples exercés par les actionneurs en fonction des positions, vitesses et accélérations des articulations. Le modèle dynamique inverse est représenté par une relation de la forme :

$$\Gamma = f(q, \dot{q}, \ddot{q}) \quad \text{I.8}$$

$\Gamma$  : Vecteur des couple/forces des actionneurs selon que l'articulation soit rotoïde ou prismatique.

$q$  : Vecteur des positions articulaires.

$\dot{q}$  : Vecteur des vitesses articulaires.

$\ddot{q}$  : Vecteur des accélérations articulaire.

Plusieurs formalismes ont été utilisés pour obtenir le modèle dynamique inverse des manipulateur robots (DOMBRE, 2007) ; Les formalismes les plus souvent utilisés sont:

- **Le formalisme de Lagrange-Euler** : cette méthode est mieux adaptée pour le calcul du modèle dynamique que l'on utilise en simulation.

- **Le formalisme de Newton-Euler** : cette méthode est mieux adaptée pour le calcul du modèle dynamique que l'on utilise en commande.

Nous allons voir dans ce qui suit la modélisation d'un robot manipulateur dynamique rigide à chaîne ouverte simple à deux degrés de liberté et en tenant compte des effets de la vitesse, on va utiliser le formalisme de Lagrange pour déterminer les équations du mouvement.

➤ **Formalisme de Lagrange (Robot à 2DDL)**

Le formalisme de Lagrange décrit les équations du mouvement en termes de travail et de l'énergie du robot. La forme générale représentant le mouvement d'un robot manipulateur constitué de  $n$  degrés de liberté et de  $n$  articulations, s'écrit suivant Lagrange :

$$\Gamma_i = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} \quad \text{I.9}$$

Avec :  $i=1,2,\dots,n$ .

$L$  : Fonction lagrangienne de robot égale à :  $L = E_c - E_p$

$E_c$  : Energie cinétique totale du robot manipulateur

$E_p$  : Energie potentielle totale du robot manipulateur

$q \in R^n$  : Vecteur des positions articulaires

$\dot{q} \in R^n$  : Vecteur des vitesses articulaires.

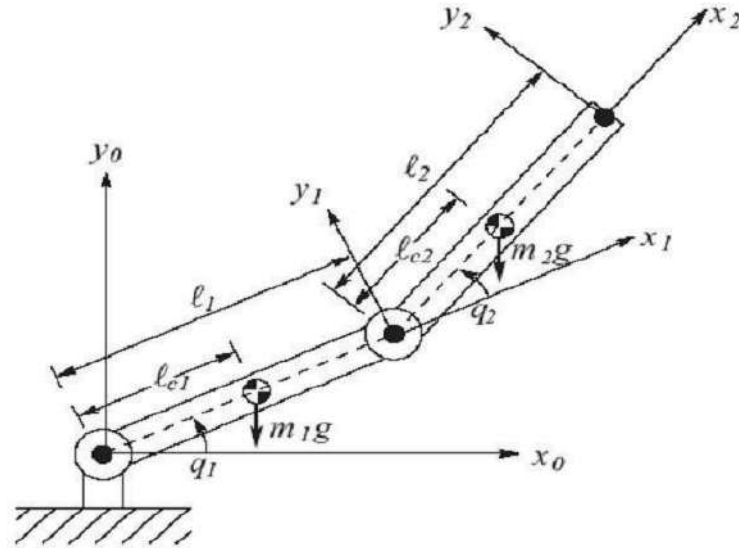
Les forces généralisées produites par cette méthode ne représentent pas nécessairement des forces physiques réelles qui peuvent être identifiées et ne sont pas constituées de forces contraintes qui n'effectuent aucun travail, car ces forces sont éliminées. L'équation pour les forces généralisées s'écrit (AZOUI, 2009):

$$\Gamma_{q_i} = \sum \bar{F} \cdot \frac{\partial \bar{P}}{\partial \bar{q}} = \sum \left( F_x \cdot \frac{\partial x}{\partial q_i} + F_y \cdot \frac{\partial y}{\partial q_i} + F_z \cdot \frac{\partial z}{\partial q_i} \right) \quad \text{I.10}$$

Où :  $F_x, F_y, F_z$  : sont les composantes des forces et  $x, y, z$  sont les coordonnées des points d'application de ces forces définies en fonction des coordonnées généralisées pour permettre leurs dérivées.

Dans le cas d'une articulation rotoïde  $q_i = \theta_i$  tandis que pour une prismatique  $q_i = d_i$ .

Dans notre travail, nous avons considéré un robot manipulateur à deux degrés de liberté (2DDL), donné par la **Figure I.3**, dont le mouvement des articulations est rotoïde, nous avons donc  $q_i = \theta_i$ .



**Figure I.3 - Bras manipulateur à deux degré de liberté (2DDL)**

Tel que :  $q = [\theta_1 \ \theta_2]^T$  et  $\Gamma = [\tau_1 \ \tau_2]^T$

A partir de la **Figure I.3**, nous déterminons les coordonnées du bras manipulateur par rapport au repère  $(x, y)$ , ce qui donne :

$$\begin{cases} x_1 = l_{c1} \cos \theta_1 \\ y_1 = l_{c1} \sin \theta_1 \end{cases} \quad \text{I.11}$$

$$\begin{cases} x_2 = l_1 \cos \theta_1 + l_{c2} \cos(\theta_1 + \theta_2) \\ y_2 = l_1 \sin \theta_1 + l_{c2} \sin(\theta_1 + \theta_2) \end{cases} \quad \text{I.12}$$

Ce qui donne :

$$\begin{cases} \dot{x}_1 = -l_{c1} \dot{\theta}_1 \sin \theta_1 \\ \dot{y}_1 = l_{c1} \dot{\theta}_1 \cos \theta_1 \end{cases} \quad \text{I.13}$$

$$\begin{cases} \dot{x}_1^2 = l_{c1}^2 \dot{\theta}_1^2 \sin^2 \theta_1 \\ \dot{y}_1^2 = l_{c1}^2 \dot{\theta}_1^2 \cos^2 \theta_1 \end{cases} \quad \text{I.14}$$

Et

$$\begin{cases} \dot{x}_2 = -(l_1 \dot{\theta}_1 \sin \theta_1 + l_{c2}(\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_1 + \theta_2)) \\ \dot{y}_2 = l_1 \dot{\theta}_1 \cos \theta_1 + l_{c2}(\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_1 + \theta_2) \end{cases} \quad \text{I.15}$$

$$\begin{cases} \dot{x}_2^2 = l_1^2 \dot{\theta}_1^2 \sin^2 \theta_1 + l_{c2}^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \sin^2(\theta_1 + \theta_2) + 2l_1 l_{c2} \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \sin \theta_1 \sin(\theta_1 + \theta_2) \\ \dot{y}_2^2 = l_1^2 \dot{\theta}_1^2 \cos^2 \theta_1 + l_{c2}^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \cos^2(\theta_1 + \theta_2) + 2l_1 l_{c2} \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \cos \theta_1 \cos(\theta_1 + \theta_2) \end{cases} \quad \text{I.16}$$

- Détermination de l'énergie cinétique  $E_c$  :

$$E_c = \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + \frac{1}{2} m_1 v_1^2 + \frac{1}{2} m_2 v_2^2 \quad \text{I.17}$$

Ce qui donne :

$$E_c = \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} I_2 \dot{\theta}_1^2 + \frac{1}{2} I_2 \dot{\theta}_2^2 + I_2 \dot{\theta}_1 \dot{\theta}_2 + \frac{1}{2} m_1 v_1^2 + \frac{1}{2} m_2 v_2^2 \quad \text{I.18}$$

Avec :

$$\begin{cases} v_1^2 = \dot{x}_1^2 + \dot{y}_1^2 = l_{c1}^2 \dot{\theta}_1^2 \\ v_2^2 = \dot{x}_2^2 + \dot{y}_2^2 = l_1^2 \dot{\theta}_1^2 + l_{c2}^2 \dot{\theta}_1^2 + l_{c2}^2 \dot{\theta}_2^2 + 2l_{c2}^2 \dot{\theta}_1 \dot{\theta}_2 + 2l_1 l_{c2} \dot{\theta}_1^2 \cos \theta_2 + 2l_1 l_{c2} \dot{\theta}_1 \dot{\theta}_2 \cos \theta_2 \end{cases} \quad \text{I.19}$$

L'équation (I.18) devient :

$$\begin{aligned} E_c = & \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} I_2 \dot{\theta}_1^2 + \frac{1}{2} I_2 \dot{\theta}_2^2 + I_2 \dot{\theta}_1 \dot{\theta}_2 + \frac{1}{2} m_1 l_{c1}^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 (l_1^2 \dot{\theta}_1^2 + l_{c2}^2 \dot{\theta}_1^2 + \\ & l_{c2}^2 \dot{\theta}_2^2 + 2l_{c2}^2 \dot{\theta}_1 \dot{\theta}_2 + 2l_1 l_{c2} \dot{\theta}_1^2 \cos \theta_2 + 2l_1 l_{c2} \dot{\theta}_1 \dot{\theta}_2 \cos \theta_2) \end{aligned} \quad \text{I.20}$$

- Détermination de l'énergie potentielle  $E_p$  :

$$E_p = m_1 g h_1 + m_2 g h_2 \quad \text{I.21}$$

Avec :

$$\begin{cases} h_1 = y_1 = l_{c1} \sin \theta_1 \\ h_2 = y_2 = l_1 \sin \theta_1 + l_{c2} \sin(\theta_1 + \theta_2) \end{cases} \quad \text{I.22}$$

L'équation (I.21) devient :

$$E_p = m_1 g l_{c1} \sin \theta_1 + m_2 g (l_1 \sin \theta_1 + l_{c2} \sin(\theta_1 + \theta_2)) \quad \text{I.23}$$



- Détermination du Lagrangien :  $L = E_c - E_p$

$$L = \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}I_2\dot{\theta}_1^2 + \frac{1}{2}I_2\dot{\theta}_2^2 + I_2\dot{\theta}_1\dot{\theta}_2 + \frac{1}{2}m_1l_{c1}^2\dot{\theta}_1^2 + \frac{1}{2}m_2(l_1^2\dot{\theta}_1^2 + l_{c2}^2\dot{\theta}_1^2 + l_{c2}^2\dot{\theta}_2^2 + 2l_{c2}\dot{\theta}_1\dot{\theta}_2 + 2l_1l_{c2}\dot{\theta}_1^2 \cos \theta_2 + 2l_1l_{c2}\dot{\theta}_1\dot{\theta}_2 \cos \theta_2) - m_1gl_{c1} \sin \theta_1 - m_2g(l_1 \sin \theta_1 + l_{c2} \sin(\theta_1 + \theta_2)) \quad I.24$$

- Détermination du modèle dynamique du robot

Nous appliquons maintenant l'équation d'Euler-Lagrange donnée par équation (I.9) pour déterminer le modèle dynamique correspondant au robot manipulateur à 2DDL, soit :

$$\Gamma_i = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} \text{ avec } i=1,2 \text{ et } q_i=\theta_i$$

Nous avons :

$$\frac{\partial L}{\partial \theta_1} = -m_1gl_{c1} \cos \theta_1 - m_2g(l_1 \cos \theta_1 + l_{c2} \cos(\theta_1 + \theta_2)) \quad I.25$$

$$\frac{\partial L}{\partial \dot{\theta}_1} = I_1\dot{\theta}_1 + I_2\dot{\theta}_1 + I_2\dot{\theta}_2 + m_1l_{c1}^2\dot{\theta}_1 + m_2(l_1^2\dot{\theta}_1 + l_{c2}^2\dot{\theta}_1 + l_{c2}^2\dot{\theta}_2 + 2l_1l_{c2}\dot{\theta}_1 \cos \theta_2 + l_1l_{c2}\dot{\theta}_2 \cos \theta_2) \quad I.26$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) = I_1\ddot{\theta}_1 + I_2\ddot{\theta}_1 + I_2\ddot{\theta}_2 + m_1l_{c1}^2\ddot{\theta}_1 + m_2(l_1^2\ddot{\theta}_1 + l_{c2}^2\ddot{\theta}_1 + l_{c2}^2\ddot{\theta}_2 + 2l_1l_{c2}\ddot{\theta}_1 \cos \theta_2 - 2l_1l_{c2}\dot{\theta}_1\dot{\theta}_2 \sin \theta_2 + l_1l_{c2}\ddot{\theta}_2 \cos \theta_2 + l_1l_{c2}\dot{\theta}_2^2 \sin \theta_2) \quad I.27$$

Ce qui donne :

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) - \frac{\partial L}{\partial \theta_1} = (I_1 + I_2 + m_1l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1l_{c2} \cos \theta_2))\ddot{\theta}_1 + (I_2 + m_2(l_{c2}^2 + (-m_2l_1l_{c2} \sin \theta_2)\dot{\theta}_2^2 + (-2m_2l_1l_{c2} \sin \theta_2)\dot{\theta}_1\dot{\theta}_2 + m_1gl_{c1} \cos \theta_1 + m_2g(l_1 \cos \theta_1 + l_{c2} \cos(\theta_1 + \theta_2))) \quad I.28$$

Nous avons aussi :

$$\frac{\partial L}{\partial \theta_2} = -m_2(l_1l_{c2}\dot{\theta}_1^2 \sin \theta_2 + l_1l_{c2}\dot{\theta}_1\dot{\theta}_2 \sin \theta_2) - m_2gl_{c2} \cos(\theta_1 + \theta_2) \quad I.29$$

$$\frac{\partial L}{\partial \dot{\theta}_2} = I_2\dot{\theta}_2 + I_2\dot{\theta}_1 + m_2(l_{c2}^2\dot{\theta}_2 + l_{c2}^2\dot{\theta}_1 + l_1l_{c2}\dot{\theta}_1 \cos \theta_2) \quad I.30$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) = I_2 \ddot{\theta}_2 + I_2 \ddot{\theta}_1 + m_2 (l_{c2}^2 \ddot{\theta}_2 + l_{c2}^2 \ddot{\theta}_1 + l_1 l_{c2} \ddot{\theta}_1 \cos \theta_2 - l_1 l_{c2} \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2) \quad \text{I.31}$$

Ce qui donne :

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} = (I_2 + m_2 (l_{c2}^2 + l_1 l_{c2} \cos \theta_2)) \ddot{\theta}_1 + (I_2 + m_2 l_{c2}^2) \ddot{\theta}_2 - l_1 l_{c2} \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 + (m_2 l_1 l_{c2} \sin \theta_2) \dot{\theta}_1^2 + m_2 g l_{c2} \cos(\theta_1 + \theta_2) \quad \text{I.32}$$

A partir de l'équation (I.9), nous avons :

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) - \frac{\partial L}{\partial \theta_1} = \tau_1 \\ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} = \tau_2 \end{cases} \quad \text{I.33}$$

En robotique, si le mouvement des articulations constituant le robot manipulateur est rotoïde, l'expression du couple peut être s'exprimer sous la forme matricielle suivante, ce qui correspond au modèle dynamique du robot (AZOUI, 2009):

$$\Gamma_i = M(\theta) \ddot{\theta} + C(\theta, \dot{\theta}) \dot{\theta} + G(\theta) \quad \text{avec } i=1,2 \quad \text{I.34}$$

$\Gamma$  : Vecteur des couples ou des forces généralisés.

