

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

**UNIVERSITE MENTOURI-CONSTANTINE**  
**FACULTE DES SCIENCES DE L'INGENIEUR**  
**DEPARTEMENT D'ELECTRONIQUE**

**N° Ordre :**  
**Série :**

## **MEMOIRE**

**Présenté pour l'obtention du diplôme de magistère**  
**électronique**

**OPTION**  
***CONTROLE DES SYSTEMES***

**Par**

**ILYES BOULKAIBET**

**UNE ETUDE EN SIMULATION DE STRATEGIES**  
**DE COMMANDE NON LINEAIRE**

**SOUTENUE LE 09 / 12/ 2007**

**Devant le jury :**

**Président :** Noura Mansouri, Professeur, Université de Constantine  
**Rapporteur :** Khaled Belarbi, Professeur, Université de Constantine  
**Examineurs :** Salim Filali, Professeur, Université de Constantine  
Amar Bentounsi, Maître de conférences, Université de Constantine

## ***REMERCIEMENTS***

Je tiens à exprimer ma profonde gratitude à mon directeur de thèse Monsieur Khaled Belarbi, Professeur à l'Université de Constantine, pour la proposition du sujet de cette thèse et pour ses conseils et son soutien tout au long de ce travail.

J'adresse mes vifs remerciements à Mlle Noura Mansouri, Professeur à l'Université de Constantine, pour l'intérêt qu'elle a porté à ce travail en acceptant de l'examiner et de présider le jury.

J'adresse mes vifs remerciements à Messieurs les membres de jury :

Salim Filali, Professeur à l'Université de Constantine,  
Amar Bentounsi, Maître de conférence à l'Université de Constantine,  
pour l'intérêt qu'ils ont voulu porter à ce travail en acceptant de l'examiner.

Que tous mes amis et collègues trouvent ici le témoignage de mon amitié et de ma reconnaissance.

# Sommaire

## *Chapitre I* : INTRODUCTION GENERALE

## *Chapitre II* : LINEARISATION PAR RETOUR DES SYSTEMES NON LINEAIRES

### **I. Introduction**

### **II Bases mathématiques**

*II.1* Introduction a la géométrie différentielle :

*II.1.1* Difféomorphisme :

*II.1.2* Champ de vecteurs :

*II.1.3* Dérivée de Lie :

*II.1.4* Théorème de Frobenius :

### **III Stabilité des systems nonlineaires**

*III.1* Stabilité du point d'équilibre :

*III.2* Méthode directe de Lyapunov ou seconde méthode :

### **IV Linearisation des systemes nonlineaires**

*IV.1* La représentation d'état affine :

*IV.2* Linéarisation par retour d'état :

*IV.3* Linéarisation partielle par retour d'état :

*IV.4* Extension aux systèmes multi-entrées :

*IV.5* Linéarisation entrée sortie:

*IV.5.1* Degré relatif global :

*IV.5.2* linéarisation en entrée-sortie par retour d'état :

## *Chapitre III* : UNE SOLUTION ANALYTIQUE A COMMANDE PREDICTIVE BASEE SUR UN MODELE NON LINEAIRE SANS CONTRAINTES

### **I Introduction**

### **II La commande prédictive basée sur un model non linéaire**

### **III Modèle de prédiction affine : Linéarisation par retour d'état**

*III.1* Position du problème :

*III.2* Solution par modèle d'état :

*III.3* Solution par la Commande GPC :

### **IV Résultats de simulation**

*IV.1* Exemple 1 :

*V.2.1* Application de la commande prédictive GPC :

*V.2.2* Application de la commande prédictive MBPC :

*V.3* Exemple 2 : Application de la commande prédictive MBPC :

## **Chapitre IV : LA METHODE DU «BACKSTEPPING»**

### **I Introduction**

### **II Commande par backstepping**

*II.1* Approche non adaptative :

*II.1.1* Principe :

*II.1.2* Simulation :

*II.2* Approche adaptative :

*II.2.1* Conditions d'implantation :

*II.2.2* Simulation :

*II.3* Généralisation :

*II.3.1* Théorie :

*II.3.1* Résultats de simulation Application à un robot avec le joint flexible :

### **III Backstepping avec observateur**

*III.1* Approche non adaptative avec observateur :

*III.2* Résultat de simulation :

# CHAPITRE 1 :

## INTRODUCTION GENERALE

Il est généralement admis que bien que tous les systèmes réels aient un comportement non linéaire, un grand nombre peut être commandés par les techniques de l'automatique linéaire. Ceci est possible lorsque le système non linéaire peut être linéarisé pour de petites variations autour d'un point de fonctionnement ou lorsque des simplifications sont introduites dans les modèles. Il existe cependant un certain nombre de systèmes, parmi lesquels on peut citer les moteurs à courant alternatif, les robots manipulateurs, les satellites, quelques réacteurs chimiques pour lesquels une modélisation linéaire n'est plus possible ni désirable ... Pour cela plusieurs méthodes de commandes basées sur des modèles non linéaires ont été proposées. Une des premières approches de conception systématique d'une commande non linéaire est la commande dite linéarisante ou linéarisation par retour, feedback linéarisation [1]. Elle est basée sur des changements de variables et des développements mathématiques rigoureux et est généralement applicable à des systèmes ayant une structure particulière en l'occurrence les systèmes affines par rapport à la commande. Un peu plus tard est apparue la commande dite backstepping [2] applicable à des systèmes dont les modèles sont donnés sous une forme particulière et la méthodologie de conception est basée sur la théorie de la stabilité de Lyapunov.

Avec la commande en régime de glissement développée dans l'ex URSS, ces deux approches sont celles qui ont eu le plus d'écho dans la communauté scientifique et qui ont permis de résoudre plusieurs problèmes pratiques.

Par ailleurs la commande prédictive à base de modèle linéaire a connu un large succès dans l'industrie. Cependant sa version non linéaire avec ou sans contraintes est confrontée à la résolution numérique en ligne d'un problème d'optimisation non linéaire difficile. Il serait alors souhaitable d'envisager comme dans le cas linéaire une solution analytique pour les problèmes sans contraintes. C'est dans cette optique que s'inscrit une partie de notre travail : nous utilisons la théorie de la commande linéarisante pour construire une solution analytique pour le cas d'une commande prédictive à base de modèle non linéaire sans

contraintes. L'autre partie de notre travail est une étude détaillée avec applications en simulation de la commande backstepping.

Ce mémoire est organisé comme suit : le deuxième chapitre introduit la théorie de la commande linéarisante, ou théorie de la linéarisation par contre réaction, *feedback linearisation*, le troisième chapitre développe la méthode de solution analytique proposée pour la commande prédictive non linéaire sans contraintes. Deux cas sont traités, le cas de la commande prédictive généralisée avec un modèle entrée sortie et le cas d'un modèle sous forme d'équations d'état linéaires. Le quatrième chapitre est une étude en simulation de la commande backstepping. Plusieurs situations sont étudiées : commande non adaptative, commande adaptative et commande avec observateur d'état.

# CHAPITRE 2 :

## LINEARISATION PAR RETOUR DES SYSTEMES NON LINEAIRES

### I. Introduction

Ce chapitre introduit les éléments de base de la commande linéarisante, feedback linéarisation. Ces éléments sont basés sur [1].

### II Bases Mathématiques

#### II.1 Introduction a la géométrie différentielle :

**II.1.1 Difféomorphisme :** Soit  $p$  un point de  $R^n$  et  $\varphi$  une application d'un ouvert  $U \subset R^n$  dans un ouvert  $V \subset R^n$ , on dit que  $\varphi$  est un difféomorphisme local dans un voisinage  $U$  de  $p$  si  $\varphi$  est inversible de  $U$  dans un voisinage  $V$  du point  $\varphi(p)$  de  $V$ , et si  $\varphi^{-1}$  et  $\varphi$  est continue.

**II.1.2 Champ de vecteurs :** un champ de vecteur sur  $R^n$  est une fonction dérivable

$$f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n : \quad f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix} \quad (2.1)$$

**II.1.3 Dérivée de Lie :** Pour une fonction scalaire  $h : x \rightarrow h(x)$  et un champ de vecteur  $f(x)$ , la dérivée de Lie est définie par :

$$L_f h = \sum_{i=1}^n f_i(x) \frac{\partial h}{\partial x_i} = \langle dh, f \rangle \quad (2.4)$$

Avec

$$L_f^i h = L_f(L_f^{i-1} h), \quad L_f^1 h = L_f h, \quad L_f^0 h = h. \quad (2.5)$$

#### a) Crochet de Lie :

Deux champs de vecteurs peuvent être composés comme suit :

$$[f, g](h) = f(g(h)) - g(f(h)) = L_f L_g h - L_g L_f h \quad (2.6)$$

$$[f, g](h) = ad_f g \quad \text{Où} \quad [f, g](h) = L_{fg} \quad (2.7)$$

**b) Propriétés :**

$$- \text{Distributivité :} \quad [\alpha_1 f_1 + \alpha_2 f_2, g] = \alpha_1 [f_1, g] + \alpha_2 [f_2, g] \quad (2.8)$$

$$[f, \alpha_1 g_1 + \alpha_2 g_2] = \alpha_1 [f, g_1] + \alpha_2 [f, g_2] \quad (2.9)$$

$$- \text{Anti-commutativité :} \quad [f, g] = -[g, f] \quad (2.10)$$

$$- \text{Identité de Jacobi :} \quad ad_f ad_g q + ad_g ad_q f + ad_q ad_f g = 0 \quad (2.11)$$

- **Formule de Leibniz's :**

$$L_f \langle d(L_f^{k-s} h), ad_f^s g \rangle = \langle d(L_f^{k-s+1} h), ad_f^s g \rangle + \langle d(L_f^{k-s} h), ad_f^{s+1} g \rangle \quad (2.12)$$

**Définition 2.1 :** Une distribution  $D$  est dite involutive si et seulement si pour tout couple de champs de vecteurs  $f$  et  $g$  de  $D$  on a  $[f, g] \in D$ .

Une distribution involutive est donc caractérisée par  $[D, D] \subset D$ . Si  $D$  n'est pas involutive, on peut définir sa clôture involutive :

**Définition 2.2 :** La clôture involutive  $\bar{D}$  ou *inv.cl.D* d'une distribution  $D$  est la plus petite distribution involutive contenant  $D$ .

**II.1.4 Théorème de Frobenius :**

Soit une distribution  $D$  de dimension  $r$  sur  $W$ , un sous-ensemble  $R^n$ , alors autour de n'importe quel point  $p \in W$  il existe un rang de voisinage  $(U, x_1, \dots, x_n)$ ,  $U$  un voisinage de  $p$  avec  $x_i = \varphi_i(q), 1 \leq i \leq n$ , tel que  $\partial \varphi_i \in D^\perp$ , c.-à.-d pour n'importe quelle fonction  $f \in D$

$$\langle \partial \varphi_i, f \rangle = 0, \quad \forall x \in U. \quad r+1 \leq i \leq n \quad (2.13)$$

si, et seulement si,  $D$  est une distribution involutive.