$\theta$  : Vecteur des variables articulaires du bras manipulateur.

$\dot{\theta}$  : Vecteur des vitesses articulaires.

$\ddot{\theta}$  : Vecteur des accélérations articulaires.

$M(\theta)$  : Matrice d'inertie de dimension  $(n \times n)$ .

$C(\theta, \dot{\theta}) \dot{\theta}$  : Vecteur des forces centrifuges et de Coriolis.

$G(\theta)$  : Vecteur de force de gravité de la charge.

S'il est nécessaire de considérer les perturbations et les frottements aux niveaux des articulations on ajoute des termes  $\tau_d$  et  $\tau_f$  à l'équation (I.34), on obtient alors :

$$\Gamma_i = M(\theta) \ddot{\theta} + C(\theta, \dot{\theta}) \dot{\theta} + G(\theta) + \tau_f + \tau_d \quad \text{I.35}$$

$\tau_d$  : perturbations externes.

$\tau_f$  : terme des frottements au niveau des articulations

Différents modèles mathématiques peuvent être utilisés pour exprimer  $\tau_f$ . Pour une modélisation simple, on choisit :

$$\tau_f = k_{fv} \cdot \dot{q} + k_{fc} \cdot \text{sign}(\dot{q}) = F(\dot{q}) \quad \text{I.36}$$

Où  $k_{fv}$  et  $k_{fc}$  sont respectivement le coefficient de frottements visqueux et de Coulomb.

Soit  $\Gamma_i = [\tau_1 \tau_2]^T$  Le développement de l'équation (I.34) donne :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 2h_{11}\dot{\theta}_2 & h_{12}\dot{\theta}_2 \\ h_{21}\dot{\theta}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad \text{I.37}$$

Ce qui donne :

$$\begin{cases} \tau_1 = M_{11}\ddot{\theta}_1 + M_{12}\ddot{\theta}_2 + 2h_{11}\dot{\theta}_2\dot{\theta}_1 + h_{12}\dot{\theta}_2^2 + g_1 \\ \tau_2 = M_{21}\ddot{\theta}_1 + M_{22}\ddot{\theta}_2 + h_{21}\dot{\theta}_1^2 + g_2 \end{cases} \quad \text{I.38}$$

L'identification entre équation (I.37) et équation (I.34) permet de définir les éléments suivants:

$$M_{11} = I_1 + I_2 + m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos \theta_2)$$

$$M_{12} = M_{21} = I_2 + m_2 (l_{c2}^2 + l_1 l_{c2} \cos \theta_2)$$

$$M_{22} = I_2 + m_2 l_{c2}^2$$

$$h_{11} = h_{12} = -h_{21} = -m_2 l_1 l_{c2} \sin \theta_2$$

$$g_1 = m_1 g l_{c1} \cos \theta_1 + m_2 g (l_1 \cos \theta_1 + l_{c2} \cos(\theta_1 + \theta_2))$$

$$g_2 = m_2 g l_{c2} \cos(\theta_1 + \theta_2)$$

La matrice d'inertie M est égale à :

$$M(\theta) = \begin{bmatrix} I_1 + I_2 + m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos \theta_2) & I_2 + m_2 (l_{c2}^2 + l_1 l_{c2} \cos \theta_2) \\ I_2 + m_2 (l_{c2}^2 + l_1 l_{c2} \cos \theta_2) & I_2 + m_2 l_{c2}^2 \end{bmatrix} \quad \text{I.39}$$

Le vecteur de termes Coriolis et centrifuge est donnée par :

$$C(\theta, \dot{\theta}) = \begin{bmatrix} -2m_2 l_1 l_{c2} \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 & -m_2 l_1 l_{c2} \dot{\theta}_2^2 \sin \theta_2 \\ m_2 l_1 l_{c2} \dot{\theta}_1^2 \sin \theta_2 & 0 \end{bmatrix} \quad \text{I.40}$$

Le vecteur des termes de forces de gravité G est donné par l'équation suivante:

$$G(\theta) = \begin{bmatrix} m_1 g l_{c1} \cos \theta_1 + m_2 g (l_1 \cos \theta_1 + l_{c2} \cos(\theta_1 + \theta_2)) \\ m_2 g l_{c2} \cos(\theta_1 + \theta_2) \end{bmatrix} \quad \text{I.41}$$

### I.5.2. Modèle dynamique direct

Pour simuler le comportement du robot et de sa boucle de commande, on utilise le modèle dynamique direct. Même si les contraintes de temps de calcul sont moins critiques que pour la commande, il est cependant intéressant de disposer d'un modèle performant, ou le modèle dynamique direct exprime la position, la vitesse et l'accélération des articulations en fonction du couple appliqué. Il est obtenu par inversion du modèle précédent, il s'écrit :

$$\ddot{q} = M^{-1}(q)(\tau - C(q, \dot{q}) - F(\dot{q}) - G(q) - \tau_d) \quad \text{I.42}$$

Nous présentons ici deux méthodes, l'une qui procède par inversion de la matrice d'inertie  $M$ , l'autre qui calcule les accélérations articulaires sans inversion.

➤ **Calcul par inversion de la matrice d'inertie  $M$  du robot**

A partir de l'équation (I.35) :

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d \quad \text{I.43}$$

Et en supposant l'existence d'un effort extérieur sur l'organe terminal. Ou  $M$  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 est définie positive (DOMBRE, 2007). Ses éléments sont fonction des variables articulaires  $q$ .

Le modèle dynamique direct s'écrit :

$$\ddot{q} = M^{-1}(q)(\Gamma - H(q, \dot{q})) \quad \text{I.44}$$

Avec :

$$H(q, \dot{q}) = C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d$$

➤ **Calcul sans inversion de la matrice d'inertie**

Cette méthode utilise les équations de Newton-Euler et permet de calculer les accélérations articulaires sans avoir à inverser la matrice d'inertie  $M$  du robot (KHALIL, 2002).

Le calcul de la matrice  $M$  par une méthode lagrangienne peut être moins couteux en temps. On peut cependant utiliser l'algorithme de Newton-Euler pour calculer ses éléments.

## I.6. Simulations

Les forces de frottement sont données par le modèle :

$$F(\dot{q}) = 0.02x\text{sign}(\dot{q})$$

Les résultats de simulations en boucle ouverte, sans perturbations externes, à une excitation sinusoïdale d'amplitude  $10^\circ$  et de fréquence 1 rad/sec, sont donnés sur les figures ci-dessous.

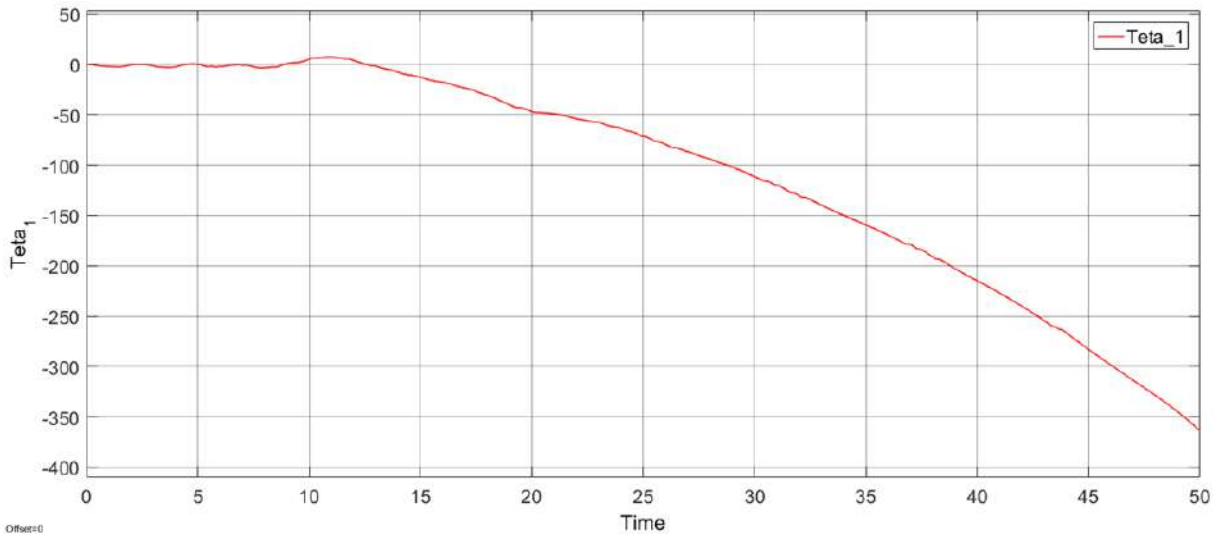


Figure I.4 - Simulation boucle ouverte : Teta\_1

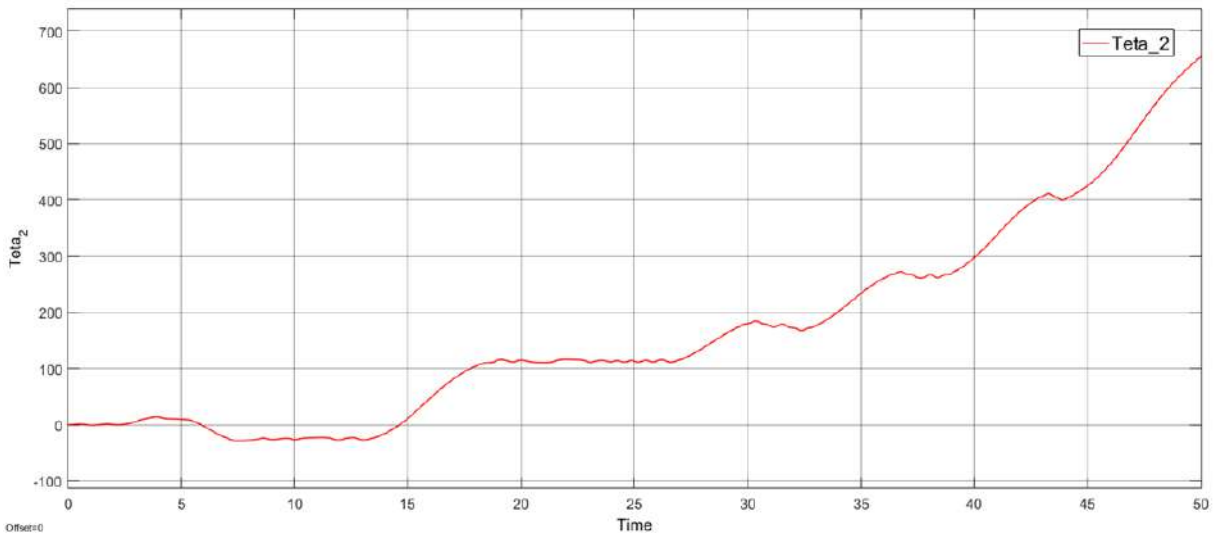


Figure I.5 - Simulation boucle ouverte : Teta\_2

**Commentaire :** Le système en boucle ouverte est instable et d'où la nécessité d'une commande en boucle fermée pour le stabiliser et réaliser les objectifs de commande (suivi de trajectoire).

### **I.7. Conclusion**

Dans ce chapitre nous avons exposé les différents modèles utilisés pour décrire les mouvements des articulations d'un robot manipulateur, dans le but de fournir les équations mathématiques qui permettent de programmer, commander et exécuter les tâches désirées. Ceci est réalisé en fonction du modèle géométrique, cinématique et dynamique. Nous avons vu que :

- Le choix du modèle géométrique est fait selon le type de la tâche à réaliser ainsi que l'architecture du manipulateur.
- Le calcul du modèle cinématique inverse pour les manipulateurs redondants est réalisé en définissant un objectif secondaire en plus de l'objectif principal, afin de choisir une solution parmi plusieurs. Cet objectif consiste à optimiser une fonction de nature cinématique ou dynamique.
- Le calcul du modèle dynamique peut être effectué soit par la méthode de Newton-Euler ou par celle de Lagrange. La première nécessite moins de calcul, elle est destinée pour des applications en temps réel. Tandis que la deuxième s'avère intéressante pour la simulation.

Dans les chapitres suivants, sera l'objet d'une étude sur la simulation des différentes commandes appliquées au robot manipulateur à deux degrés de liberté.

*Chapitre II :*  
*Commande par Calcul*  
*de Couple*

## Chapitre II - Commande par calcul du couple

### II.1. Introduction

La résolution du problème de la commande des robots manipulateurs nécessite la détermination d'un ensemble d'entrées articulaires (les couples  $\tau$ ) qui résulte par le suivi de l'organe effecteur d'une trajectoire désirée, spécifiée typiquement par des séquences de positions et de vecteurs d'orientation de l'organe effecteur  $x$  ou par une trajectoire continue. Plusieurs types de commandes ont été étudiées pour les robots manipulateurs, la méthode la plus simple et qui reste toujours employée pour les manipulateurs industriels est la commande articulaire indépendante ou chaque articulation du manipulateur est commandée comme un système à une seule entrée et une seule sortie (S.I.S.O) (MAKHLOUFI, 2015).

La stratégie de cette commande est que chaque actionneur d'une articulation est contrôlé indépendamment : tous les effets de couplage entre les deux articulations sont ignorées ou traitées comme des perturbations. Cette commande a donné des résultats satisfaisants pour les simples déplacements, mais n'est pas convenable pour les déplacements rapides et ceux avec une large variation.

La méthode de la commande articulaire indépendante peut être considérée comme une forme simplifiée de la méthode du calcul du couple qui est la technique commune la plus utilisée pour les robots manipulateurs, la plupart des méthodes de commandes des manipulateurs peuvent être considérés comme des cas spéciaux de cette technique. La méthode du calcul du couple en elle-même est une application de la technique de linéarisation du retour (Feed-Back Linearization) pour les systèmes non linéaires.

### II.2. Commande par calcul de couple

Au fil des ans il a été proposé de nombreuses sortes de systèmes de commande de robot. Comme il arrive, la plupart d'entre eux peuvent être considérés comme des cas particuliers de la classe des contrôleurs type calcul du couple. Cette technique est aussi connue sous les noms de commande dynamique. Lorsque l'application exige des évolutions rapides avec des contraintes dynamiques, la commande doit prendre en compte les forces d'interaction. Ce type de technique considère l'ensemble des articulations et, pour les découpler, utilise la théorie du découplage non linéaire.



Cette théorie utilise le modèle dynamique du robot pour le calcul de la loi de commande, ce qui conduit à des lois de commande centralisées non linéaires. Des signaux d'anticipation peuvent être utilisés pour minimiser des effets non linéaires (MAKHLOUFI, 2015). Le Calcul de couple (CT) en même temps est une application particulière de la rétroaction (feedback) linéarisation des systèmes non linéaires, qui a gagné en popularité dans la théorie des systèmes modernes. En fait, une façon de classer les systèmes de contrôle du robot est de les diviser comme « calcul de couple » ou « non calcul de couple ». Calcul de couple comme la commande apparaît dans la commande robuste, la commande adaptative, commande par apprentissage, et ainsi de suite. La commande par calcul de couple nous permet de tirer facilement des contrôleurs des robots très efficaces, tout en fournissant un cadre pour rassembler le contrôle commun indépendant classique et quelques techniques de conception modernes.