Par suite du théorème de Frobenius, considérons  $r$  champ de vecteurs  $f_1, \dots, f_r$  dans  $R^n$ , il existe une solution non triviale  $\varphi$  pour l'équation différentielle :

$$\langle d\varphi, f_i \rangle = 0 \quad 1 \leq i \leq r \quad (2.14)$$

Si et seulement si la clôture involutive de la distribution  $span(f_1, \dots, f_r)$  a une dimension inférieure ou égale à  $n-1$  où  $span$  signifie "sous-espace vectoriel engendré par".

### III Stabilité des systèmes nonlinéaires

#### III.1 Stabilité du point d'équilibre :



**a) Définition 2.3 :** Stabilité au sens de Lyapunov

L'état d'équilibre  $x_e$  est dit **stable** si :  $\forall \varepsilon > 0, \exists \alpha > 0$  tel que si  $\|x(0) - x_e\| < \alpha$  alors  $\|x(t) - x_e\| < \varepsilon \quad \forall t \geq 0$ . Dans le cas contraire,  $x_e$  est dit **instable**.

**b) Définition 2.4 :** *stabilité asymptotique*

Un point d'équilibre  $x_e$  est asymptotiquement stable s'il est stable et s'il existe  $\alpha > 0$  tel que :  $\|x(0) - x_e\| < \alpha \Rightarrow \lim_{t \rightarrow +\infty} x(t) = x_e$ .

**c) Définition 2.5 :** *stabilité exponentielle*

Un point d'équilibre  $x_e$  est exponentiellement stable s'il existe  $\alpha > 0$  et  $\lambda > 0$  tel que :

$$\forall t > 0, \exists B_r(x_e, r), \forall x_0 \in B_r, \|x(t) - x_e\| \leq \alpha \|x(0) - x_e\| e^{-\lambda t}$$

**d) Définition 2.6 :** *stabilité globale*

Si la propriété de stabilité asymptotique, (exponentielle) est vérifiée quelque soit  $x_0$ , le point d'équilibre est globalement asymptotiquement, (exponentiellement) stable

### **III 2 Méthode directe de Lyapunov ou seconde méthode :**

**Théorème 2.1 :** *stabilité (asymptotique) locale*

S'il existe une fonction scalaire  $V(x)$  de l'état dont les dérivées partielles premières sont continues et telle que:

1-  $V$  est une fonction candidate de Lyapunov.

2-  $\dot{V}$  est localement semi définie négative dans un voisinage de l'origine,  $\Omega$ .

Alors le point d'équilibre 0 est stable et un domaine de conditions initiales stables est délimité par n'importe quelle équipotentielle de Lyapunov contenue dans  $\Omega$ .

Si  $\dot{V}$  est localement définie négative dans  $\Omega$ , alors la stabilité est dite **localement asymptotique** dans la partie de l'espace délimité par n'importe quelle équipotentielle de Lyapunov contenue dans  $\Omega$ .

**Théorème 2.2:** *stabilité globale asymptotique*

S'il existe une fonction  $V$  telle que

1-  $V$  est une fonction candidate de Lyapunov.

2-  $\dot{V}$  est définie négative.

3- La condition  $\|x\| \rightarrow +\infty$  implique  $V(x) \rightarrow +\infty$

Alors 0 est un point d'équilibre globalement asymptotiquement stable.

## IV Linéarisation des systèmes nonlinéaires

### IV.1 La représentation d'état affine :

De nombreux systèmes dynamiques admettent une représentation d'état non linéaire particulière dite **affine** en la commande. Elle est donnée par les équations suivantes :

**Cas continu :**

$$\begin{cases} \dot{x}(t) = f(x) + \sum_{i=1}^m g_i(x)u_i(t) \\ y(t) = h(x) \end{cases} \quad (2.15)$$

Si le système est monovariante, on a :

$$\begin{cases} \dot{x}(t) = f(x) + g(x)u(t) \\ y(t) = h(x) \end{cases} \quad (2.16)$$

**Cas discret :**

$$\begin{cases} x(k+1) = f(x) + \sum_{i=1}^m g_i(x)u_i(k) \\ y(k) = h(x) \end{cases} \quad (2.17)$$

Si le système est monovariante on a :

$$\begin{cases} x(k+1) = f(x) + g(x)u(k) \\ y(k) = h(x) \end{cases} \quad (2.18)$$

### IV.2 Linéarisation par retour d'état :

**Théorème 2.3 :**

Le système à une seule entrée (2,16) est localement linéarisable par retour d'état si, et seulement si, autour d'un voisinage de l'origine U :

**i)**  $span\{g, \dots, ad_f^{n-1}g\} = R^n$ ,

**ii)** la distribution  $G_{n-2} = span\{g, \dots, ad_f^{n-2}g\}$  est involutive et à un rang constant égal à  $n-1$ ,

Ou, d'une manière équivalente, si et seulement si :

**iii)** Les distributions  $G_i = span\{g, \dots, ad_f^i g\}$ ,  $0 \leq i \leq n-1$  sont involutives et de rang constant égal à  $i+1$ .

Si (i) et (ii) sont satisfaites donc nous pouvons appliquer le théorème de Frobenius, qui garantit l'existence d'une fonction dérivable  $h$  telle que dans un voisinage de l'origine :

$$\begin{aligned} \langle dh, ad_{(-f)}^{n-1}g \rangle &\neq 0 \\ \langle dh, ad_{(-f)}^i g \rangle &= 0, \quad 0 \leq i \leq n-2 \end{aligned} \quad (2.19)$$

et  $h(0) = 0$ . En d'autres termes (i) et (ii) garantissent l'existence d'une solution  $h$  pour l'ensemble d'équations partielles linéaires :

$$\langle dh, g \rangle, \langle dh, ad_{(-f)}g \rangle, \dots, \langle dh, ad_{(-f)}^{n-1}g \rangle = (0, \dots, 0, \gamma(x)) \quad (2.20)$$

Pour une certaine fonction  $\gamma(x)$ ,  $\gamma(0) = 0$ , définissons  $n$  fonctions  $h(x), L_f h(x), \dots, L_f^{n-1} h(x)$ , nous pouvons montrer que :

$$z = (z_1, z_2, \dots, z_n) = (h(x), L_f h(x), \dots, L_f^{n-1} h(x)) = T(x) \quad (2.21)$$

Avec  $z = T(x)$  un difféomorphisme local. Le système (2.16) est exprimé en  $z$ -coordonnées par :

$$\dot{z} = L_f^i h(x) + L_g L_f^{i-1} h(x) u, \quad 1 \leq i \leq n. \quad (2.22)$$

De (2.20), on applique la formule de Leibniz

$$\begin{aligned} L_g L_f^i h(x) &= 0, & 0 \leq i \leq n-2 \\ L_g L_f^{n-1} h(x) &= \gamma(x) \end{aligned} \quad (2.23)$$

L'équation (2.22) devient :

$$\begin{aligned} \dot{z}_i &= z_{i+1}, & 1 \leq i \leq n-1 \\ \dot{z}_n &= L_f^n h(x) + L_g L_f^{n-1} h(x) u. \end{aligned} \quad (2.24)$$

On définit alors le retour d'état

$$u = -\frac{L_f^n h(x)}{L_g L_f^{n-1} h(x)} + \frac{1}{L_g L_f^{n-1} h(x)} v \quad (2.25)$$

#### IV.3 Linéarisation partielle par retour d'état :

Le système (2.16) est localement partiellement linéarisable par retour d'état avec l'indice  $r$  si la distribution  $\overline{G_{r-2}}$ , la clôture involutive de  $G_{r-2}$ , à un rang constant au moins égal à  $n-1$  en  $U_0$  et au voisinage de l'origine, il existe un nombre entier  $r$ ,  $2 \leq r \leq n$ , tel que

$$ad_f^{r-1} g(x) \notin \overline{G_{r-2}}(x) = \text{inv.cl.span}\{g, \dots, ad_f^{r-2} g\}, \quad \forall x \in U_0. \quad (2.26)$$

Nous pouvons appliquer le théorème de Frobenius qui garantit l'existence d'une fonction dérivable  $h$  telle que dans un voisinage de l'origine :

$$\begin{aligned} \langle dh, ad_{(-f)}^{r-1} g \rangle &\neq 0 \\ \langle dh, \overline{G_{r-2}} \rangle &= 0, & 0 \leq i \leq n-2 \end{aligned} \quad (2.27)$$

Avec :

$$\begin{aligned} ad_f^{r-1}g &\notin \overline{G_{r-2}} \\ rank \{g, \dots, ad_f^{r-2}, ad_f^{r-1}\} &= r, \end{aligned} \quad (2.28)$$

Par conséquent, si on applique encore le théorème de Frobenius, puisque  $span(g)$  est une distribution involutive de rang égal à 1 on peut déterminer, en plus les fonctions déjà déterminées  $(h, \dots, L_f^{r-1}h)$ ,  $n-r$  fonctions  $\xi_1(x), \dots, \xi_{n-r}(x), \xi_i(0) = 0$ , telles que :

$$L_g \xi_i = \langle d\xi_i(x), g \rangle = 0, \quad 1 \leq i \leq n-r \quad (2.29)$$

Le résultat du développement mathématique donne :

$$\begin{aligned} \xi_i &= \xi_i(x), & 1 \leq i \leq n-r \\ z_j &= L_f^{j-1}h, & 1 \leq j \leq r \end{aligned} \quad (2.30)$$

Donc on a:

$$\begin{aligned} \dot{\xi}_i &= L_f \xi_i(x) \\ \dot{z}_j &= z_{j+1}, & 1 \leq j \leq r-1, \\ \dot{z}_r &= L_f^r h + u L_g L_f^{r-1} h, \end{aligned} \quad (2.31)$$

Si on définit :

$$u = \frac{1}{L_g L_f^{r-1} h} (-L_f^r h + v), \quad (2.32)$$

On obtient :

$$\begin{aligned} \dot{\xi}_i &= L_f \xi_i(x) \\ \dot{z}_j &= z_{j+1}, & 1 \leq j \leq r-1, \\ \dot{z}_r &= v, \end{aligned} \quad (2.33)$$