La stratégie de la commande par calcul de couple peut être considérée comme un contrôleur en deux parties, une section étant basée modèle alors que l'autre étant la portion de la loi de servocommande. La Figure II.1 montre la stratégie de la commande par calcul de couple, montrant les deux boucles internes et externes.

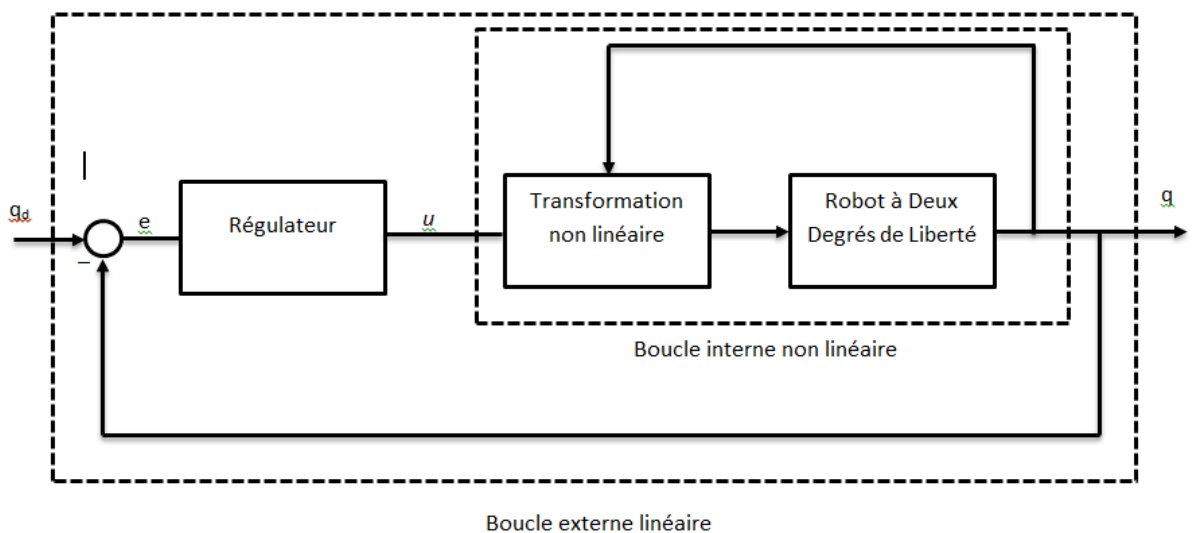


Figure II.1 - Système de commande par calcul de couple

### II.3. Dérivation de la boucle interne (Feed-Forward)

La dynamique du robot manipulateur est donné par :

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d \quad \text{II.1}$$

Ou simplement :

$$\tau = M(q)\ddot{q} + N(q, \dot{q}) + \tau_d \quad \text{II.2}$$

Avec :

$$N(q, \dot{q}) = C(q, \dot{q})\dot{q} + G(q) + F(\dot{q})$$

$N(q, \dot{q})$ : Les limites non linéaires

$F(\dot{q}) = F_v(\dot{q}) + F_d(\dot{q})$  : est le vecteur des coefficients de frottement.

$F_v(\dot{q})$  : est le vecteur  $n \times 1$  des coefficients de frottement visqueux

$F_d(\dot{q})$  : est le vecteur  $n \times 1$  des coefficients de frottement dynamique.

$q \in R^n$ : Variable de joint

$q$  et  $\dot{q}$  sont respectivement les vecteurs  $n \times 1$  de la position et de la vitesse communes.

$\tau$ : est le vecteur  $n \times 1$  des couples appliqués par le bras. Si cette équation comprend la dynamique de l'actionneur à moteur, alors  $\tau$  est une tension d'entrée.

$M(q)$  est la matrice d'inertie  $n \times n$  bien déterminée positive et symétrique,

$C(q, \dot{q})$  est le vecteur des forces centrifuges et de Coriolis.

$G(q)$  est le vecteur  $n \times 1$  des couples de la gravité.

$\tau_d$  : des perturbations inconnues bornées (incluant les dynamiques non structurées, et non modélisées).

Pour assurer le suivi de trajectoire par la variable joint  $q$ , nous choisissons une commande qui stabilise le system, de sorte que l'erreur  $e(t)$  tende vers zéro.

$$e = q_d - q \quad \text{II.3}$$

Pour démontrer l'influence de l'entrée  $\tau$  sur l'erreur de poursuite, nous dérivons deux fois pour obtenir :

$$\text{Vecteur de l'erreur de vitesse : } \dot{e} = \dot{q}_d - \dot{q} \quad \text{II.4}$$

$$\text{Vecteur de l'erreur d'accélération : } \ddot{e} = \ddot{q}_d - \ddot{q} \quad \text{II.5}$$

Résoudre maintenant  $\ddot{q}$  dans (II.2) on obtient :

$$\ddot{q} = M^{-1}(\tau - N - \tau_d) \quad \text{II.6}$$

et son remplacement dans l'équation (II.5):

$$\ddot{e} = \ddot{q}_d + M^{-1}(N + \tau_d - \tau) \quad \text{II.7}$$

Définir la fonction d'entrée de commande :

$$u = \ddot{q}_d + M^{-1}(N - \tau) \quad \text{II.8}$$

Et la fonction de perturbation :

$$w = M^{-1} \tau_d \quad \text{II.9}$$

Nous pouvons définir un état  $x(t) \in R^{2xn}$  par:

$$x = [e \quad \dot{e}]^T \quad \text{II.10}$$

Et écrire la dynamique d'erreur de poursuite par :

$$\frac{d}{dt} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u + \begin{bmatrix} 0 \\ I \end{bmatrix} w \quad \text{II.11}$$

Il s'agit d'un système d'erreur linéaire sous forme canonique de Brunovsky (MAKHLOUFI, 2015) constitué de  $n$  paires de doubles intégrateurs  $\frac{1}{s^2}$  un par joint. Elle est entraînée par l'entrée de commande  $u(t)$  et la perturbation  $w(t)$ .

La transformation de linéarisation de feedback en (II.8) peut être inversée pour donner :

$$\tau = M(\ddot{q}_d - u) + N \quad \text{II.12}$$

Nous appelons cela la loi de commande par calcul de couple. L'importance de ces manipulations est le suivant : Il y a eu aucune transformation espace-état en passant de (II.1) à (II.11).

Par conséquent, si nous choisissons une commande  $u(t)$  qui stabilise (II.11) de sorte que  $e(t)$  tend vers zéro, puis l'entrée de commande non linéaire donnée par  $\tau(t)$  (II.12) va provoquer la poursuite de la trajectoire des articulations du robot manipulateur décrit par l'équation (II.1).

En fait, en remplaçant le couple (II.12) dans (II.2) ça donne :

$$M\ddot{q} + N + \tau_d = M(\ddot{q}_d - u) + N \quad \text{II.13}$$

Ou bien :

$$\ddot{e} = u + M^{-1}\tau_d \quad \text{II.14}$$

Ce qui est exactement (II.11).

La stabilisation du système (II.11) n'est pas difficile. En fait, la transformation non linéaire (II.8) a converti un problème de conception des contrôles non linéaires complexes en un problème de conception simple pour un système linéaire constitué de  $n$  sous-systèmes découplés, obéissant chacun les lois de Newton (MAKHLOUFI, 2015). Le système de contrôle résultant apparaît dans la **Erreur ! Source du renvoi introuvable.** Il est important de noter qu'il est constitué d'une boucle non linéaire interne plus un signal de commande externe  $u(t)$ . Il existe plusieurs façons pour choisir  $u(t)$ ; Comme  $u(t)$  dépend de  $q(t)$  et  $\dot{q}(t)$ , la boucle externe sera une boucle de rétroaction (feedback).

Il est important de savoir que le calcul de couple dépend de l'inversion de la dynamique du robot, et est d'ailleurs parfois appelée commande de la dynamique inverse. En fait, (II.12) montre que  $\tau(t)$  est calculé en substituant  $\ddot{q}_d - u$  pour  $\ddot{q}$  dans (II.2). Une bonne façon de calculer le couple  $\tau(t)$  est d'utiliser la formulation dynamique de Newton-Euler de l'inverse efficace avec du  $\ddot{q}_d - u$  pour à la place de  $\ddot{q}$ .

#### II.4. Conception de la boucle externe (PD)

Une façon de sélectionner le signal de commande auxiliaire  $u(t)$  est la proportionnelle et dérivée (PD).

$$u = -k_d \dot{e} - k_p e \quad \text{II.15}$$

Ensuite, l'entrée globale de bras de robot devient :

$$\tau = M(q)(\ddot{q}_d + k_d \dot{e} + k_p e) + N(q, \dot{q}) \quad \text{II.16}$$

La dynamique d'erreur en boucle fermée est :

$$\ddot{e} + k_d \dot{e} + k_p e = w \quad \text{II.17}$$

Ou sous forme d'espace-état :

$$\frac{d}{dt} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -k_d & -k_p \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} w \quad \text{II.18}$$

##### II.4.1. Choix des gains PD :

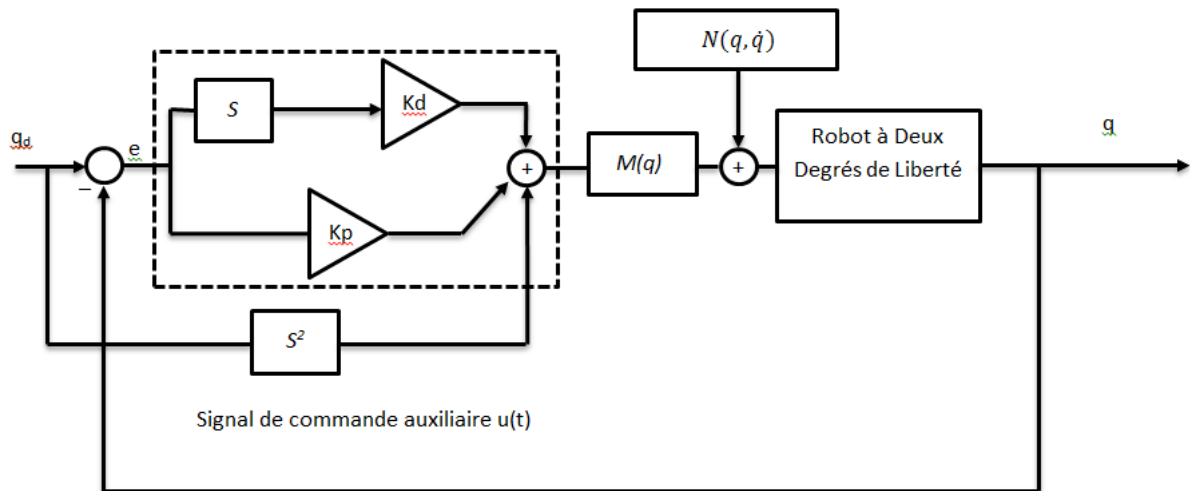
Il est habituel de prendre la matrices des gains diagonales  $n \times n$  de telle sorte que :

$$k_d = \text{diag}\{k_{di}\} \text{ et } k_p = \text{diag}\{k_{pi}\} \text{ avec } i=1,2$$

Il est important de noter qu'en choisissant la diagonale de matrices de gain de PD dans le contrôle détaché au niveau de boucle extérieure, il n'aboutit pas à une stratégie de contrôle commun indépendant. C'est parce que la multiplication par le terme couplé  $M(q)$  et l'addition des limites non linéaires (feed forward)  $N(q, \dot{q})$  dans la boucle intérieure perturbe le signal  $u(t)$  par tous les joints (articulations). Ainsi, les informations sur toutes les positions  $q(t)$  et des accélérations  $\ddot{q}(t)$  est généralement nécessaires pour calculer le contrôle  $\tau(t)$  pour n'importe quel joint (articulation) donné.

### II.4.2. Calcul de couple avec PD

Le schéma de commande par calcul de couple avec PD est donné par la **Figure II.2**.



**Figure II.2 - Commande par calcul de couple avec PD**

Le modèle dynamique du robot est donné par:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d \quad \text{II.19}$$

La loi de commande par calcul de couple (PD) est donnée par:

$$\tau = M(q)(\ddot{q}_d + k_d\dot{e} + k_p e) + N(q, \dot{q}) \quad \text{II.20}$$

En choisissant les matrices de gain  $K_d$  et  $K_p$  comme des matrices diagonales avec des valeurs positives le long de la diagonale, cette erreur du système (II.19) peut être rendue asymptotiquement stable. Il est important de noter que malgré la sélection des matrices diagonales de gain, il en résulte un découplage de la commande au niveau de la boucle externe mais ceci n'implique pas une stratégie de commande articulaire découplée.

Parce que la multiplication par la matrice d'inertie et l'addition de termes non linéaires dans la loi basée modèle laisse la loi de commande  $u(t)$  affecte toutes les articulations. Pour calculer l'entrée couple de n'importe quelle articulation, les positions et vitesses des autres articulations sont nécessaires.

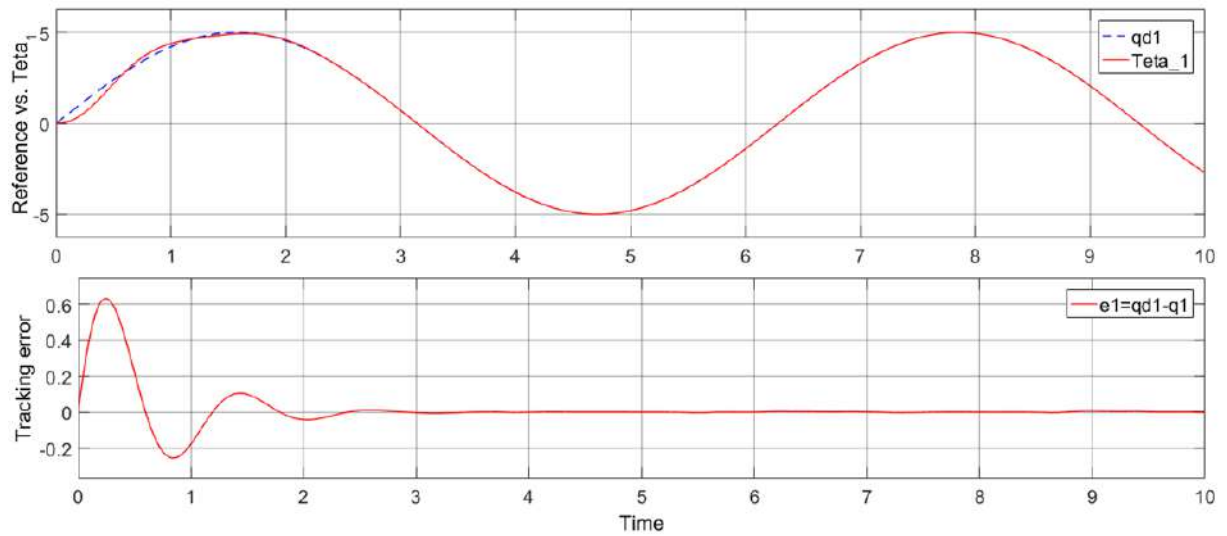
**II.5. Simulations**

Nous présenterons dans cette section les résultats de simulation de notre système en utilisant MATLAB-Simulink.

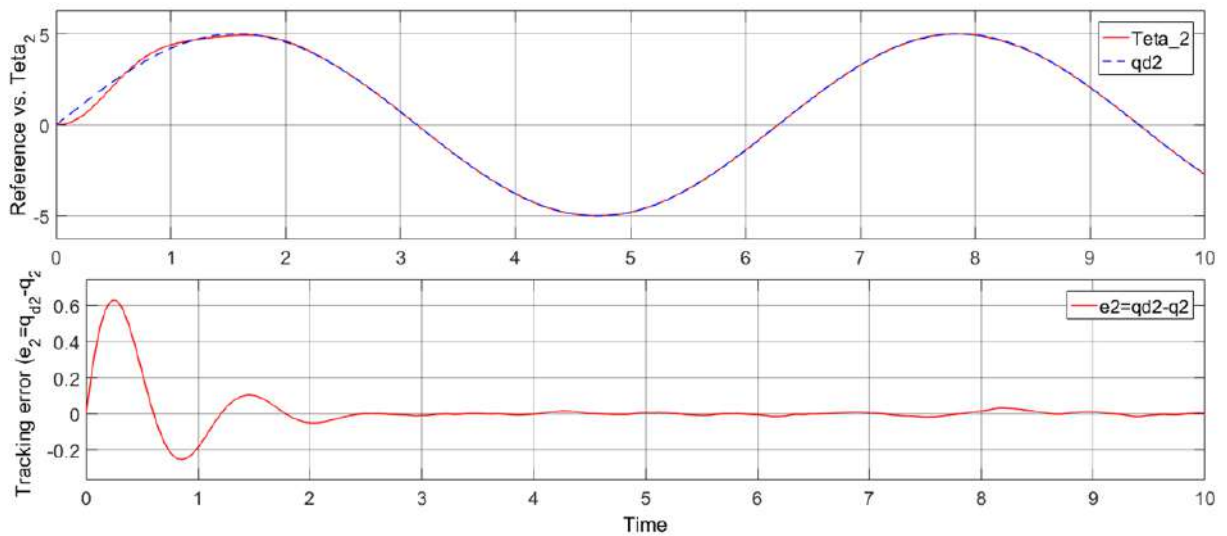
On choisit une consigne sinusoïdale donné par la forme :

$$q_{1\_d}(t) = 5*\sin(t) ; q_{2\_d}(t) = 5*\sin(t)$$

Ci-dessous les résultats de simulations avec la commande avec calcul de couple sans perturbations externes et pour  $K_{p1} = K_{p2} = 30$  et  $K_{d1} = K_{d2} = 5$ .



**Figure II.3 - Commande par calcul de couple: Teta\_1 et l'erreur e\_1**



**Figure II.4 - Commande par calcul de couple: Teta\_2 et l'erreur e\_2**

**Commentaire :** On remarque que la sortie  $\theta_1$  suit sa référence  $\theta_{1d}$  et l'erreur de poursuite commence important puis s'affaiblie pour avoisiner pratiquement le zéro. Mêmes remarques pour la sortie  $\theta_2$ .

On introduit des perturbations à l'entrée de la commande (couplets) de la forme suivante :

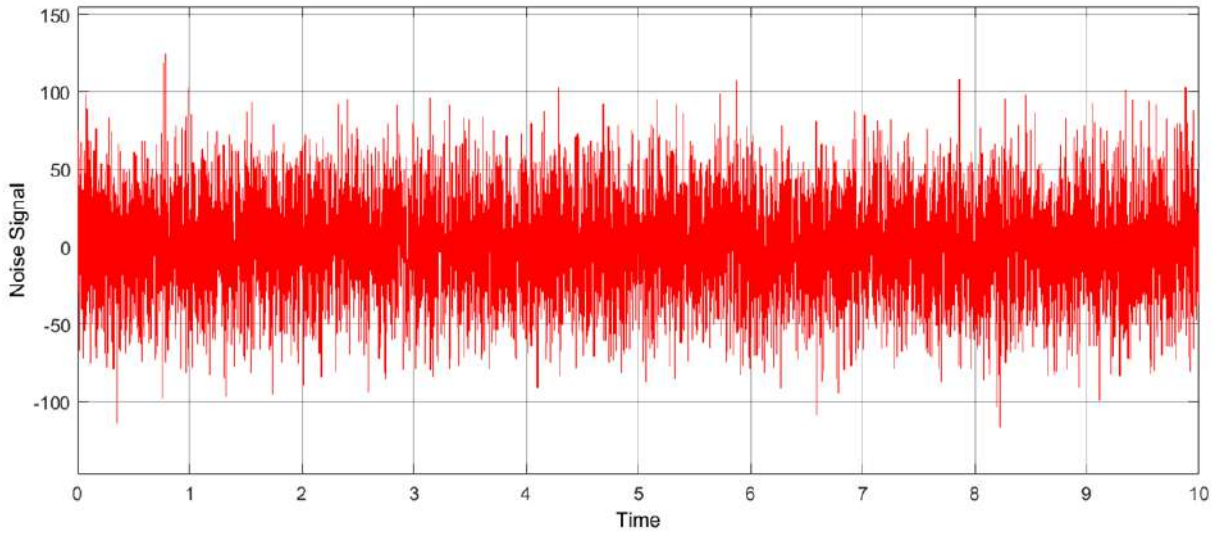


Figure II.5 - Forme des perturbations externes au niveau des couples de commande

Les résultats de simulation sont illustrés sur les deux figures suivantes :

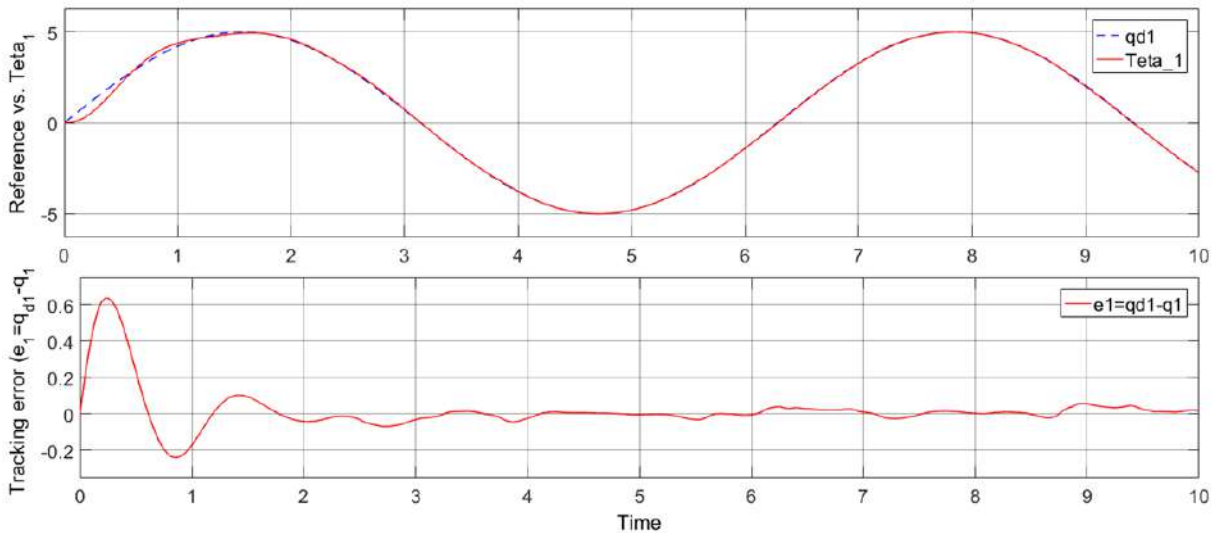


Figure II.6 - Commande par calcul de couple: Teta\_1 et l'erreur e\_1 avec perturbations



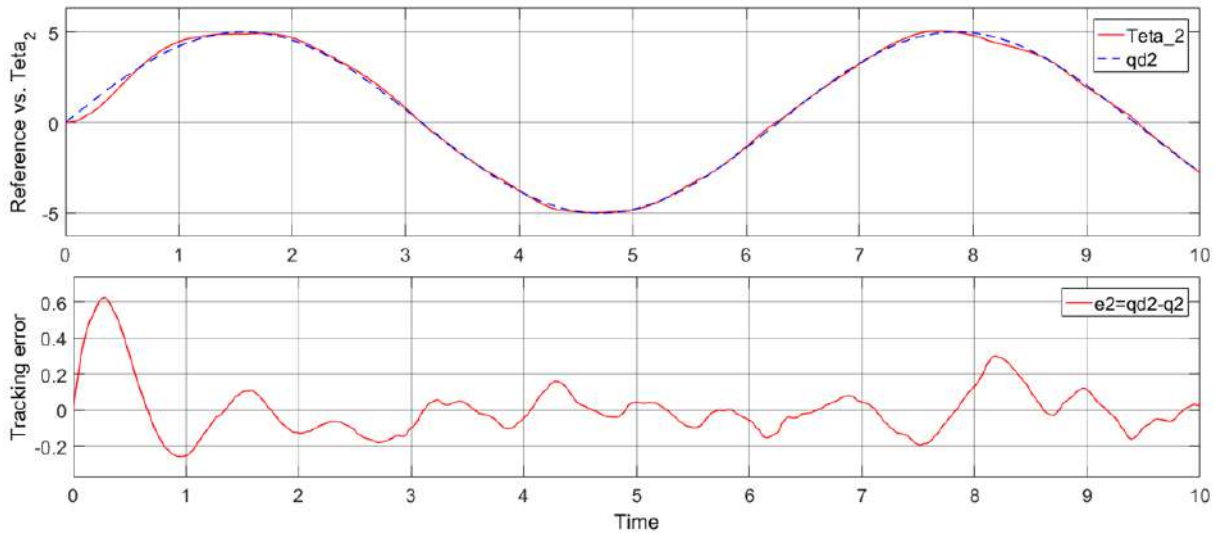


Figure II.7 - Commande par calcul de couple:  $\theta_2$  et l'erreur  $e_2$  avec perturbations

**Commentaire :** Avec l'introduction des perturbations à l'entrée de la commande, les performances de la commande par calcul de couple reste très excellentes ; on remarque l'apparition des petites fluctuations au niveau des oscillations des erreurs de poursuite mais ce sont assez négligeables. On peut voir également que la sortie  $\theta_1$  est moins touchée par ces perturbations que  $\theta_2$ .

Pour améliorer la forme de l'erreur de poursuite (pour certaines applications sensibles), on doit augmenter l'ordre de grandeur des gains PD. Les deux figures suivantes montre les réponses avec des gains plus important :  $K_{p1} = K_{p2} = 100$  et  $K_{d1} = K_{d2} = 50$ .

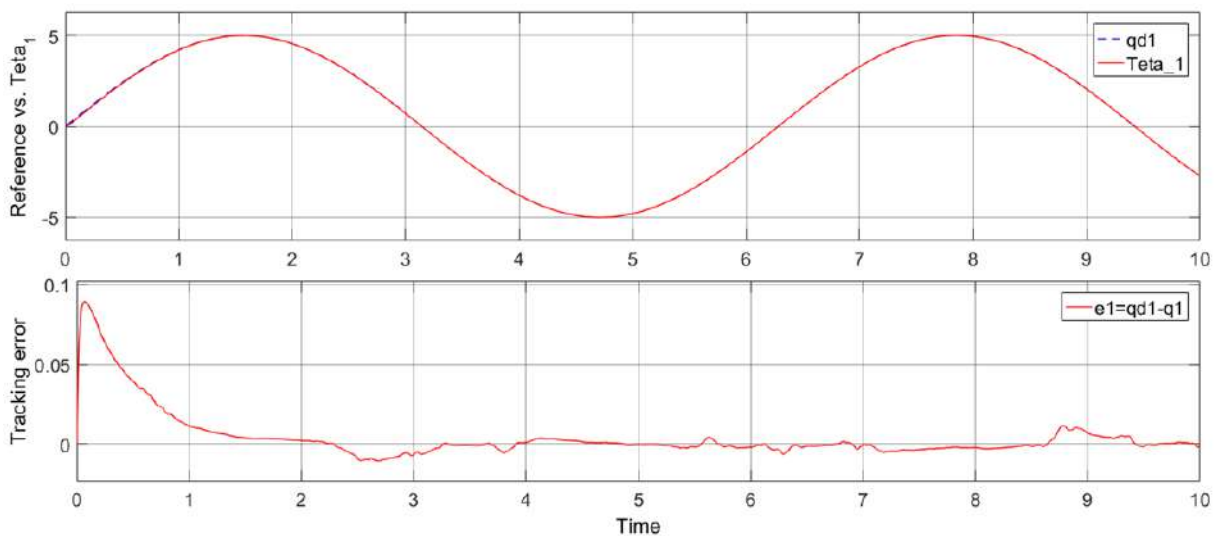
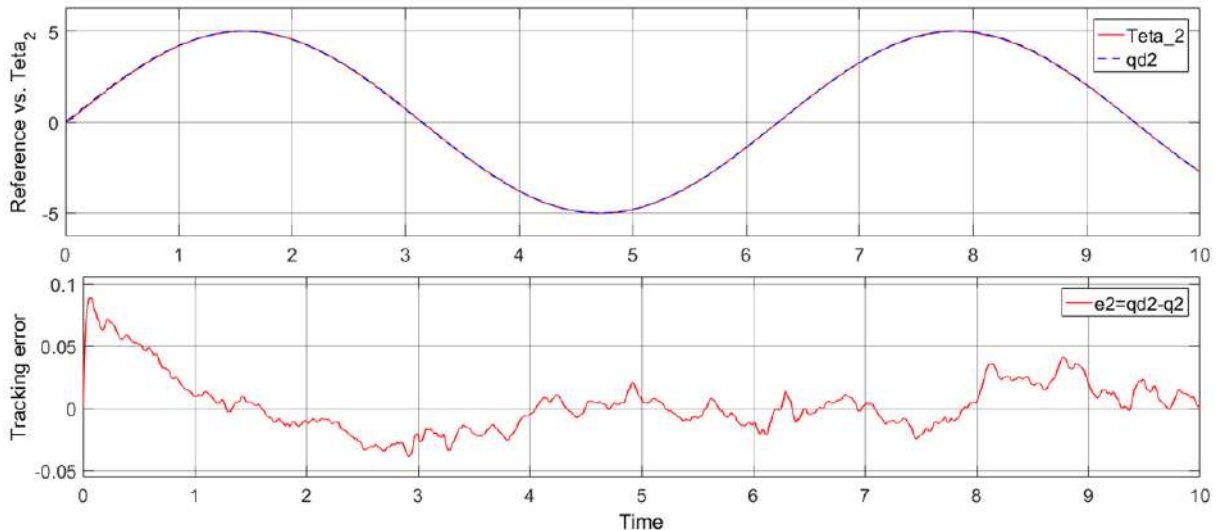


Figure II.8 - Commande par calcul de couple:  $\theta_1$  et l'erreur  $e_1$  avec perturbations et gains PD importants



**Figure II.9 - Commande par calcul de couple: Teta\_2 et l'erreur e\_2 avec perturbations et gains PD importants**

**Commentaire :** En imposant des gains plus importants l'erreur permanente oscillant encore plus étroitement entre +/- 0.05 pour les deux sorties.