#### IV.4 Extension aux systèmes multi-entrées :

##### Théorème 2.4 :

Le système non linéaire (2.15) est localement linéarisable dans un voisinage de l'origine, par une **transformation de retour d'état non singulière** (qui se compose d'un retour non singulier et d'un difféomorphisme dans les systèmes contrôlables linéaires sous la forme de contrôleur de Burnovsky avec les indices de contrôlabilité  $k_1 \geq k_2 \geq \dots \geq k_m$  si, et seulement si, en  $U_0$ , un voisinage de l'origine :

i)  $G_{k_i-2}, 1 \leq i \leq m$ , est involutive et a un rang constant,

ii)  $\text{rank}G_{k_1-1} = n$ ,

Avec :

a) **le retour d'état non singulier :**

$$u = K(x) + \beta(x)v; \quad K(0) = 0 \quad (2.34)$$

$K(x)$  est une fonction dérivable de  $V0$  en  $R^n$ ,  $\beta(x)$  est une matrice de  $m \times m$  avec les entrées dérivable, non singulières en  $V0$

b) **un difféomorphisme local en  $V0$**

$$z = T(x), \quad T(0) = 0 \quad (2.35)$$

c) les indice de contrôlabilité  $\{k_1, \dots, k_m\}$  associés au système (2.15) sont définis par :

$$k_i = \text{card}\{m_j \geq i: j \geq 0\}, \quad 1 \leq i \leq m \quad (2.36)$$

Avec :

$$\begin{aligned} m_0 &= \text{rank}G_0 \\ m_1 &= \text{rank}G_1 - \text{rank}G_0 \\ &\vdots \\ m_{n-1} &= \text{rank}G_{n-1} - \text{rank}G_{n-2} \end{aligned} \quad (2.37)$$

Les condition (i) et (ii) garantissent l'existence de  $m$  fonctions dérivable  $\phi_1(x), \dots, \phi_m(x)$  telles que :

$$\langle d\phi_i, G_{k_j-2} \rangle = 0, \quad j \geq i \quad (2.38)$$

Et la matrice :

$$\begin{bmatrix} \langle d\phi_1, ad_f^{k_1-1} g_1 \rangle & \dots & \langle d\phi_1, ad_f^{k_1-1} g_m \rangle \\ \vdots & & \vdots \\ \langle d\phi_m, ad_f^{k_m-1} g_1 \rangle & \dots & \langle d\phi_m, ad_f^{k_m-1} g_m \rangle \end{bmatrix} \quad (2.39)$$

est non singulière en  $V(0)$ . La transformation et le difféomorphisme du retour d'état non singuliers de linéarisation sont, respectivement :

$$v = \begin{bmatrix} L_f^{k_1} \phi_1 \\ \vdots \\ L_f^{k_m} \phi_m \end{bmatrix} + \begin{bmatrix} L_{g_1} L_f^{k_1-1} \phi_1 & \dots & L_{g_m} L_f^{k_1-1} \phi_1 \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{k_m-1} \phi_m & \dots & L_{g_m} L_f^{k_m-1} \phi_m \end{bmatrix} u \quad (2.40)$$

Avec :

$$z = \begin{bmatrix} \phi_1 \\ \vdots \\ L_f^{K_1-1} \phi_1 \\ \vdots \\ \phi_m \\ \vdots \\ L_f^{K_m-1} \phi_m \end{bmatrix} \quad (2.41)$$

#### IV.5 Linéarisation entrée sortie

Dans cette section nous présentons une fonction de sortie dérivable  $h(x): R^n \rightarrow R^n$  qui définit la variable de sortie  $y$

$$y = h(x)$$

Nous supposons que le variable  $y$  doit être commandable et que le variable d'état  $x$  est disponible pour les mesures. Le système est décrit :

$$\begin{aligned} \dot{x} &= f(x) + g(x)u, \quad x \in R^n, \quad u \in R \\ y &= h(x), \quad y \in R \end{aligned} \quad (2.42)$$

##### IV.5.1 Degré relatif global :

Le degré relatif global  $\rho$  du système (2.42) est défini comme un nombre entier tels que:

$$\begin{aligned} L_g L_f^i h(x) &= 0, \quad \forall x \in R^n, 0 \leq i \leq \rho - 2 \\ L_g L_f^{\rho-1} h(x) &\neq 0, \quad \forall x \in R^n. \end{aligned} \quad (2.43)$$

Si  $L_g L_f^i h(x) = 0, \quad \forall x \in R^n, \forall i \geq 0$  On dit que  $\rho = \infty$

##### IV.5.2 Linéarisation en entrée-sortie par retour d'état :

Le système (2.42) est localement partiellement linéarisable par retour d'état avec l'index  $\rho$ , et localement linéarisable en entrée-sortie par retour d'état c.-à-d localement la relation entrée-sortie équivalente à:

$$\begin{aligned}
\dot{\xi} &= \phi(\xi, z), & \xi \in R^{n-\rho} \\
\dot{z}_i &= z_{i+1}, & 1 \leq i \leq \rho-1 \\
\dot{z}_\rho &= v \\
y &= x_1 \\
u &= \frac{1}{L_g L_f^{\rho-1} h} (v - L_f^\rho h)
\end{aligned} \tag{2.44}$$

Si et seulement si:  $\rho \leq n$ ,

**Système avec incertitudes :**

Dans cette partie le système est de la forme :

$$\dot{x} = f(x) + q(x, \theta(t)) + g(x)u \tag{2.45}$$

**Théorème 2.5: (incertitudes triangulaires)** si le système (2,45) vérifie :

i) Le système nominal  $(f, g)$  est localement (globalement) linéarisable par retour d'état,

ii) La supposition stricte de triangularité :

$$ad_q G_i \subset G_i, \quad 0 \leq i \leq n-2$$

est satisfaite dans  $U_0$ , voisinage de l'origine (dans  $R^n$ ) ;

Le système (2,45) est localement (globalement) linéarisable par retour d'état équivalent à:

$$\begin{aligned}
\dot{z}_j &= z_{j+1} + \Phi_j(z_1, \dots, z_j, \theta(t)), & 1 \leq j \leq n-1 \\
\dot{z}_n &= v + \Phi_n(z_1, \dots, z_n, \theta(t)).
\end{aligned} \tag{2.46}$$

Avec :

$$u = \frac{1}{L_g L_f^{n-1} h} (v - L_f^n h)$$

# CHAPITRE 3

## UNE SOLUTION ANALYTIQUE A COMMANDE PREDICTIVE BASEE SUR UN MODELE NON LINEAIRE

### I. Introduction

Dans ce chapitre nous introduisons une méthode pour construire une solution analytique pour le cas d'une commande prédictive à base de modèle non linéaire sans contraintes. Cette méthode utilise la théorie de la commande linéarisante. Deux cas sont traités, le cas de la commande prédictive généralisée avec un modèle entrée sortie et le cas d'un modèle sous forme d'équations d'état linéaires.

### II. La commande prédictive à base de modèle

On considère le système discret non linéaire :

$$x(k+1) = f(x(k), u(k)) \quad (3.1)$$

Où  $x$  est l'état du système,  $u$  est le vecteur d'entrée du système.

L'objectif est de trouver une loi de commande qui permet de régler l'état de système à l'origine. Pour trouver cette loi, il faut résoudre le problème d'optimisation suivant :

$$\underset{u}{\text{Min}} J_N(x, u) = \sum_{j=1}^{j=N-1} H(\hat{x}(k+j), u(k+j-1)) \quad (3.2)$$

Avec

$$\hat{x}(k+j) = f(\hat{x}(k+j-1), u(k+j-1)), \hat{x}(k) = x(k), \hat{x}(k+j) \in X \text{ et } u(k+j) \in U$$

$x(k)$  est l'état initial ,

$\hat{x}(k+j), j=1 \dots N$ ,  $j$  sont les prédictions sur les états futurs obtenues à l'instant  $k$  à partir du modèle de prédiction (3.1) et le  $N$  est l'horizon de prédiction.



La solution de ce problème donne une séquence optimale des commandes

$$\mathbf{u}^* = (u^*(k), u^*(k+1), \dots, u^*(k+N-1)) \quad (3.3)$$

Et un trajectoire optimale  $\mathbf{x}^*$  :

$$\mathbf{x}^* = (\hat{x}(k+1), \hat{x}(k+2), \dots, \hat{x}(k+N)) \quad (3.4)$$

La définition du problème et sa méthode de résolution dépend la nature du modèle de prédiction (3.1) et de la fonction coût (3.2). Lorsque le modèle est linéaire et qu'il n'y a pas de contraintes sur les différentes variables, une solution analytique peut être facilement obtenue [4], s'il y a des contraintes il existe des méthodes efficaces et rapides pour la résolution numérique en ligne du problème d'optimisation. Cependant dans le cas de modèle non linéaire, les méthodes de résolution demandent un temps de calcul assez important, ce qui limite l'applicabilité de la commande prédictive non linéaire. Dans ce qui suit nous utilisons la théorie de la linéarisation par retour introduite au chapitre précédent pour construire une solution analytique à la commande prédictive non linéaire sans contraintes dans le cas d'un modèle de prédiction affine en la commande. Deux cas seront traités : la commande prédictive linéaire à base de modèle d'état et la commande prédictive généralisée.

### III Modèle de prédiction affine : Linéarisation par retour d'état

#### III.1 Position du problème et solution

Dans le cas où le système (3.1) est dans une représentation d'état non linéaire *affine* en la commande donnée par les équations suivantes :

$$\begin{aligned} x(k+1) &= f(x(k)) + g(x(k))u \\ y &= h(x(k)) \end{aligned} \quad (3.5)$$

Si on trouve une relation entre les états  $x$  et une autre variable  $z$  tel que :

$$\begin{aligned} z &= T(x) \\ z(k+1) &= A_z z + B_z v \\ y &= Cz \\ \text{avec } v &= \beta(x)u + k(x) \end{aligned} \quad (3.6)$$

La loi de commande qui permet de régler l'état de système à l'origine, devient le problème d'optimisation suivant :

$$\text{Min}_v J_N(z, v) = \sum_{j=1}^{j=N-1} R(\hat{z}(k+j), v(k+j-1)), \quad (3.7)$$

### III.2 La commande prédictive à base de modèle d'état

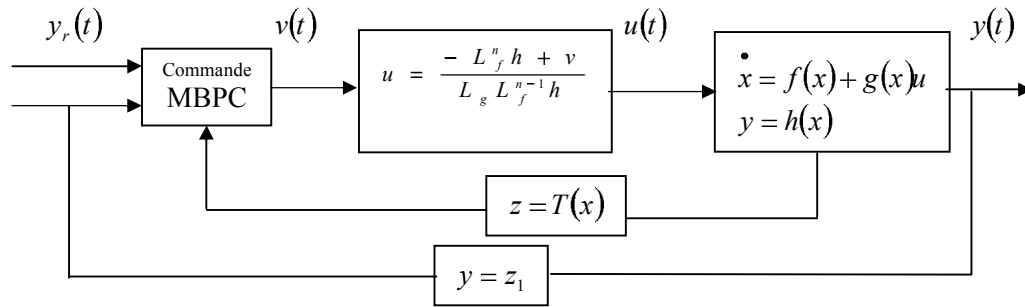
Dans ce cas:

$$\text{Min}_v J(y, v) = \sum_{j=N_\omega}^{N_H} \|\hat{y}(k+j|k) - y_r(k+j)\|_{Q(j)}^2 + \sum_{j=0}^{N_u-1} \|\Delta \hat{v}(k+j|k)\|_{R(j)}^2 \quad (3.8)$$

En vertu des calculs présentés au deuxième chapitre, la loi de commande est donnée par :

$$u = \frac{-L_f^n h + v}{L_g L_f^{n-1} h}$$

La figure (3. 1) montre la méthode qui permet de calculer la loi de commande.