## II.6. Conclusion

D'après les résultats de simulation, on remarque que, après une phase transitoire, les angles des articulations  $\theta_1$  et  $\theta_2$  suivent leurs références avec une erreur permanente oscillante autour de très faibles valeurs. Cette erreur est toujours appréciée et modérée par rapport aux exigences de l'application dédiée au bras manipulateur. Cette modération sera opérée par le choix des gains PD tel que autant ces derniers sont grands l'erreur devienne plus petite.

Ainsi la commande par calcul de couple s'avère très performante dans ce cas ; Cependant ses performances restent toujours liées à la fiabilité du modèle mathématique établi du robot comme tous les autres types de commandes basées modèle, elles donnent de bons résultats tant que les paramètres du bras sont exactement connus.

*Chapitre III :*  
*Commande Adaptative*  
*Floue à Mode Glissant*

## **Chapitre III - Commande Adaptative Floue A Mode Glissant**

### **III.1. Introduction**

Dans la commande des systèmes non linéaires ou ayant des paramètres non constants, les lois de commande classique peuvent être insuffisantes car elles sont non robustes surtout lorsque les exigences sur la précision et autres caractéristiques dynamiques des systèmes sont strictes. On doit faire appel à des lois de commande insensibles aux variations des paramètres, aux perturbations et aux non linéarités.

Dans cet objectif, plusieurs outils sont proposés dans la littérature, dont on cite la commande à structure variable (CSV) et la logique floue. La commande par mode de glissement fait partie de la famille des contrôleurs à structure variable, c.à.d. des commandes commutant entre plusieurs lois de commande différentes. L'importance des contrôleurs par mode glissant réside dans : la grande précision, la réponse dynamique rapide, la stabilité, la simplicité de la conception et l'implantation, et la robustesse vis-à-vis la variation des paramètres internes ou externes (DEGHBOUDJ, 2013).

La commande par mode glissant a largement prouvé son efficacité à travers plusieurs études théoriques, ses principaux domaines d'application sont la robotique et la commande des moteurs électriques. L'avantage que procure une telle commande, qui la rend aussi importante, est sa robustesse vis-à-vis des perturbations et des incertitudes du modèle. Cependant, ces performances sont obtenues au prix de certains inconvénients : l'apparition du phénomène de chattering ou broutement provoqué par la partie discontinue de la commande qui peut avoir un effet néfaste sur les actionneurs; le système est soumis à chaque instant à une commande élevée afin d'assurer sa convergence vers l'état désiré et ceci n'est pas souhaitable.

Parmi les solutions proposées à ces problèmes on peut citer la commande par mode glissant à bande limite qui consiste à remplacer la fonction de commutation par une fonction de saturation. Mais, cette solution n'est qu'un cas particulier de la commande par mode glissant flou, d'où l'intérêt à utiliser une commande qui combine la logique floue et le mode glissant avec gain adaptatif afin d'obtenir une commande robuste et lisse.

Certains chercheurs ont essayé de trouver une similitude entre la commande floue non linéaire et la commande par mode glissant et de combiner les deux lois de commande, c'est-à-dire la commande floue pour sa rapidité et sa facilité de mise en œuvre, et le régime glissant pour ses fondements théoriques rassurant du point de vue stabilité et robustesse. Cette combinaison reliant les avantages d'invariance par rapport aux incertitudes et aux perturbations de la commande à régime glissant avec ceux de la rapidité et de la bonne poursuite de trajectoire de la commande floue, permet de s'affranchir des problèmes de broutement de la commande par mode glissant et de déficit en outils d'analyse de la commande à base de la logique floue sans oublier la diminution du nombre de règles floues.

### III.2. Synthèse de la commande robuste par mode glissant

Considérons un système non linéaire décrit par la représentation d'état suivante :

$$\begin{cases} \dot{x}(t)^{(n)} = f(X, t) + b(X, t)u(t) + d(t) \\ y(t) = x(t) \end{cases} \quad \text{III.1}$$

avec :

$X(t) = [x_1(t) \ x_2(t) \ x_3(t) \ \dots \ x_n(t)]^T = [x(t) \ \dot{x}(t) \ \ddot{x}(t) \ \dots \ x^{(n-1)}(t)]^T$  est le vecteur d'état  
 $f(X, t)$  et  $b(X, t)$  sont des fonctions non linéaires du vecteur d'état avec  $b(X, t) \geq b > 0$ .

$u(t)$  : la commande.

$d(t)$  : la perturbation considérée comme étant bornée :  $d(t) < D$ .

Le but de la commande est de trouver une loi de commande telle que, les trajectoires du vecteur d'état tendent vers zéro malgré la présence des perturbations.

La mise en œuvre d'une commande par mode glissant passe par trois étapes :

- Choix de la surface de glissement ;
- Détermination des conditions d'existence du régime glissant ou conditions d'accès ;
- Synthèse des lois de commande du mode glissant.

#### III.2.1. Le choix de la surface de glissement

Le choix de la surface de glissement concerne non seulement le nombre nécessaire de ces surfaces, mais également leurs formes en fonction de l'application et de l'objectif visé. Le procédé le plus judicieux et le plus simple consiste à choisir une surface de commutation égale au vecteur d'erreur du vecteur d'état.

Généralement, le nombre des surfaces de glissement est choisi égal à la dimension du vecteur de commande. La surface de glissement est une fonction scalaire telle que la variable à régler glisse sur cette surface et tend vers l'origine du plan de phase. Plusieurs formes de la surface de glissement ont été proposées dans la littérature, chacune présente des meilleures performances pour une application donnée. La surface la plus utilisée pour obtenir le régime de glissement qui garantit la convergence de l'état vers sa référence est définie par (ABDELLI, 2015) :

$$S(X) = \left( \frac{\partial}{\partial t} + \lambda \right)^{r-1} e(X) \quad \text{III.2}$$

Où :

$r$  : est le degré relatif au nombre qu'il faut dériver la sortie pour faire apparaître la commande, pour notre cas  $r=n$  ;

$\lambda$  : est une matrice diagonale à constante positive.

$e$  : erreur de poursuite donné par :

$$e = x_d - x \quad \text{III.3}$$

### III.2.2. La Condition d'accès au mode glissant

Cette condition est en fait la condition sous laquelle le mode de glissement existe et sous laquelle la trajectoire d'état va effectivement atteindre la surface de glissement en un temps fini.

Afin de déterminer ces conditions d'existence et de convergence du régime glissant, deux approches ont été proposées :

- L'approche directe qui est la plus ancienne, a été proposée et étudiée par Emilyanov et Utkin (ABDELLI, 2015). Elle est globale mais ne garantit pas, en revanche un temps d'accès fini. Elle est exprimée par :

$$S(X)\dot{S}(X) < 0 \quad \text{III.4}$$

- L'approche de Lyapunov qui est une condition globale d'accès au mode glissant. Il s'agit de choisir une fonction scalaire positive appelée fonction de Lyapunov qui peut être donnée par :

$$V(X) = \frac{1}{2} S^2(X) \quad \text{III.5}$$

L'idée est de choisir cette fonction pour garantir l'attraction de la variable à contrôler vers sa valeur de référence, et de concevoir une commande  $u$  telle que le carré de la surface correspond à une fonction de Lyapunov. La dérivée de la fonction  $V(x)$  donne :

$$\dot{V}(X) = S(X)\dot{S}(X) \quad \text{III.6}$$

Pour que la fonction candidate de Lyapunov puisse décroître et converger vers zéro (pour garantir l'existence du mode de glissant), il suffit d'assurer que:

$$S(X)\dot{S}(X) < 0 \quad \text{III.7}$$

L'équation (III.5) signifie que le carré de la distance vers la surface mesuré par  $S^2(X)$  diminue tout le temps, contraignant la trajectoire du système à se diriger vers la surface de glissement dans les deux côtés. Cette condition suppose un régime glissant idéal où la fréquence de commutation est infinie.

### III.2.3. Synthèse des lois de commande du mode glissant

Afin d'assurer l'attractivité et l'invariance de la surface de glissement, on doit définir une loi de commande sous la forme suivante :

$$U = U_{eq} + U_n \quad \text{III.8}$$

Telle que  $U_{eq}$  est la commande équivalente définie par Utkin (ABDELLI, 2015). Elle sert à maintenir la variable à contrôler sur la surface de glissement (propriété d'invariance). La commande équivalente est exprimée, en considérant que la dérivée de la surface est nulle, c'est-à-dire  $\dot{S}(X) = 0$ . En effet, on peut interpréter la grandeur de la commande équivalente comme étant la valeur moyenne de la commande lors de la commutation rapide entre ces deux valeurs  $U_{max}$  et  $U_{min}$ .

Tandis que  $U_n$  est la commande discrète qui est déterminée afin de vérifier la condition de convergence en dépit de l'imprécision sur les paramètres du modèle du système. Quand le système défini par l'équation (III.1) fonctionne en régime glissant, sa dynamique vérifie la condition  $S(X) = 0$ .

La dérivée de la surface de glissement est donnée par (pour notre cas  $r=n=2$ ):

$$\dot{S} = \ddot{e} + \lambda \cdot \dot{e} \quad \text{III.9}$$

En développant on obtient :

$$\dot{S} = \ddot{X}_d - \ddot{X} + \lambda \dot{e} \quad \text{III.10}$$

En remplaçant l'expression de  $\ddot{X}$  donné dans (III.1) pour  $n=2$ , on aura :

$$\dot{S} = \ddot{X}_d - f(X, t) - b(X, t)u(t) - d(t) + \lambda \dot{e} \quad \text{0.11}$$

Lorsque le mode glissant est atteint et en régime permanent, la surface du glissement est nulle, et par conséquent, sa dérivée et la partie discontinue sont aussi nulles. D'où, on déduit l'expression de la commande équivalente :

$$U_{eq} = b^{-1}(X, t)[\ddot{X}_d + \lambda \dot{e} - f(X, t) - d(t)] \quad \text{0.12}$$

Pour assurer l'attractivité de la surface de glissement, il suffit d'ajouter le terme  $U_n$  à la loi de commande, de telle sorte que :

$$S(X)\dot{S}(X) < 0 \quad \text{III.13}$$

Pour vérifier cette condition, une solution simple est proposée pour  $U_n$  :

$$U_n = K \cdot \text{sign}(S) \quad \text{III.14}$$

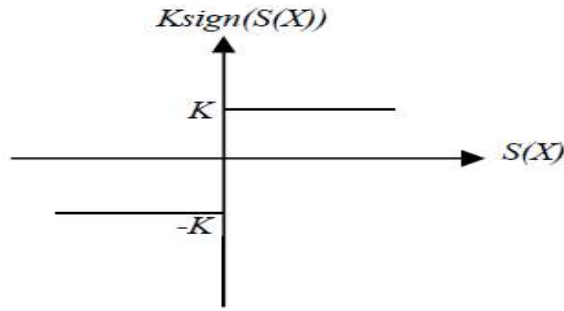
Où  $K$  est une matrice diagonale à coefficients constants positive et la fonction scalaire  $\text{sign}$  est définie comme suite :

$$\text{sign}(S) = \begin{cases} 1 & \text{si } S > 0 \\ 0 & \text{si } S = 0 \\ -1 & \text{si } S < 0 \end{cases} \quad \text{III.15}$$

Le choix de ce gain est très influant car s'il est très petit, le temps de réponse sera très long, et s'il est choisi très grand, le temps de réponse sera très petit.

La figue suivante représente la fonction de la commande discrète de type relais :



Figure 0.1 - Fonction  $sign()$  de la commande discrète

### III.3. Application au robot à deux degré de liberté

Le modèle dynamique indirecte du robot à deux degré de liberté est donné par :

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d \quad \text{III.16}$$

Alors que le modèle dynamique directe est donné par :

$$\ddot{q} = M^{-1}(q)(\tau - C(q, \dot{q})\dot{q} - F(\dot{q}) - G(q) - \tau_d) \quad \text{III.17}$$

Ou :

$$\ddot{q} = M^{-1}(q)(-C(q, \dot{q})\dot{q} - F(\dot{q}) - G(q) - \tau_d) + M^{-1}(q)\tau \quad \text{III.18}$$

Et on peut écrire :

$$\ddot{q} = f(q, \dot{q}) + g(q)u + w(q) \quad \text{III.19}$$

Avec :

$$f(q, \dot{q}) = -M^{-1}(q)(C(q, \dot{q})\dot{q} + F(\dot{q}) + G(q)) \quad \text{III.20}$$

$$g(q) = M^{-1}(q) \text{ et } u = \tau \quad \text{III.21}$$

$$w(q) = -M^{-1}(q)(\tau_d) \quad \text{III.22}$$

La surface du glissement est définie par :

$$S(X) = \left(\frac{\partial}{\partial t} + \lambda\right)^{r-1} e(X) \quad \text{III.23}$$

Pour  $r=n=2$  on aura :

$$S(q) = \dot{e} + \lambda e \quad \text{III.24}$$

Avec :  $e = q_d - q$  ou :  $q_d$  est la trajectoire désirée et  $q = [\theta_1 \ \theta_2]^T$ .

La dérivée de la surface de glissement sera donnée par :

$$\dot{S} = \ddot{e} + \lambda \dot{e} = (\ddot{q}_d - \ddot{q}) + \lambda \dot{e} \quad \text{III.25}$$

En substituant  $\ddot{q}$  par l'équation de la dynamique directe du robot on obtient :

$$\dot{S} = (\ddot{q}_d - f(q, \dot{q}) - g(q)u - w(q)) + \lambda \dot{e} \quad \text{III.26}$$

On remplace la commande  $u$  par ses deux termes équivalent et discontinu :

$$\dot{S} = (\ddot{q}_d - f(q, \dot{q}) - g(q)(U_{eq} + U_n) - w(q)) + \lambda \dot{e} \quad \text{III.27}$$

Lorsque le mode glissant est atteint et en régime permanent, la surface de glissement est nulle, et par conséquent, sa dérivée et la partie discontinue sont aussi nulles, l'équation (III.27) devient :

$$(\ddot{q}_d + \lambda \dot{e}) - f(q, \dot{q}) - g(q)(U_{eq}) - w(q) = 0 \quad \text{III.28}$$

Enfin on déduit la loi de commande équivalente comme suite :

$$U_{eq} = g(q)^{-1}((\ddot{q}_d + \lambda \dot{e}) - f(q, \dot{q}) - w(q)) \quad \text{III.29}$$

Et comme  $g(q) = M^{-1}(q)$  alors  $g(q)^{-1} = M(q)$  ; on obtient :

$$U_{eq} = M(q)(\ddot{q}_d + \lambda \dot{e}) - M(q)f(q, \dot{q}) - M(q)w(q) \quad \text{III.30}$$

Ce qui donne :

$$U_{eq} = M(q)(\ddot{q}_d + \lambda \dot{e}) + C(q, \dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d \quad \text{III.31}$$

Or :  $(\dot{q}_d + \lambda e) = \dot{q}$

Alors finalement on déduit que :

$$U_{eq} = M(q)(\ddot{q}_d + \lambda \dot{e}) + C(q, \dot{q})(\dot{q}_d + \lambda e) + F(\dot{q}) + G(q) + \tau_d \quad \text{III.32}$$

Pour assurer l'attractivité de la surface de glissement, il suffit d'ajouter le terme  $U_n$  à la loi de commande avec :

$$U_n = K \cdot \text{sat}(S)$$

La fonction  $\text{sat}()$  est telle donné par le figure suivante :

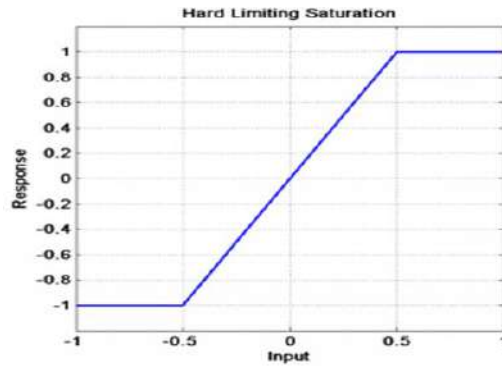


Figure III.2 - Fonction Saturation  $sat()$

### III.4. Simulations

Les résultats de simulations de la commande à mode glissant sus définie sont illustrés sur les deux figures ci-dessous, pour  $\lambda_1=\lambda_2=10$  et  $K_{v1}=K_{v2}=20$  :

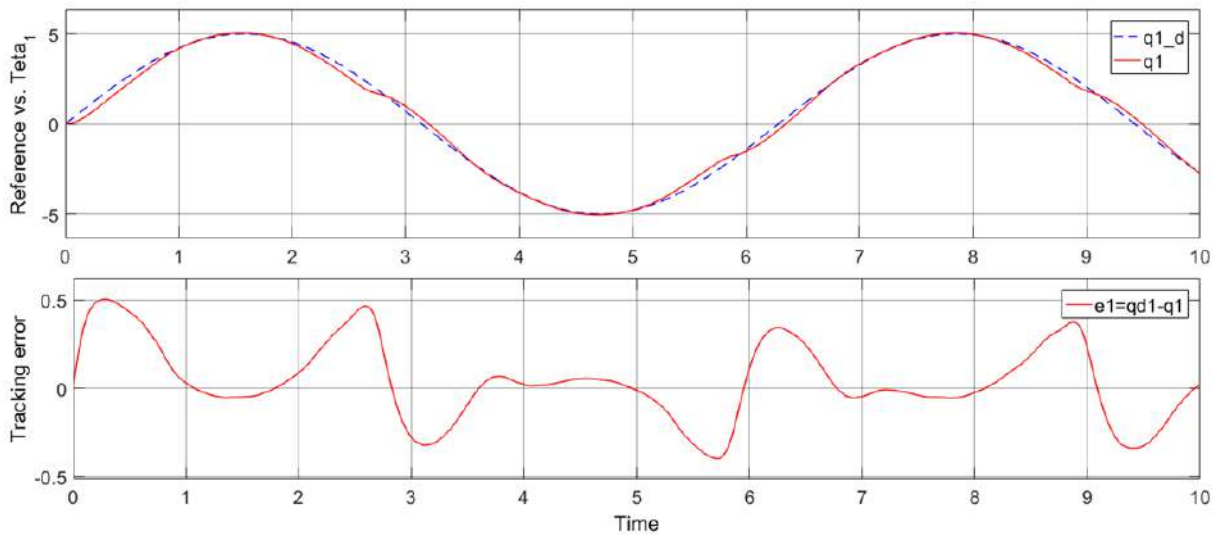


Figure III.3 - Simulation Teta\_1 et l'erreur e1

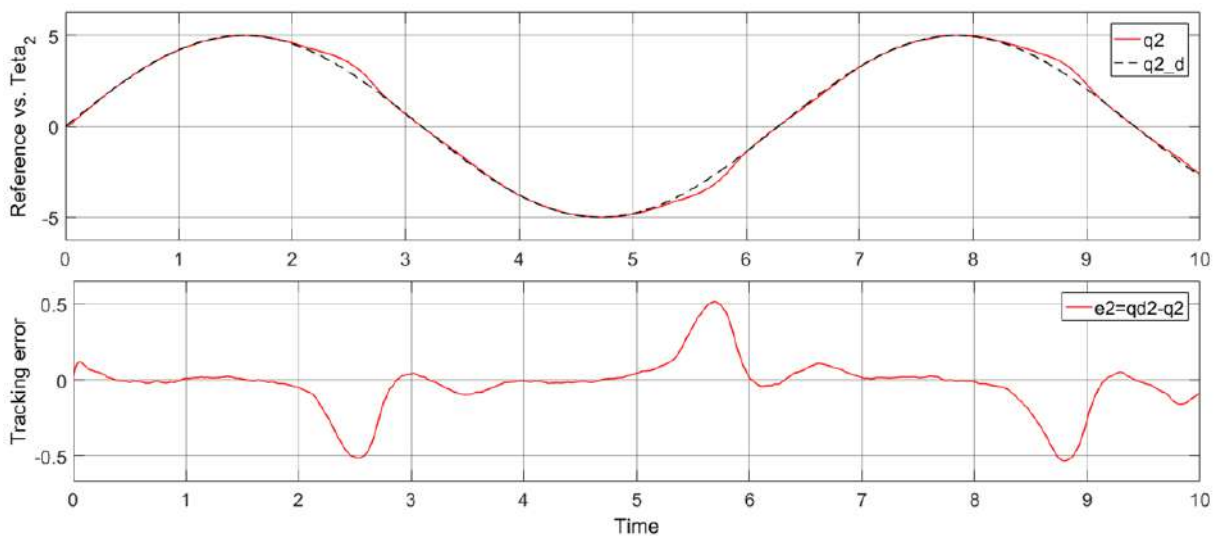


Figure III.4- Simulation Teta\_2 et l'erreur e2

**Commentaire :** Les sorties de notre système (angles des articulations  $\theta_1$  et  $\theta_2$ ) suivent leurs références avec des erreurs de poursuite qui varient (en régime permanent) entre +/-0.5 pour  $\theta_1$  et  $\theta_2$ ; ces résultats sont très satisfaisantes et montre la performance de la commande à mode glissant en présence des perturbations externes.

### III.5. Conclusion :

La commande à mode glissant a prouvé sa robustesse et sa fiabilité dans la commande du robot manipulateur à deux degrés de liberté. C'est en jouant sur les paramètres de la commande (à savoir  $\lambda_i$  et  $K_i$  qui devrait être choisi suffisamment grand) pour assurer les performances désirées du système en boucle fermé.

C'est clair que pour calculer la commande à mode glissant il est nécessaire de calculer au préalable les termes  $M(q)$ ,  $C(q, \dot{q})$ ,  $G(q)$  qui dépendent complètement des paramètres du notre robot manipulateur. Hors ses fonctions sont parfois pas parfaitement définies et contiennent des incertitudes et parfois ce ne sont pas de tout connus.

Dans ce cas on fera appelle à la logique floue pour approximer ces fonctions en temps réels. Le système flou de Takagi-Sugeno (T-S) peut approximer efficacement tout système non linéaire et pour notre cas on va l'utiliser pour approximer la dynamique non linéaire fortement couplée de notre robot afin de calculer la commande le terme équivalent de la commande à mode glissant.

### **III.6. Synthèse d'un système flou pour calculer la commande à mode glissant**

#### **III.6.1. Rappelle théorique sur la logique floue**

##### **III.6.1.1. Introduction**

En automatique, la majorité des approches de la commande non linéaire exige la disponibilité d'un modèle mathématique du système, et ceci n'est pas toujours réalisable à cause de l'imprécision et l'incertitude liées aux paramètres mal connus, difficilement identifiables et des dynamiques négligées. D'autre part, les performances assurées seront directement liées à l'exactitude du modèle utilisé. Par conséquent, les automaticiens sont confrontés au problème de définir un modèle mathématique précis sachant que les systèmes deviennent de plus en plus complexes, les méthodes de modélisation traditionnelles s'avèrent souvent incapables de représenter le comportement global d'un système.

L'utilisation des contrôleurs basés sur l'expertise humaine peut être une alternative à la commande de ce type de systèmes. Parmi ces approches, nous distinguons celles utilisant la commande par logique floue.

Dans cette partie de ce travail, nous introduisons les concepts de base d'un système flou type-1, ainsi que la conception des contrôleurs flous type-1.

##### **III.6.1.2. Principes et définitions de la logique floue**

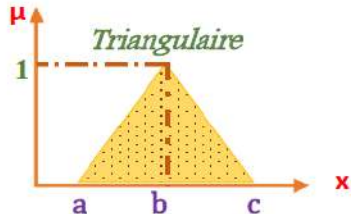
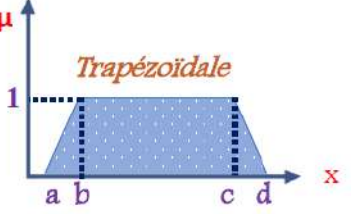
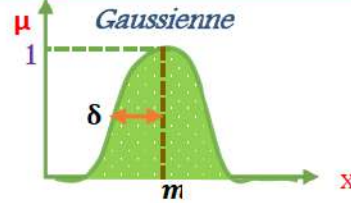
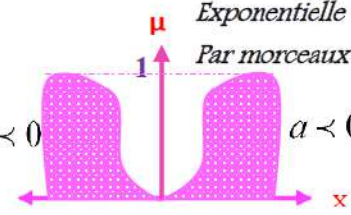
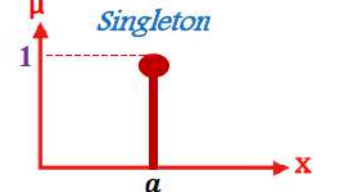
La logique floue (Fuzzy Logic) propose une approche des problèmes beaucoup plus pragmatique que mathématique, dans certains problèmes où les mathématiques peinent par impossibilité ou difficulté de modélisation, la logique floue apporte une étonnante efficacité.

La logique floue est très proche du processus de la pensée humaine. Elle met en œuvre un jeu de règles comme, implicitement, nous en utilisons chaque jour. Elle apprécie les variables d'entrées de façon approximative (faible, élevée, loin, proche), fait de mêmes pour les variables de sorties (freinage léger ou fort) et édicte un ensemble de règles permettant de déterminer les sorties en fonction des entrées. Dans la logique classique, les variables gérées sont Booléennes. C'est à dire qu'elles ne prennent que deux valeurs 0 ou 1. La logique floue a pour but de raisonner à partir de connaissances imparfaites qui opposent résistance à la logique classique. Pour cela la logique floue se propose de remplacer les variables booléennes par des variables flous.

**III.6.1.3. Fonction d'appartenance type-1**

Une définition des variables linguistiques type-1 à l'aide des fonctions d'appartenance type-1 est nécessaire dans le but de traiter des déductions floues par ordinateur. Dans ce contexte, est attribuée à chaque valeur de la variable linguistique type-1 une fonction d'appartenance type-1, dont la valeur varie entre 0 et 1. Le plus souvent, nous utilisons des fonctions d'appartenance de type triangulaire ou trapézoïdale gaussienne.

Tableau III-1 - Fonctions d'appartenance

<i>Fonction</i>	<i>La Forme Algébrique</i>	<i>Forme</i>
<p>✚ <u>Triangulaire</u></p>	$u(x) = \max(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0)$	
<p>✚ <u>Trapézoïdale</u></p>	$u(x) = \max(\min\left(\frac{x-a}{b-a}, \frac{d-x}{d-c}\right), 0)$	
<p>✚ <u>Gaussienne</u></p>	$u(x) = \exp\left(-\frac{(x-m)^2}{2\delta^2}\right)$	
<p>✚ <u>Sigmoïdale</u></p>	$u(x) = \frac{1}{1 + \exp(-a(x-c))}$	
<p>✚ <u>Singleton</u></p>	$\mu_A(x) = \begin{cases} m & \text{pour } x = a \\ 0 & \text{pour } x \neq a \end{cases}$	

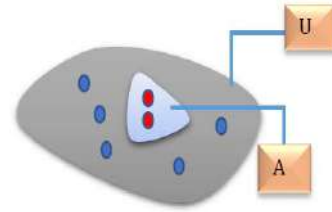
### III.6.1.4. Univers de discours

L'univers de discours  $U$  représente l'ensemble de référence ou le domaine de la variation de la variable linguistique.

### III.6.1.5. Sous-Ensembles flous type-1

Soient  $U$  l'univers du discours.

$A$  : un sous-ensemble de  $U$



**Définition** : Un sous ensemble flou  $A$  de  $U$  est défini par une fonction d'appartenance qui associe à chaque élément de  $U$  le degré  $(x)$ , compris entre 0 et 1 :  $A = \sum_{u_i \in U} \mu(u_i) / u_i$

### III.6.1.6. Opérations sur les ensembles flous type-1

Comme dans le cas des ensembles « classiques », les opérations logiques d'union (ou), d'intersection (et) et de complémentation (non) peuvent être appliquées aux ensembles flous type-1. Leur définition n'est pas unique. Les définitions les plus souvent rencontrées sont : le max et le min (Mamdani), le produit et la somme-produit (Sugeno) :

Mamdani:  $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$

$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$

Sugeno:  $\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - (\mu_A(x) \cdot \mu_B(x))$

$\mu_{A \cap B}(x) = (\mu_A(x) \cdot \mu_B(x))$

Dans les deux cas  $\overline{\mu_A(x)} = 1 - \mu_A(x)$  pour  $x \in U$

### III.6.1.7. Variable linguistique type-1

Une variable linguistique type-1 est définie par un triplé  $(V, U, T_v)$  où  $V$  représente une variable classique (vitesse, température,) définie sur l'univers de discours  $U$ .  $T_v$  désigne le sous ensemble flou repérés par leur label  $A_i$  : on écrit ainsi  $T_v = (A_1, A_2, A_3, \dots, A_N)$ .

Une variable linguistique peut être représentée graphiquement, comme dans la Figure I.5 qui représente une variable définie comme  $\{V, U, T_v = (A_1, A_2, A_3, A_4)\}$  .

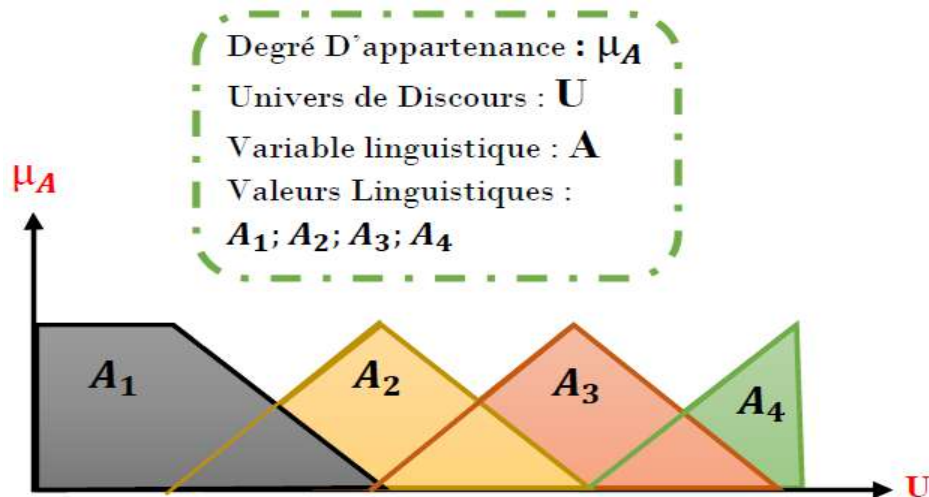


Figure I.5 - Variables linguistiques et Fonction d'appartenance

### III.6.1.8. Structure du réglage par logique floue

Le contrôle par logique floue est chargé de calculer à chaque instant la commande réelle  $R_s$  (non floue) à appliquer au système afin de réaliser l'objectif de contrôle qui est l'annulation de l'erreur de réglage  $e(k) = y_{ref}(k) - y(k)$  (la référence – la sortie mesurée).