Figure(3.1)

### III.3 Solution pour la commande prédictive généralisée (GPC) :

On peut mettre l'équation (3.6) sous la forme d'une fonction de transfert par une représentation linéaire donnée par l'expression suivante :

$$A(z^{-1}) \cdot y(t) = B(z^{-1}) \cdot v(t-1) + X(t) \quad (3.9)$$

Où:  $A$  et  $B$  sont des polynômes de l'opérateur de retard  $z^{-1}$ .

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{na} z^{-na}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_{nb} z^{-nb}$$

$na$  et  $nb$  étant des nombres entiers représentant les degrés de  $A$  et  $B$  resp.

$X(t)$  : est un terme relatif à une perturbation survenant à un instant  $t'$  quelconque.

Dans le cas de la GPC,  $x(t)$  est tel que  $X(t) = c(z^{-1}) \cdot e(t) / \Delta$ ,

en substituant ceci dans l'équation (5.9) on aura :

$$A(z^{-1}) \cdot y(t) = B(z^{-1}) \cdot v(t-1) + C(z^{-1}) \cdot e(t) / \Delta \quad (3.10)$$

Où :  $C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{nc} z^{-nc}$  ( $nc$  est le degré de  $C$ ).

$\Delta = 1 - z^{-1}$ , appelé opérateur de différence, son rôle est d'assurer une action intégrale sur le contrôleur à fin d'annuler l'effet des perturbations.

$e(t)$  : est un bruit gaussien blanc à moyenne nulle.

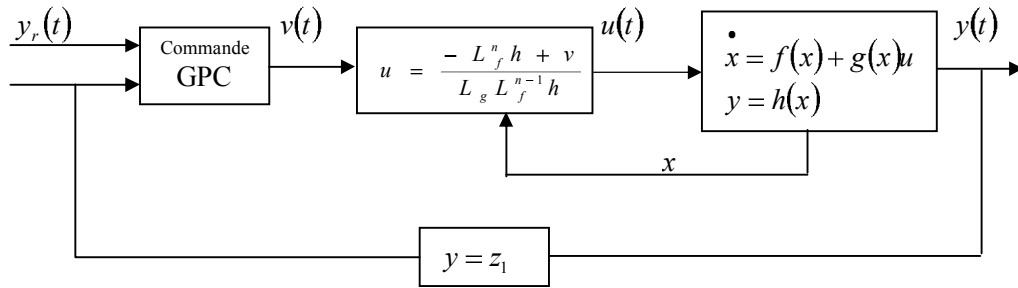
Le critère devient :

$$\text{Min}_v J_N(z, v) = \sum_{j=1}^{j=N_2} (\hat{y}(k+j) - y_r(k+j))^2 + \lambda \sum_{j=1}^{j=N_1} (\nabla v(k+j-1))^2, \quad (3.11)$$

Dela même façon que ci-dessus la loi de commande est donnée par :

$$u = \frac{-L_f^n h + v}{L_g L_f^{n-1} h}$$

La figure (3. 2) montre cette stratégie de commande



Figure(3.2)

#### IV Résultats de simulation

Dans cette section nous présentons deux exemples d'applications de la méthodologie présentées. Le premier exemple est la commande de la position d'un robot à un seul joint, la deuxième est une commande multivariable d'un moteur asynchrone.

##### IV.1 Exemple 1 : Commande d'un robot

On considère ici la commande d'un robot avec le joint flexible. Les variables d'états  $x_1, x_2, x_3, x_4$  sont  $x_1 = q_1, x_2 = \dot{q}_1, x_3 = q_2, x_4 = \dot{q}_2$ . Il s'agit de commander la position représentée par l'angle  $q_1$  en agissant sur le couple  $u$ . Le modèle est réécrit sous la forme suivante :

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -\left(\frac{F_l}{J_l}\right)x_2 - \left(\frac{Mgl}{J_l}\right)\sin(x_1) - \left(\frac{k}{J_l}\right)(x_1 - x_3) \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= -\left(\frac{F_m}{J_m}\right)x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) + \left(\frac{1}{J_m}\right)u
\end{aligned} \tag{3.12}$$

$$f(x) = \begin{bmatrix} x_2 \\ -\frac{F_l}{J_l}x_2 - \left(\frac{Mgl}{J_l}\right)\sin(x_1) - \left(\frac{k}{J_l}\right)(x_1 - x_3) \\ x_4 \\ -\frac{F_m}{J_m}x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix}, \tag{3.13}$$

#### IV.1.1 Application de la commande prédictive GPC :

On définit la fonction  $h = x_1$ , avec  $z = T(x)$  un difféomorphisme local défini par :

$$z = T(x) = [h, L_f h, L_f^2 h, L_f^3 h] = [z_1, z_2, z_3, z_4] \tag{3.14}$$

On obtient alors:

$$\begin{aligned}
z_1 &= x_1 \\
z_2 &= x_2 \\
z_3 &= -\frac{F_l}{J_l}x_2 - \left(\frac{Mgl}{J_l}\right)\sin(x_1) - \left(\frac{k}{J_l}\right)(x_1 - x_3) \\
z_4 &= -\left(\frac{Mgl}{J_l}\cos(x_1) + \frac{k}{J_l}\right)x_2 + \frac{F_l}{J_l}\left[\frac{F_l}{J_l}x_2 + \frac{Mgl}{J_l}\sin(x_1) + \frac{k}{J_l}(x_1 - x_3)\right] + \frac{k}{J_l}x_4.
\end{aligned} \tag{3.15}$$

Le résultat de linéarisation par retour d'état est donnée par :

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
\dot{z}_3 &= z_4 \quad \text{avec} \quad u = -\frac{L_f^4 h}{L_g L_f^3 h} + \frac{1}{L_g L_f^3 h} v \\
\dot{z}_4 &= v
\end{aligned} \tag{3.16}$$

Avec

$$\begin{aligned}
L_f^4 h &= \left( \frac{Mgl}{J_1} \sin(x_1) + \frac{F_1 Mgl}{J_1^2} \cos(x_1) + \frac{kF_1}{J_1^2} \right) x_2 \\
&\quad + \left( \frac{Mgl}{J_1} \cos(x_1) + \frac{k}{J_1} - \frac{F_1^2}{J_1^2} \right) \left[ \frac{F_1}{J_1} x_2 + \left( \frac{Mgl}{J_1} \right) \sin(x_1) + \left( \frac{k}{J_1} \right) (x_1 - x_3) \right] \\
&\quad - \frac{kF_1}{J_1^2} x_4 + \frac{k}{J_1} \left[ \frac{k}{J_m} (x_1 - x_3) - \frac{F_m}{J_m} x_4 \right] \\
L_g L_f^3 &= \frac{k}{J_1 J_m}.
\end{aligned} \tag{3.17}$$

On peut écrire le système (3.16) sous la forme d'une fonction de transfert, donc :

$$z_1^{(4)} = y^{(4)} = v \quad \Rightarrow H(s) = \frac{1}{s^4} \tag{3.18}$$

Si on discrétise le système pour  $Te=0.1s$  on a alors :

$$\begin{aligned}
A(z^{-1}) &= 1 - 4z^{-1} + 6z^{-2} - 4z^{-3} + z^{-4} \\
B(z^{-1}) &= 8.333 \times 10^{-7} + 2.167 \times 10^{-5} z^{-1} \\
&\quad + 5.5 \times 10^{-5} z^{-2} + 2.167 \times 10^{-5} z^{-3} + 8.333 \times 10^{-7} z^{-4}.
\end{aligned} \tag{3.19}$$

Pour les constantes est les valeurs initiales

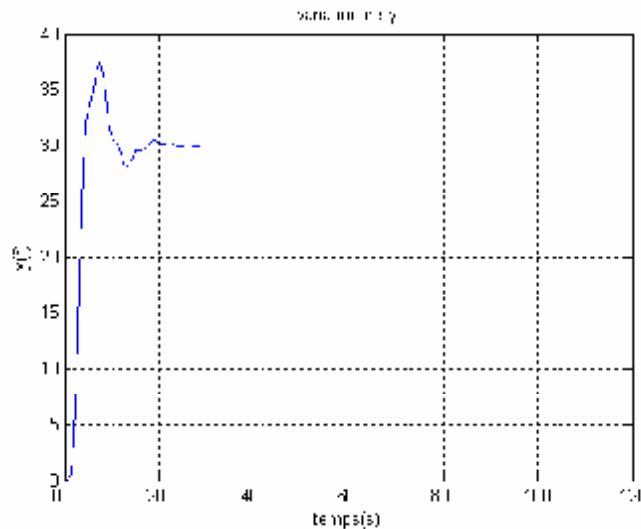
$$M = 1, \quad g = 9.8, \quad l = 2, \quad x_1(0) = x_2(0) = x_3(0) = x_4(0) = 0$$

$$J_1 = 4, \quad J_m = 2, \quad k = 100, \quad Te = 0.1s,$$

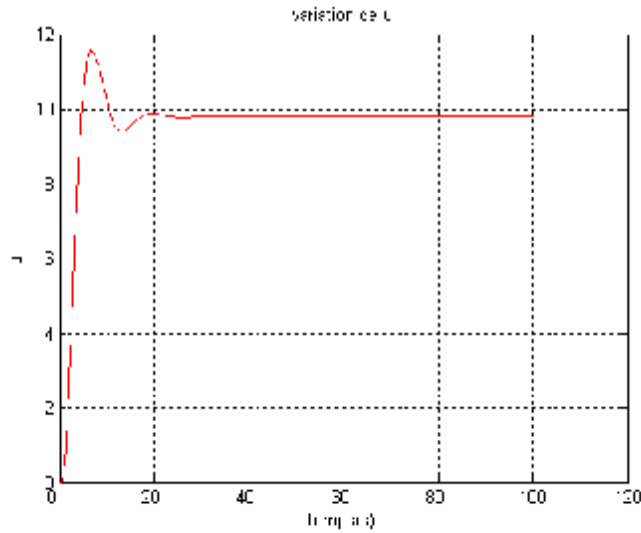
$$F_1 = 1, \quad F_m = 1, \quad z_1(0) = z_2(0) = z_3(0) = z_4(0) = 0,$$

$$\lambda = 1, \quad N_u = 1, \quad N_2 = 20,$$

On obtient les résultats de simulation donnés par les figure 3.3 et 3.4: ou on remarque que l'angle présente un dépassement relativement important dû à l'effet des non linéarités.