Les entrées de réglage sont en général :

- $e$  : l'erreur, elle est définie par la différence entre la consigne et la grandeur à réguler.
- $de$  : la dérivée de l'erreur.

La sortie du régulateur est donnée par :  $R_s(k) = R_s(k-1) + dR_s(k)$

### III.6.1.9. Structure d'un régulateur Flou type-1

Un régulateur flou type-1 est composé de 3 blocs principaux (Figure I0.6)

- Interface de Fuzzification.
- Base de Connaissances et le Moteur D'inférence.
- Interface de Défuzzification.



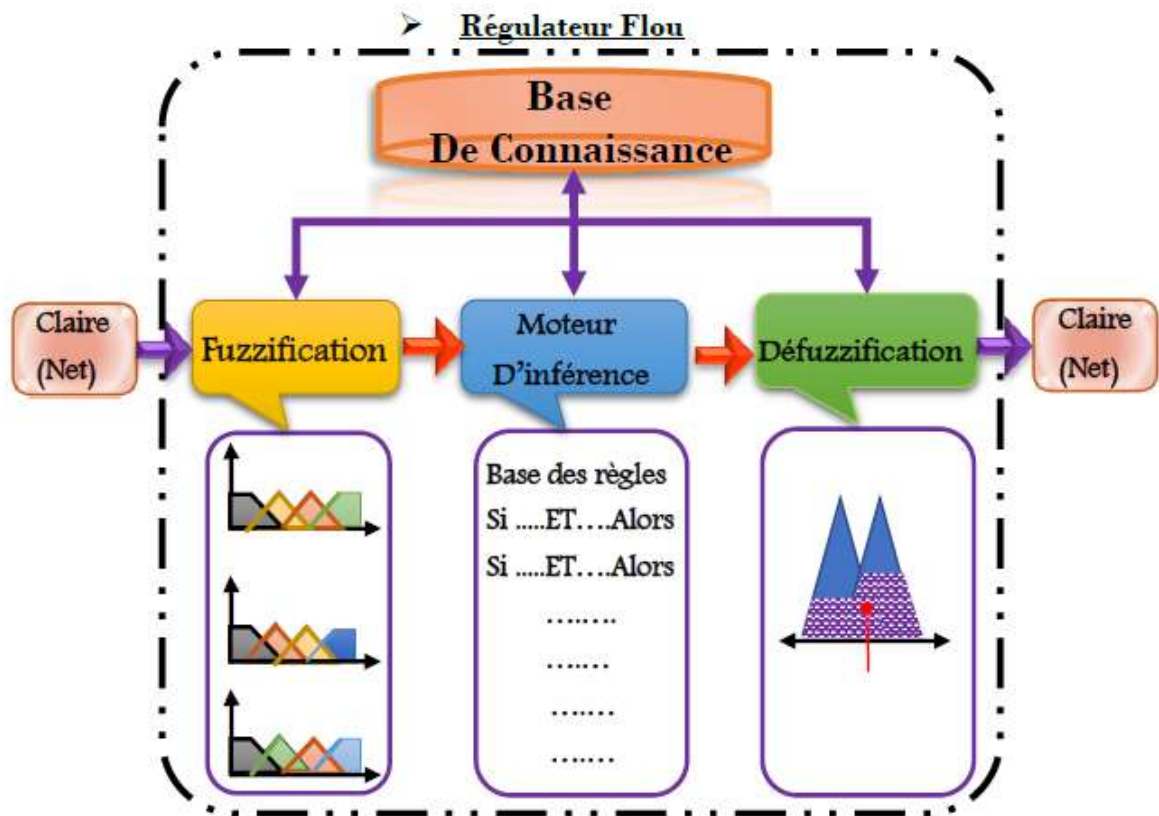


Figure I0.6 - Structure de base d'un régulateur flou

- **Fuzzification**

C'est l'opération de projection des variables physiques réelles sur des ensembles flous caractérisant les valeurs linguistiques prises par ces variables. Le choix de la forme des fonctions d'appartenance (triangulaires, trapézoïdales, exponentielles, gaussiennes,) est arbitraire.

Les formes triangulaires facilitent la programmation ce qui explique qu'elles soient le plus fréquemment utilisées. Quant au nombre de fonctions d'appartenance, il est généralement impair car elles se répartissent autour de zéro.

- **Base de connaissance**

La base de connaissance est composée de deux blocs :

- la base de données : qui fournit les définitions nécessaires utilisées pour définir les règles de contrôles linguistiques et la manipulation des données floues dans le contrôleur.
- la base de règles : qui caractérise les buts et stratégies de commande émis par les experts du domaine au moyen d'un ensemble de règles linguistiques du contrôle.

- **Table de décision:**

La table de décision contient l'ensemble des règles floues nécessaire dans la commande du système. Parmi les tables standards la table Mac Vicar - Whelan (Tableau I-2) qui peut être appliquée à plusieurs problèmes de commande floue.

Tableau I-2 - Table de décision de Mac Vicar-Whelan

<i>de</i> \ <i>e</i>	<u>NG</u>	<u>N</u>	<u>EZ</u>	<u>P</u>	<u>PG</u>
<u>NG</u>	<i>NG</i>	<i>NG</i>	<i>NG</i>	<i>N</i>	<i>EZ</i>
<u>N</u>	<i>NG</i>	<i>NG</i>	<i>N</i>	<i>EZ</i>	<i>P</i>
<u>EZ</u>	<i>NG</i>	<i>N</i>	<i>EZ</i>	<i>P</i>	<i>PG</i>
<u>P</u>	<i>N</i>	<i>EZ</i>	<i>P</i>	<i>PG</i>	<i>PG</i>
<u>PG</u>	<i>EZ</i>	<i>P</i>	<i>PG</i>	<i>PG</i>	<i>PG</i>

- **Moteur d'inférence**

C'est le 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 Max-Min (Mamdani)
- Méthode Max-Produit (Larsan)
- Méthode Somme-Produit (Sugeno). la plus utilisée car son temps de calcul est court

- **Défuzzification :**

La défuzzification consiste à transformer l'ensemble flou résultant de l'agrégation des règles en une grandeur de commande précise. Là aussi il existe plusieurs méthodes, parmi lesquelles :

- La méthode de la hauteur
- La moyenne des maximas
- Le centre de gravité
- Le premier des maximas
- Le centre des maximas
- Le centre des aires
- Le dernier des maximas
- Le centre de la plus grande surface

Tableau I0-3 - Méthodes de défuzzification

Méthode	Formule	Schéma Significatif
<ul style="list-style-type: none"> <li>• <i>La moyenne des maximas</i></li> </ul>	$u = \frac{\sum_{i=1}^n h_A(x_i) * x_i}{\sum_{i=1}^n h_A(x_i)}$	
<ul style="list-style-type: none"> <li>• <i>Centre de gravité</i></li> </ul>	$u = \frac{\sum_{i=1}^n x_i * v_i}{\sum_{i=1}^n v_i}$	

III.6.1.10. Types de régulateurs flous

- **Régulateur flou de type Mamdani :**

Dans ce type, les conclusions sont symboliques ou linguistiques et le calcul de la sortie nécessite l'utilisation d'une méthode de défuzzification.

- **Régulateur flou de type Sugeno :**

Ce modèle de Takagi-Sugeno ou T-S devient la technique la plus utilisée car sa simplification de représenter n'importe quel système non-linéaire complexe avec des règles floues réduite par rapport au type de Mamdani, et aussi la conséquence qui est une fonction numérique.

Un modèle Takagi-Sugeno flou est défini par la notion des règles floues sous la forme suivante :

**Si**  $z_1(t)$  est  $M_{i1}$  et ... et  $z_p(t)$  est  $M_{ip}$  **Alors**  $y_i = a_{0i} + a_{1i} z_1(t) + a_{2i} z_2(t) + \dots + a_{pi} z_p(t)$

Alors que  $M_{i1}, \dots, M_{ip}$  sont les ensembles flous,  $a_{0i} \dots a_{pi}$  sont les coefficients de  $i^{\text{ème}}$  conséquence linéaire et aussi c'est la sortie de  $i^{\text{ème}}$  valeur numérique de la sortie

Pour le modèle flou de T-S la sortie est calculée comme suit :

$$y = \frac{\sum_{i=1}^r w_i(t) y_i}{\sum_{i=1}^r w_i(t)} \tag{III.33}$$

Avec,  $0 < w_i(t) < 1$ ,  $\sum_{i=1}^r w_i(t) > 0$  et  $w_i(t) = \prod_{j=1}^p M_{ij}(z_j(t))$  avec  $M_{ij}(z_j(t))$  est le degré de la fonction d'appartenance de  $z_j(t)$  en  $M_{ij}$ .

L'équation (III.33) donne un modèle T-S flou affine, pour  $a_{0i}=0$ , on obtient un modèle T-S linéaire. Le modèle T-S représente un système dynamique, avec des règles floues Si-Alors (If-Then) qui représente des relations localement linéaire entre l'entrée et la sortie du système dynamique non-linéaire. L'aspect principal du modèle T-S floue est de représenter chaque règle floue locale dynamique par un sous-modèle linéaire, après le système flou global est obtenu par le calcul de la somme floue des sous-modèles linéaires.

C'est par ce type de régulateur flou qu'on va estimer la partie de la commande équivalente de notre commande à mode glissant appliquée à un robot manipulateur à deux degrés de liberté.

### III.6.2. Synthèse d'un système adaptatif flou pour la commande SMC

#### III.6.3.1. Etape 1: Fuzzification

Les entrées de notre système flou seront l'erreur et sa dérivé telle que  $e = q_d - q$ . Pour passer des valeurs réelles aux variables floues on va définir trois sous-ensembles flous : Négatif (N), Zero (Z), Positif (P). Les fonctions d'appartenance à ces trois sous-ensembles sont des fonctions gaussiennes définies comme suit :

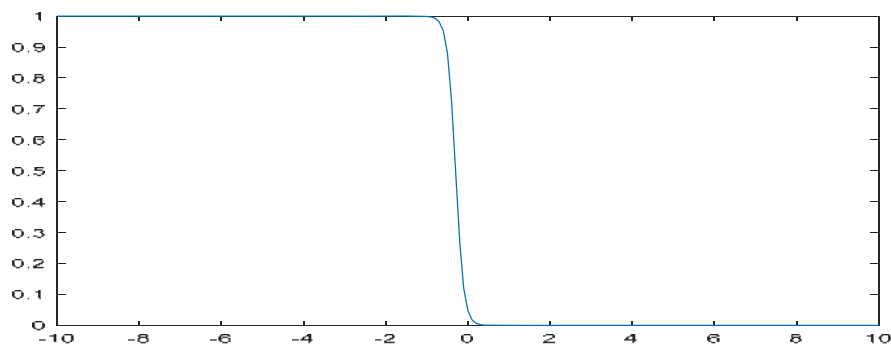


Figure 0.7 - Fonction d'appartenance au sous ensemble flou Négatif :  $A=-10$  et  $C=-0.3$

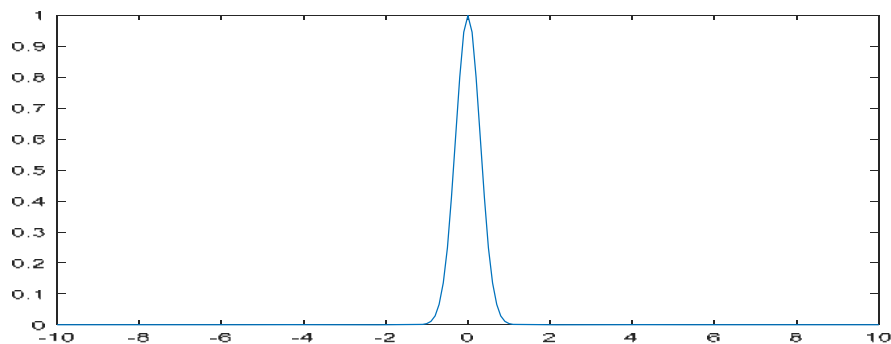


Figure 0.8 - Fonction d'appartenance au sous ensemble flou Zero :  $\delta=0.3$  et  $C=0$

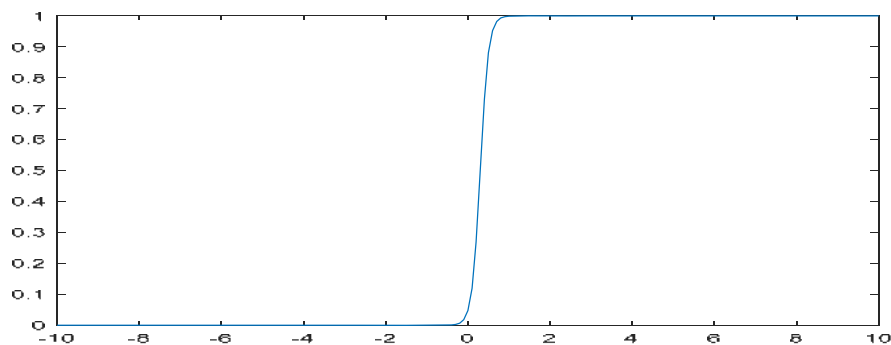


Figure 0.9 - Fonction d'appartenance au sous ensemble flou Positif :  $A=10$  et  $C=0.3$

Avec ce choix des sous-ensembles flous, on aura 9 règles constituant notre base de décision.

### III.6.3.2. Etape 2: Fonctions de sortie

En utilisant une stratégie d'inférence en produit et une défuzzification en moyen des maxima la fonction de sortie de notre système flou sera donné par :

$$\mathbf{y}(x) = \frac{\sum_{j=1}^m y^j (\prod_{i=1}^n \mu_{A_i^j}(x_i))}{\sum_{j=1}^m (\prod_{i=1}^n \mu_{A_i^j}(x_i))} \quad \text{III.34}$$

Avec  $\mu_{A_i^j}(x_i)$  est la fonction d'appartenance des variables linguistique d'entrée  $i=1,2$  (à savoir l'erreur et sa dérivé) ;  $y^j$  est le point dans R ou  $\mu_{B_i^j}$  atteint son maximum en assumant que  $\mu_{B_i^j}(y^j) = 1$ .

En introduisant le concept du vecteur de fonction base du système flou  $\xi(x)$ , l'équation (III.34) devient :

$$\mathbf{y}(x) = \boldsymbol{\theta}^T \boldsymbol{\xi}(x) \quad \text{III.35}$$

Avec :  $\boldsymbol{\theta} = (y^1, \dots, y^m)^T$  et  $\boldsymbol{\xi}(x) = [\xi^1(x), \dots, \xi^m(x)]$  ou  $\xi^j(x) = \frac{\prod_{i=1}^n \mu_{A_i^j}(x_i)}{\sum_{j=1}^m (\prod_{i=1}^n \mu_{A_i^j}(x_i))}$

Les fonctions  $\mu_{A_i^j}(x_i)$  sont celles définie dans l'étape 1.

### III.6.3.3. Etape 3: Loi d'adaptation des paramètres

Afin que le système flou approxime le système en temps réel (on-line) une loi d'adaptation des paramètres du système flou doit être considéré.

Une loi d'adaptation est proposée et la convergence et la stabilité asymptotique du système en boucle fermée est prouvé en (Wang, Rad, & Chan, 2001). Cette loi est donnée par :

$$\dot{\boldsymbol{\theta}} = \mathbf{r} \cdot \mathbf{s} \cdot \boldsymbol{\xi}(x) \quad \text{III.36}$$

Avec :  $\mathbf{r}$  est une matrice diagonale à paramètre positifs.

Donc les paramètres du système flou qui approxime la dynamique du robot sont ajustée via une loi qui dépend de la surface du glissement. Et ainsi quand le système atteint le régime du glissement désiré le système flou est figé car ses paramètres sont plus adaptés car  $\dot{\boldsymbol{\theta}} = 0$ .

Le système global de commande est représenté dans la Figure III.10 ci-dessous.

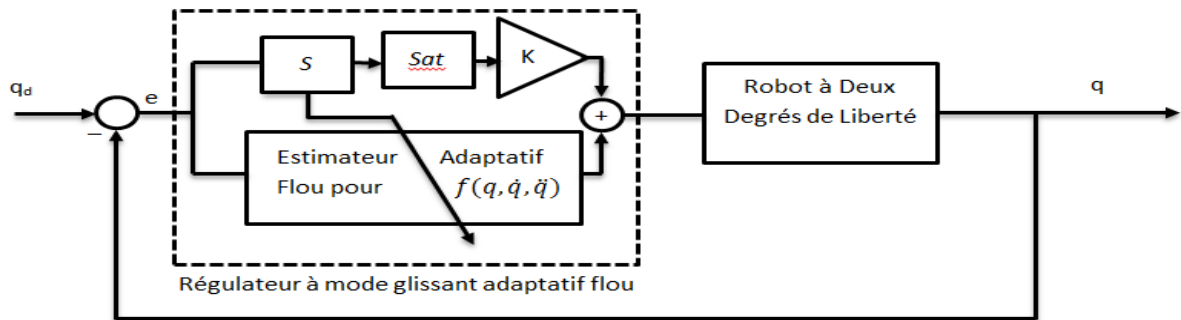


Figure III.10 – Schéma de commande AFSMC

Les résultats de simulations sont illustrés dans les figures ci-dessous :

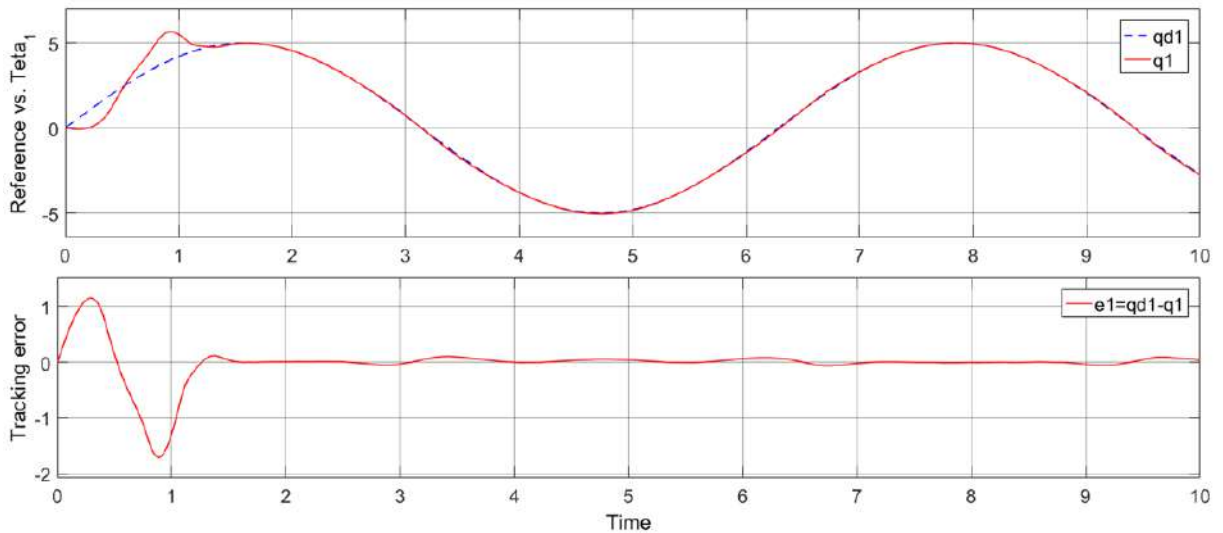


Figure 0.11 - Simulation AFSMC : Teta\_1 pour  $\lambda_1=10, K_v1=100, r1=10$

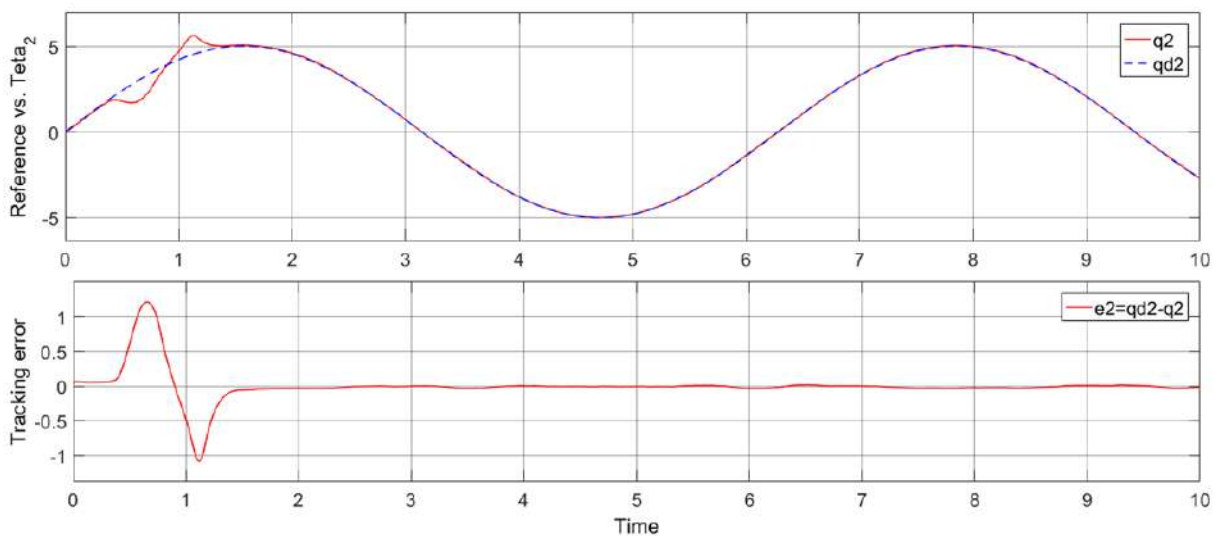


Figure 0.12 - Simulation AFSMC : Teta\_2 pour  $\lambda_2=10, K_v2=70, r2=10$

**Commentaire :** Les angles  $\theta_1$  et  $\theta_2$  suivent parfaitement leurs références après un régime transitoire nécessaire pour l'adaptation du système flou (de durée de moins de 1.5 second) ; l'erreur permanente oscillant entre +/-0.05 pour les deux angles des deux articulations. Le premier dépassement sur les angles  $\theta_1$  et  $\theta_2$  étant plus important (de  $1.2^\circ$ ) par rapport à un consigne de  $5^\circ$ .

### **III.7. Conclusion**

Les résultats de simulations montrent des performances très satisfaisantes de la commande floue adaptative à mode glissant appliquée à un robot à deux degrés de liberté. Tandis que les paramètres de la commande AFSMC (à savoir  $\lambda, K_v$ ) sont presque les mêmes appliqués pour une SMC conventionnelle mais les résultats de simulations montrent des variation d'erreur de poursuite plus adoucis et réduite (SMC : +/-0.05 ; AFSMC : +/-0.01) mais avec des dépassements un peu plus importants (SMC : 1.2 ; AFSMC : 1.2) ceci reste un prix modeste à payer en contrepartie à la non connaissance du modèle mathématique du robot ; en effet c'est l'application de robot qui décidera de la tolérance de l'erreur et son dépassement.

On peut conclure que la commande adaptative floue à mode glissant est d'une performance appréciable en application à un robot à deux degrés de liberté. Le système flou de Takagi-Sugeno est très fiable en l'estimation (modélisation) des dynamiques non linéaires de robot à 2DDL en vue de surpasser les incertitudes en paramètres.



# *Conclusion générale*

## Conclusion générale

Les travaux, présentés dans ce mémoire, traitent la modélisation et la commande d'un bras manipulateur à deux degrés de liberté, l'idée directrice de ce travail est d'apporter une commande efficace et robuste vis-à-vis les incertitudes de modélisations, inconnaisance du modèle et les perturbations externes.

En effet, un robot à deux degrés de liberté est un système non linéaire multi-input multi-output (MIMO) fortement couplé qui présente des incertitudes de modélisation (paramètres mal connu et parfois non disponible) et soumis à des perturbations aléatoires externes. Il est destiné à des applications particulières et nécessite un outil de modélisation fiables et des stratégies de commandes performantes et robustes.

L'objectif de ce mémoire était d'élaborer une commande à mode glissant (non linéaire et robuste) qui profite des avantages de la logique flou. Cette commande hybride et adaptative nous assurera un suivi de trajectoire pour les deux articulations du robot manipulateur à deux degrés de liberté en l'absence d'une partie ou tous les paramètres du robot et en présence des perturbations externes. Pour notre cas la dynamique du robot est entièrement estimée par un compensateur flou à paramètres adaptables via la surface du glissement.

En ce qui concerne la modélisation, on a tout d'abord parcouru les différents modèles du robot manipulateur puis on a établi un modèle dynamique direct et indirect en vue de la simulation de nos commandes robustes.

En première partie c'est la commande par calcul du couple qui simplifie la commande non linéaire en une commande linéaire en l'erreur de suivi (Feedback Linearization) via une transformation non linéaire en deux boucles interne et externe. On a choisi une stratégie de commande linéaire de type PD et les résultats de simulations étaient très satisfaisantes.

Pour la commande adaptative floue à mode glissant, on a substitue la partie de la commande équivalente dépendante des paramètres du robot (matrice du modèle dynamique), la partie discontinue demeure la même que la commande conventionnelle appliquée en première partie et qui a prouvé sa robustesse et fiabilité en suivi de trajectoire désirée pour les deux angles d'articulations rotoïde. L'adaptation des paramètres est assurée par une loi qui dépend de la surface du glissement, ainsi une fois le régime du glissement est atteint (objectif

de la commande à mode glissant) la variation des paramètres est arrêtée et les performances obtenue à ce niveau sont préservées. Les objectifs de la commande sont parfaitement accomplis et le suivi des trajectoires désirées est bien assuré.

Des développements peuvent être envisagés comme extension de ce travail. Nous donnons ci-après quelques directions et perspectives :

- Synthèse d'un estimateur flou pour la partie discontinu de la commande SMC.
- Extension de cette approche hybride aux systèmes à évènement discret.
- Application de cette approche hybride à un autre type du système non linéaire.
- Synthèse d'un réseau de neurones pour estimer la dynamique non linéaire du robot et la proposition d'une loi de mise à jour dynamique des paramètres en vue d'un apprentissage en ligne.

# *Bibliographie*

## Bibliographie

- (NASIRI, 2013). Adaptive sliding mode control for a class of MIMO nonlinear systems with uncertainties. *Journal of the Franklin Institute*.
- (AALIM & AL-SAIF, 2014). Modeling, Simulation and Control of 2-R Robot. *Global Journal of Researches in Engineering*. Global Journals Inc. (USA).
- (ABDELLI, 2015). Commande par Mode de Glissement d'une Machine Asynchrone à Double Alimentation. *Mémoire de Magistère*. Université d'Oran.
- (ALLOUCHE, 2016). Modélisation et commande des robots : nouvelles approches basées sur les modèles Takagi-Sugeno. *Thèse de doctorat*. Université de Valenciennes et du Hainaut-Cambresis.
- (AZOUI, 2009). Commande non linéaire d'un bras manipulateur. *Thèse de Magistère*. Université de Batna.
- (BENAMOR, 2011). Sliding Mode Control, with Integrator, for a Class of MIMO Nonlinear Systems. *Scientific Research Engineering*.
- (DEGHOUDJ, 2013). COMMANDE DES SYSTEMES NON LINEAIRES PAR MODE GLISSANT D'ORDRE SUPERIEUR. *Thèse de Magistère*. Université de Constantine.
- (DOMBRE, 2007). Modeling, Performance Analysis and Control of Robot Manipulators. *Revue Scientifique*. ISTE.
- (KHALIL, 2002). Modelisation Identification and Control of Robots. *Revue Scientifique*. Hermes Penton Science.
- (KHALIL, 2012). Bases de la modélisation et de la commande des robots-manipulateurs de type série. *GDR Robotique*. Université Numérique Ingénieur et Technologie.
- (LEWIS, KAI & YESILDIREK, 1995). Neural Net Robot Controller with Guaranteed Tracking Performance. *IEEE TRANSACTIONS ON NEURAL NETWORKS*. IEEE.
- (MAKHLOUFI, 2015). Modélisation et commande des robots manipulateurs par les outils de l'intelligence artificielle. *Thèse de Doctorat*. Université de Annaba.
- (MOKHTARI & KOLLA, 2016). Conception d'un contrôleur flou pour un système de climatisation. *Mémoire de Master*. Université Djilali Bounaama Khemis Miliana.
- (PISANO, 2011). On the multi-input second-order sliding mode control of nonlinear uncertain systems. *Int. J. Robust. Nonlinear Control*. University of Cagliari, Cagliari, Italy.
- (SAGGAI & MEDAB, 2017). Commande floue mode glissant : Application sur le système de grue 3D. *Mémoire Master*. Université de Ouargla.
- (SALAH, 2013). Association des contrôleurs Glissant – Floue à la commande de la trajectoire d'un robot mobile non holonome. *Mémoire de Magister*. Université El Hadj Lakhdar de Batna.
- (Wang, Rad & Chan 2001). Indirect Adaptive Fuzzy Sliding Mode Control. *Revue Scientifique*. Fuzzy Sets and Systems.
- (Wang, 1993). Stable Adaptive Fuzzy Control of Non Linear Systems. *IEEE Int. Conf. IEEE TRANSACTIONS ON FUZZY SYSTEMS*.
- (Yoo & Ham, 2000). Adaptive Control of Robot Manipulator Using Fuzzy Compensator. *IEEE Int. Conf. IEEE TRANSACTIONS ON FUZZY SYSTEMS*.