Figure(3.3) La variation de l'angle y



Figure(3.4) La variation de la commande  $u$

Dans le cas où le système non-linéaire est sous contraintes en utilisant la GPC avec contraintes. La procédure de commande sous contrainte est la suivante. Nous avons :

$$\begin{aligned} \Delta u_{\min} < \Delta u < \Delta u_{\max} \\ u_{\min} < u < u_{\max} \end{aligned} \quad (3.20)$$

Les états du système sont bornés, donc les états de retour d'état  $z_1, \dots, z_n$  sont bornés,

Donc on peut écrire que :

$$\begin{aligned} \Delta v_{\min} < \Delta v < \Delta v_{\max} \\ v_{\min} < v < v_{\max} \end{aligned} \quad (3.21)$$

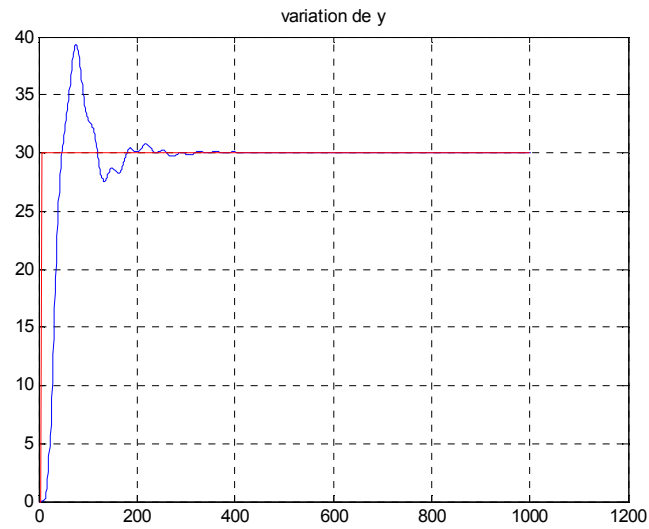
La technique de commande est basée sur l'algorithme suivant

- Calcul des matrices constantes de GPC
- On fait une simulation pour calculer les valeurs max des équations (3.21) compatible et équivalentes aux équations (3.20),
- On applique la fonction **QP** de Matlab

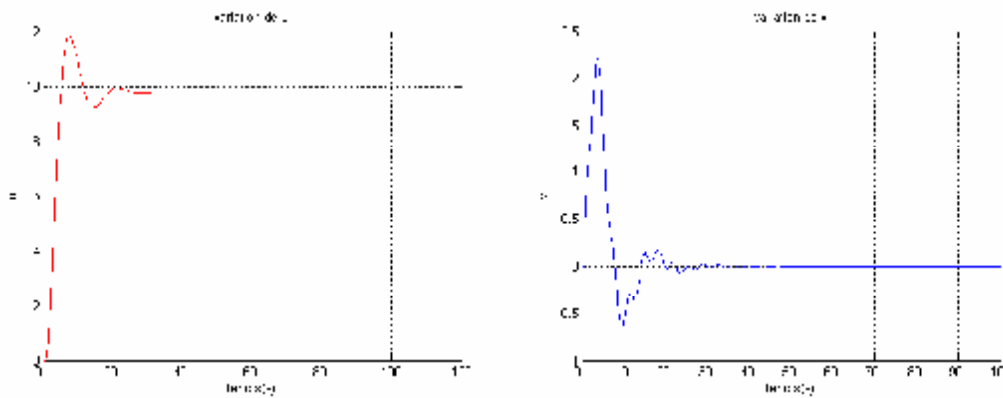
Dans la même application on prend les mêmes valeurs et les conditions initiales de l'exemple précédent avec :

$$\begin{aligned} \Delta u_{\max} = 0.32 & \quad \Delta v_{\max} = 0.4 \\ \Delta u_{\min} = -0.07 & \quad \Rightarrow \quad \Delta v_{\min} = -0.1 \\ u_{\min} = -4 \quad u_{\max} = 12 \quad v_{\max} = 2.2 \quad v_{\min} = -0.7 \\ N_u = 1, \quad N_2 = 20, \quad \lambda = 6, \end{aligned} \quad (3.22)$$

Les résultats de simulation sont donnés par les figures 3.5, 3.6 et 3.7. L'effet des non linéarités est assez apparent sur la sortie  $y$ , et les commandes et leurs incréments satisfont les contraintes.



Figure(3.5) : La variation de l'angle  $y$



Figure(3.6) : La variation de  $v$  et de  $u$

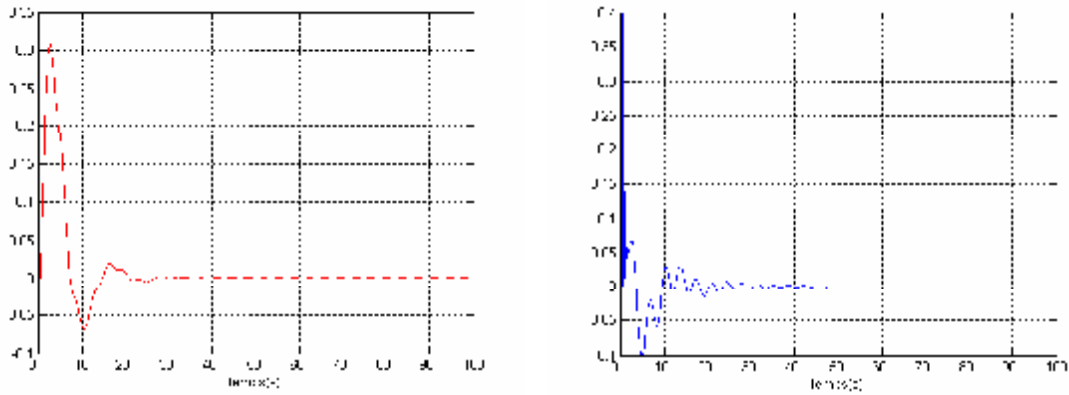


Figure (3.7) : La variation de  $\Delta v$  et de  $\Delta u$

#### IV.1.2 Application de la commande prédictive MBPC :

Dans ce cas on utilise le même exemple d'application, le résultat de linéarisation par retour d'état est donné par (3.16). On peut écrire

$$\begin{aligned} \dot{z} &= Az + Bv \\ y &= Cz \end{aligned} \quad (3.23)$$

La discrétisation de système (3.23) pour un temps d'échantillonnage  $T_e = 0.1s$  donne le résultat suivant

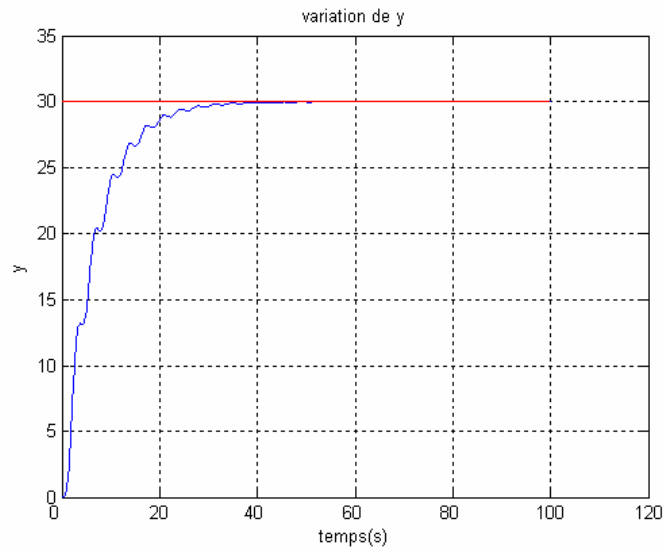
$$\begin{aligned} z(k+1) &= A_z z(k) + B_z v(k) \\ y(k) &= C z(k) \end{aligned} \quad (3.24)$$

Avec :

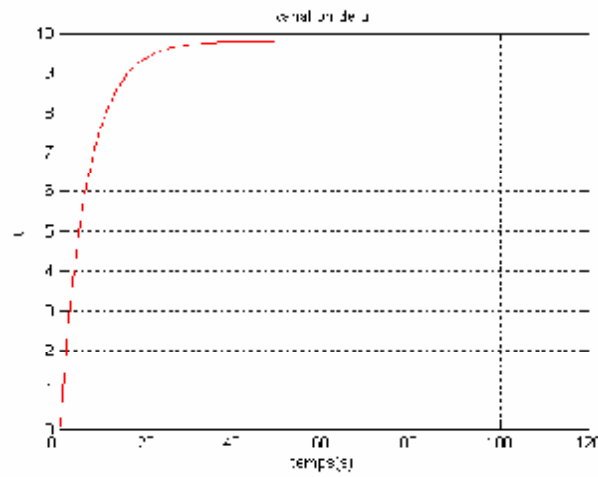
$$A_z = \begin{bmatrix} 1 & 0.1 & 0.005 & 0.0002 \\ 0 & 1 & 0.1 & 0.005 \\ 0 & 0 & 1 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B_z = \begin{bmatrix} 0 \\ 0.0002 \\ 0.005 \\ 0.1 \end{bmatrix} \quad (3.25)$$

Pour les mêmes valeurs initiales avec :  $N_u = 2, N_H = 20, R = 1, Q = 1$  on obtient les résultats des simulations donnés par les figures 3.8 et 3.9. L'effet des non linéarités sur la sortie est caractérisé par les fluctuations de la réponse (figure 3.8).





Figure(3.8) : La variation de l'angle  $y(^{\circ})$



Figure(3.9) La variation de la commande u

Dans le cas où les systèmes non-linéaire est sous contraintes, on utilise la commande *MBPC* avec contraintes. Nous considérons le cas de la commande prédictive *MBPC* avec contraintes qui peuvent être écrites sous la forme :

$$E \begin{bmatrix} \Delta U(k) \\ I \end{bmatrix} \leq 0 \quad (3.26)$$

$$F \begin{bmatrix} U(k) \\ I \end{bmatrix} \leq 0 \quad (3.27)$$

$$G \begin{bmatrix} Z(k) \\ I \end{bmatrix} \leq 0 \quad (3.28)$$

Dans cet cas on cherche une relation entre les équation (3.26), (3.27) et (3.28) et les équations suivantes :

$$E_1 \begin{bmatrix} \Delta V(k) \\ 1 \end{bmatrix} \leq 0 \quad (3.29)$$

$$F_1 \begin{bmatrix} V(k) \\ 1 \end{bmatrix} \leq 0 \quad (3.30)$$

$$G_1 \begin{bmatrix} Z(k) \\ 1 \end{bmatrix} \leq 0 \quad (3.31)$$

Où  $V(k) = [\hat{v}(k|k)^T, \dots, \hat{v}(k+N_u-1|k)^T]^T$  est défini de la même manière que  $\Delta V(k)$ .

La technique de commande est basée sur l'algorithme suivant

- d) Calcul des matrices constantes de MBPC
- e) On fait une simulation pour calculer une relation entre  $f, w$  des équations (3.26), (3.27) et (3.28), et les valeurs  $f_1, w_1$  les équations (3.29)....,
- f) On applique la fonction **QP** du Matlab,

Dans la même application on prend les mêmes valeurs et les conditions initiales que l'exemple précédent avec :

$$\text{pour } N_u = 2 \Rightarrow f = \begin{bmatrix} -0.6 \\ -0.6 \\ 0.6 \\ 0.6 \end{bmatrix}, \quad w = \begin{bmatrix} 0.14 \\ 0.14 \\ -0.14 \\ -0.14 \end{bmatrix} \text{ et } g = \begin{bmatrix} -0.7 \\ -0.7 \\ 0.7 \\ 0.7 \end{bmatrix}$$

$$\text{donc : } f_1 = \begin{bmatrix} -10 \\ -10 \\ 10 \\ 10 \end{bmatrix}, \quad w_1 = \begin{bmatrix} 0.2 \\ 0.2 \\ -0.2 \\ -0.2 \end{bmatrix}. \quad N_H = 20, \quad R = 1, Q = 1.$$

Les figures 3.10, 3.11 et 3.12 donnent les résultats de simulation. L'effet des nonlinéarités est encore apparent sur la sortie et on remarque que les commandes et leurs incréments satisfont les contraintes.

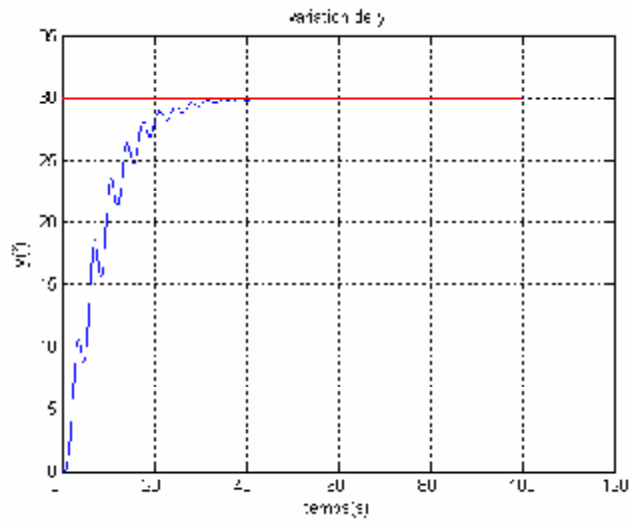
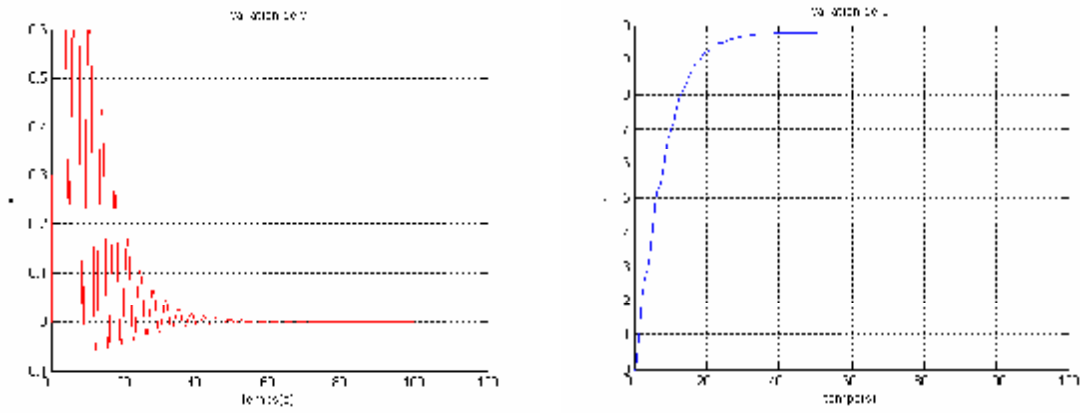
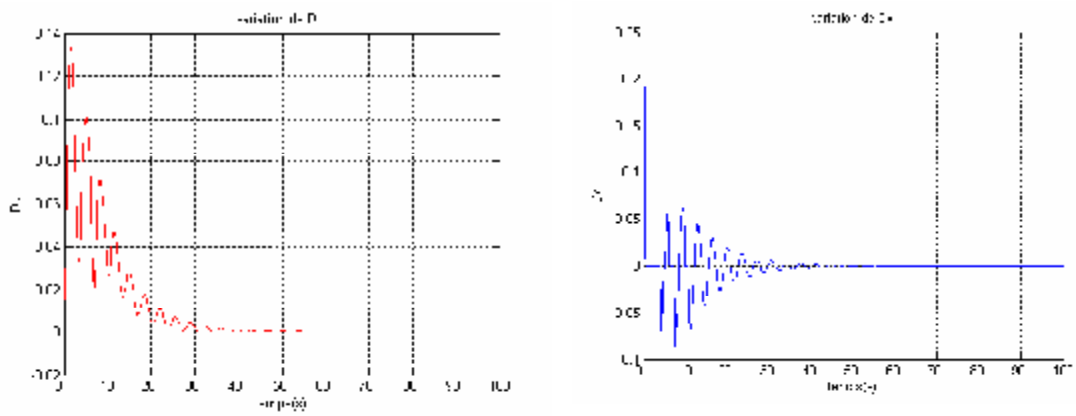


Figure (3.10) : La variation de la sortie  $y(^{\circ})$



Figure(3.11) : La variation du retour  $v$  et la commande  $u$



Figure(3.12) : La variation de la retour  $\Delta u$  et la commande  $\Delta v$

#### IV.2 Exemple 2 Commande d'un moteur synchrone :

Les équations dynamiques d'un moteur synchrone à aimant permanent avec la distribution sinusoïdale de flux sont données par:

$$\begin{aligned}\frac{\partial \delta}{\partial t} &= w \\ \frac{\partial w}{\partial t} &= -\frac{K_m}{J} i_a \sin(p\delta) + \frac{K_m}{J} i_b \cos(p\delta) - \frac{F}{J} w - \frac{T_L}{J} \\ \frac{\partial i_a}{\partial t} &= -\frac{R}{L} i_a + \frac{K_m}{L} w \sin(p\delta) + \frac{u_a}{L} \\ \frac{\partial i_b}{\partial t} &= -\frac{R}{L} i_b - \frac{K_m}{L} w \cos(p\delta) + \frac{u_b}{L}\end{aligned}\quad (3.32)$$

Dans lesquels  $\delta$  et  $w$  sont la position et la vitesse de rotor, et  $(i_a, i_b)$  et  $(u_a, u_b)$  courants et tensions de stator exprimées en armature fixe stator. Les paramètres du moteur sont le  $R$  résistance d'enroulement de redresseur, et self-inductance  $L$ , le constant de couple de moteur  $K_m$ , l'inertie de rotor  $J$ , le constant de frottement  $F$  et le nombre des pôles paires  $P$ . Tous les paramètres, sauf  $T_L$ , sont positifs. On suppose que  $P = 1$ . L'objectif est de commander la vitesse et de minimiser l'énergie.

On définit la fonction  $h = \delta$ , avec  $z = T(x)$  et un difféomorphisme local est défini :

$$z = T(x) = [z_1, z_2, z_3, z_4] \quad (3.33)$$

Donc on obtient:

$$\begin{aligned}z_1 &= \delta \\ z_2 &= w \\ z_3 &= -\frac{K_m}{J} i_a \sin(\delta) + \frac{K_m}{J} i_b \cos(\delta) - \frac{F}{J} w - \frac{T_L}{J} \\ z_4 &= K_m i_a \cos(\delta) + K_m i_b \sin(\delta)\end{aligned}\quad (3.34)$$

la variable  $z_4$  représente l'énergie consommée. Les sorties à commander sont données par :  $y=w$  ;  $y=z_4$ . Le modèle est réécrit sous la forme suivant :

$$\begin{aligned}\dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= v_1 \\ \dot{z}_4 &= v_2\end{aligned}\quad \text{Avec} \quad \frac{K_m}{L} \begin{bmatrix} -\frac{\sin(\delta)}{J} & \frac{\cos(\delta)}{J} \\ \cos(\delta) & \sin(\delta) \end{bmatrix} \begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} \phi_a \\ \phi_b \end{bmatrix} \quad (3.35)$$

### Application de la commande prédictive MBPC :

Dans ce cas on utilise la notion de commande MBPC multi-variable, le résultat de linéarisation par retour d'état est donnée par (3.35). On peut écrire

$$\begin{aligned} \dot{z} &= A_z z + B_z v \\ y &= C z \end{aligned} \quad (3.36)$$

La discrétisation de système (3.23) pour un temps d'échantillonnage  $T_e = 0.001s$  donne le résultat suivant

$$\begin{aligned} z(k+1) &= A_z z(k) + B_z v(k) \\ y(k) &= C z(k) \end{aligned} \quad (3.37)$$

Avec :

$$\begin{aligned} A_z &= \begin{bmatrix} 1 & 0.001 & 0 & 0 \\ 0 & 1 & 0.001 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B_z = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.001 & 0 \\ 0 & 0.001 \end{bmatrix} \\ C &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (3.38)$$

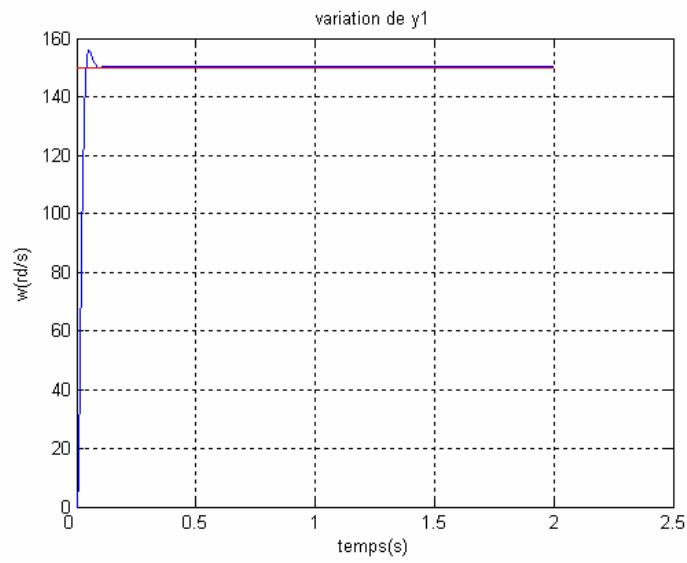
Pour les constantes et valeurs initiales suivantes :

$$N_u = 3, N_H = 55, R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 10000000 & 0 \\ 0 & 100000 \end{bmatrix},$$

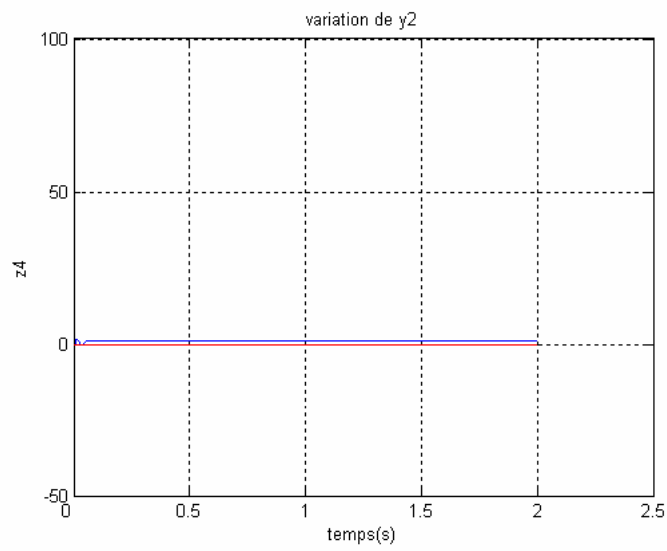
$$K_m = \frac{3}{2}, J = 0.03, T_L = 5, R = 1.93, F = 0.001, L = 0.04244, T_e = 0.001,$$

$$z_1(0) = z_2(0) = z_3(0) = z_4(0) = 0, \delta(0) = w(0) = i_a(0) = i_b(0) = 0,$$

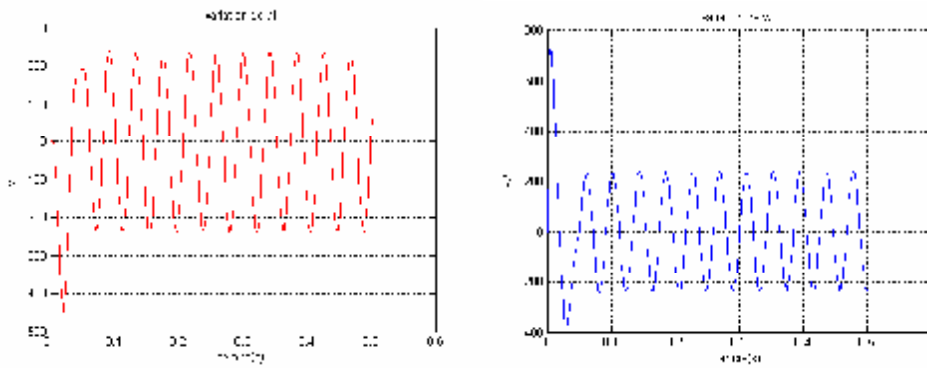
On obtient les résultats donnés par les figures 3.11, 3.12 et 3.13. La vitesse ne présente pas de dépassement important, l'énergie consommée est maîtrisée, les courants de commande sont sinusoïdaux.



Figure(3.11) : La variation de la sortie  $y_1$  (vitesse  $w$  en rad/s)



Figure(3.12) : La variation de la sortie  $y_2$  (maximum d'énergie)



Figure(3.13) : La variation de la retour  $u_a$  et la commande  $u_b$

Pour le cas des systèmes non-linéaires multi-variables, on utilise la méthode MBPC avec contraintes. De la même manière que l'exemple précédent, avec les mêmes valeurs des paramètres du système et les mêmes conditions initiales. On choisit :

$$N_u = 3, N_2 = 55, Q = \begin{bmatrix} 1000000 & 0 \\ 0 & 1000 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$y_{1\max} = 160, y_{1\min} = -160, y_{2\max} = 4, y_{2\min} = -4,$$

$$u_{a\max} = 330, u_{a\min} = -330, u_{b\max} = 260, u_{b\min} = -360,$$

$$\Delta u_{a\max} = 40, \Delta u_{a\min} = -40, \Delta u_{b\max} = 80, \Delta u_{b\min} = -40,$$

Les résultats de simulation sont donnés sur les figures 3.14, 3.15 et 3.16. On remarque que les contraintes sont satisfaites pour toutes les variables.

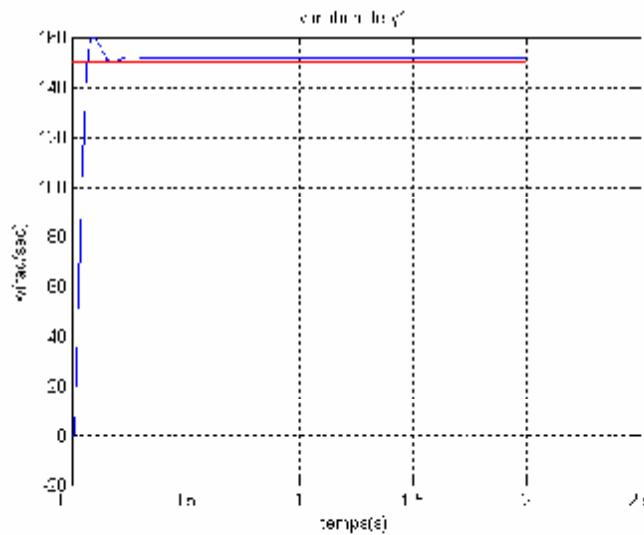


Figure (3.14) : La variation de la sortie  $y_1$  avec contrainte (vitesse  $w$  en rad/s)

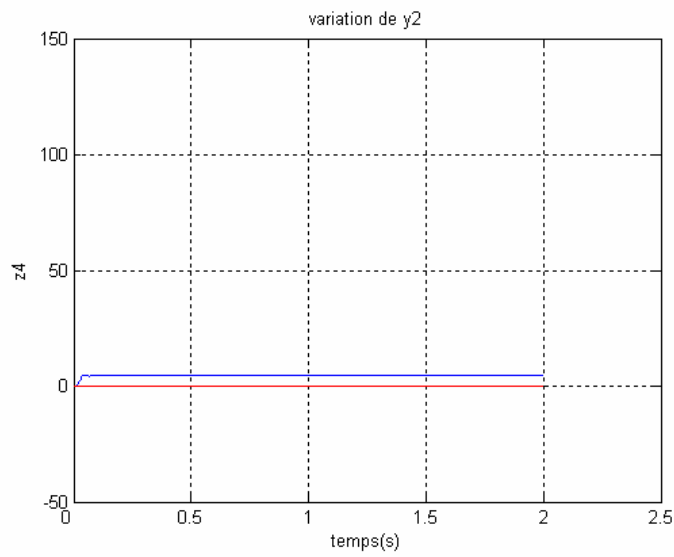


Figure (3.15) : La variation de la sortie  $y_2$  avec contrainte (maximum d'énergie)

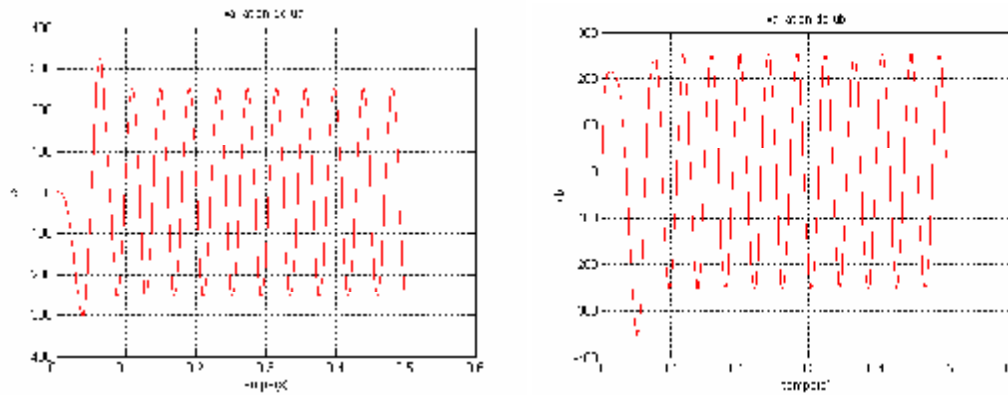


Figure (3.16) : La variation de  $u_a$  et la commande  $u_b$  avec contrainte



# CHAPITRE 4 :

## LA

### METHODE DU «BACKSTEPPING»

#### I. Introduction

Dans ce chapitre nous développons une étude de la méthode de commande backstepping. L'utilisation de cette technique exige une méthodologie systématique pour la conception. Les éléments théoriques de ce chapitre sont basés sur [2]. Plusieurs simulations seront présentées dans les cas adaptatif et non adaptatifs.

#### II Commande par backstepping

##### II.1 Approche non adaptative

###### II.1.1. Principe

Pour simplifier la présentation, nous commençons par développer un exemple de commande non adaptative (figure 3.1) par la technique backstepping dans le but d'atteindre la convergence des erreurs afin de réaliser la stabilité et l'équilibre  $y = x_1$  du système dont  $y_r$  est l'entrée de référence.

Soit le système:

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi(x_1)^T \theta \\ \dot{x}_2 &= u \\ y &= x_1 \end{aligned} \tag{4.1}$$

Tel que  $\theta$  : vecteur paramétrique connu,

$\varphi(x_1)$  : Vecteur de fonctions non linéaires dérivable, tel que  $\varphi(0) = 0$ .

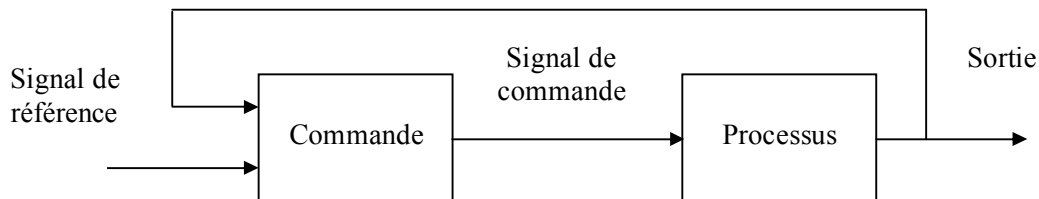


Figure 4.1 Schéma de principe de la commande non adaptative

La conception du backstepping est récursive. D'abord, on choisit l'état  $x_2$  comme étant la commande virtuelle de l'état  $x_1$  (voir figure 4.1), ensuite on adopte la fonction stabilisante suivante:

$$\alpha_1(x_1) = -c_1(x_1 - y_r) - \varphi(x_1)^T \cdot \theta \quad . c_1 > 0 \quad (4.2)$$

Cette solution est conçue pour stabiliser l'équation (4.1), en supposant que  $x_2 = \alpha_1(x_1)$  peut être implantée. Puisque ce n'est pas le cas, on définit:

$$\begin{aligned} z_1 &= x_1 - y_r \\ z_2 &= x_2 - \alpha(x_1) - \dot{y}_r \end{aligned} \quad (4.3)$$

$z_2$  est la variable qui exprime la réalité que «  $x_2$  n'est pas la commande exacte ». Alors, le système complet (4.1) peut être formulé en utilisant les nouvelles coordonnées  $z_1$  et  $z_2$ :

$$\begin{aligned} \dot{z}_1 &= -c_1 z_1 + z_2 \\ \dot{z}_2 &= u - \frac{\partial \alpha_1}{\partial x_1} (x_2 + \varphi(x_1)^T \cdot \theta) - \ddot{y}_r - \frac{\partial \alpha_1}{\partial y_r} \cdot \dot{y}_r \end{aligned} \quad (4.4)$$

Pour le système d'équations (4.4), on va concevoir une loi de commande  $u = \alpha(x_1, x_2)$  afin de rendre la dérivée de la fonction de Lyapunov définie positive:

$$V = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 \quad (4.5)$$

La dérivée de (4.5) le long de la trajectoire donne:

$$\begin{aligned} \dot{V} &= z_1 \cdot \dot{z}_1 + z_2 \cdot \dot{z}_2 \\ &= -c_1 z_1^2 + z_2 \left[ u + z_1 - \frac{\partial \alpha_1}{\partial x_1} (x_2 + \varphi(x_1)^T \cdot \theta) - \ddot{y}_r - \frac{\partial \alpha_1}{\partial y_r} \cdot \dot{y}_r \right] \end{aligned} \quad (4.6)$$

Le chemin évident pour réaliser et atteindre la négativité de  $\dot{V}$  est de choisir la commande  $u$  comme:

$$u = \alpha_2(x_1, x_2) = -c_2 z_2 - z_1 + \frac{\partial \alpha_1}{\partial x_1} (x_2 + \varphi(x_1)^T \cdot \theta) + \ddot{y}_r + \frac{\partial \alpha_1}{\partial y_r} \cdot \dot{y}_r \quad (4.7)$$

Alors:

$$\dot{V} = -c_1 z_1^2 - c_2 z_2^2 \leq 0 \quad (4.8)$$

Ce qui signifie que l'équilibre est globalement asymptotiquement stable. Le système en boucle fermée est stable:

$$\dot{Z} = AZ \quad (4.9)$$

Avec:

$$A = \begin{bmatrix} -c_1 & 1 \\ -1 & -c_2 \end{bmatrix}, \quad Z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

La solution dans ce cas se traduit par:

$$Z = Z(0) \exp(-At) \quad (4.10)$$

### II.1.2 Résultats de simulation :

On prend le même exemple (4.1) avec :

$$\varphi(x_1) = \begin{bmatrix} \sin(x_1) \\ x_1^3 \end{bmatrix}, \quad \theta = \begin{bmatrix} 10 \\ 7 \end{bmatrix}, \quad c_1 = 6, \quad c_2 = 1, \\ x_1(0) = 0, \quad x_2(0) = 0,$$

Les résultats de simulation sont donnés par la figure 4.2.

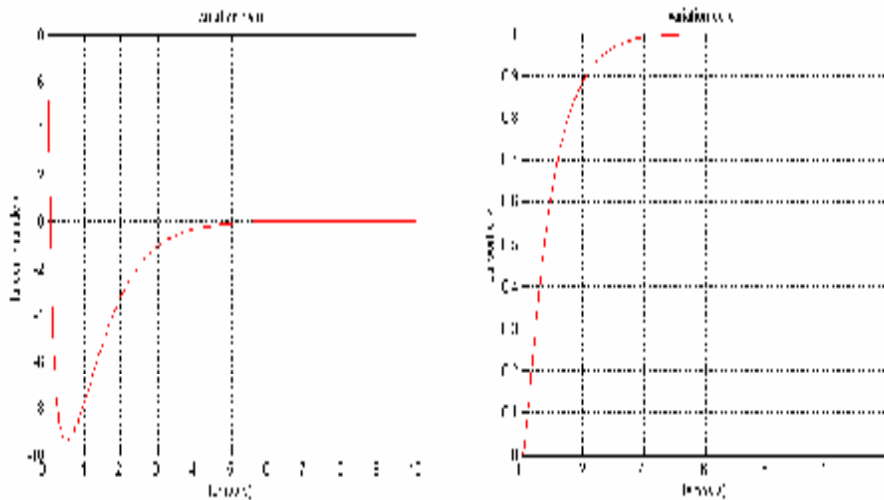


Figure 4.2. Résultat de simulation d'une commande backstrpping approche  
Non-adaptative

### II.2 Approche adaptative

Les modèles réels des systèmes physiques ne sont pas linéaires et habituellement caractérisés par des paramètres (masses, inductances, ..... ) qui sont peu connus ou dépendent d'un petit changement d'environnement. Si ces paramètres varient dans un intervalle important, il serait préférable d'employer une loi d'adaptation pour estimer les paramètres du système. La figure 4.3 donne le schéma de principe d'une commande adaptative.

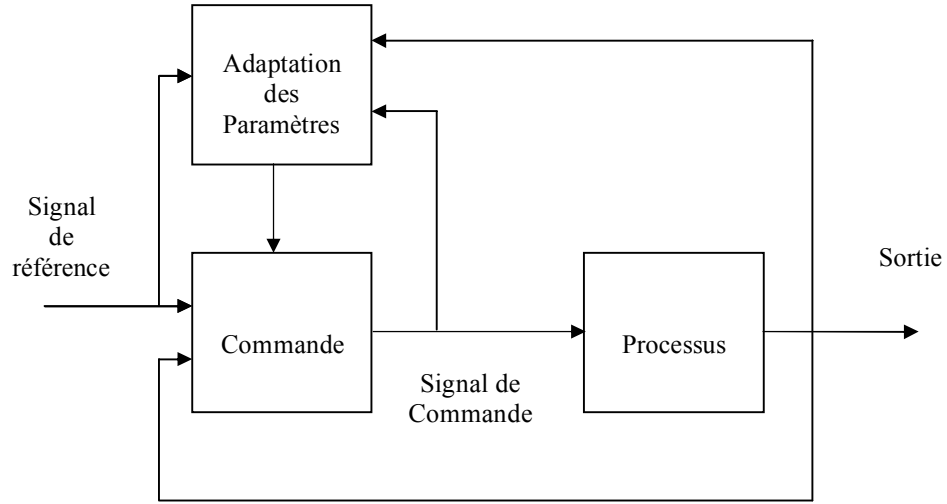


Figure 4.3 : Schéma de principe de la commande adaptative

### II.2.1 Conditions d'implantation

Comme la majorité des méthodes de commande des systèmes non linéaires, l'application de la technique backstepping est limitée à certaines classes de systèmes. Les systèmes dans ce cas doivent être sous une certaine forme triangulaire. La forme générale du système à analyser est donnée par :

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \varphi_1(x_1)^T \cdot \theta \\
 \dot{x}_2 &= x_3 + \varphi_2(x_1, x_2)^T \cdot \theta \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 \dot{x}_{n-1} &= x_n + \varphi_{n-1}(x_1, \dots, x_{n-1})^T \cdot \theta \\
 \dot{x}_n &= \beta(x)u + \varphi_n(x)^T \cdot \theta \\
 y &= x_1
 \end{aligned} \tag{4.11}$$

Où chaque  $\varphi_i : R^i \mapsto R^p$  : est un vecteur de fonctions non linéaires, et  $\theta \in R^p$  est un vecteur de coefficients constants. La commande  $u$  est multipliée par la fonction  $\beta(x)$ , avec  $\beta(x) \neq 0, \forall x \in R^n$ . Si le but est d'atteindre la trajectoire désirée  $y_r$  en utilisant l'état  $x_i$ , alors l'algorithme du backstepping peut être utilisé pour la stabilisation globale asymptotique de l'erreur du système (on note l'erreur primaire par  $z \in R^n$ ). Puisque le vecteur  $\theta$  est inconnu, alors avec une augmentation du système par la dynamique de l'estimateur  $\hat{\theta}$ , une version algorithmique adaptative du backstepping est utilisée dans le

but d'avoir une stabilité globale et asymptotique de l'erreur du système. En général, l'algorithme de la commande adaptative backstepping peut être utilisé pour atteindre la stabilité globale et asymptotique de l'erreur du système si les états et les conditions suivantes sont respectées :

- Le système est introduit selon la forme (4.11) ;
- Les fonctions non linéaires  $\varphi_i$  sont connues ;
- La paramétrisation est linéaire ;
- La fonction  $\beta(x)$  satisfait la condition,  $\beta(x) \neq 0, \forall x \in R^n$  ;
- Chaque  $\varphi_i$  est continûment dérivable ;
- Le signal de référence  $y_r$  est continu ;
- Tous les états sont mesurables.

Le diagramme, présenté par la figure 4.4, présente un exemple d'ordre trois avec  $\beta(x)=1$  et les fonctions non linéaires dépendent seulement des variables d'état.

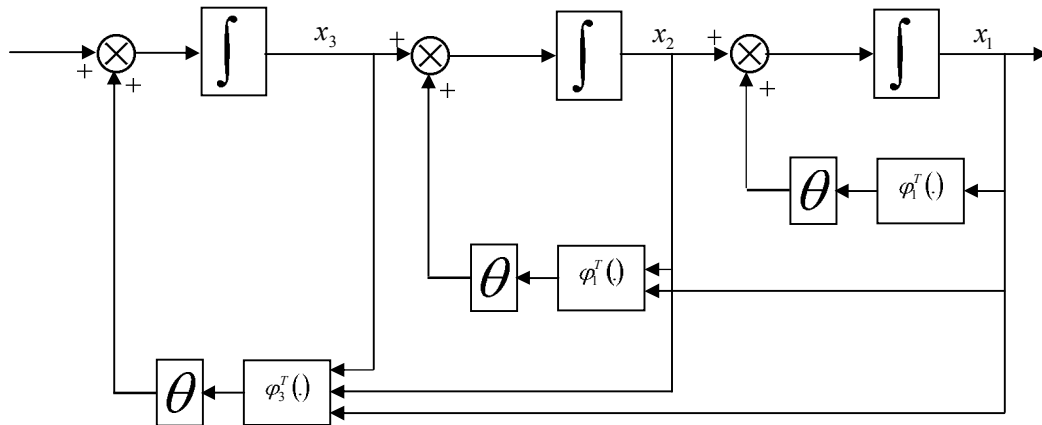


Figure 4.4 : Schéma du système d'ordre trois

Dans ce qui suit, on va présenter l'étude d'un système non linéaire et l'implantation du backstepping sera introduite afin de réaliser une commande adaptative. Le problème d'adaptation surgit à cause du vecteur paramétrique inconnu  $\theta$  et le vecteur de fonctions non-linéaires  $\varphi(x_1)$  est connu avec  $\varphi(0) = 0$ .

Soit le système :

$$\begin{aligned}
 \dot{x} &= x_2 + \varphi(x_1)^T \cdot \theta \\
 \dot{x}_2 &= u \\
 y &= x_1
 \end{aligned}
 \tag{4.12}$$

Pour concevoir une commande adaptative dans cette partie, on remplace le vecteur de paramètres réels  $\theta$  par son estimation  $\hat{\theta}$  dans la fonction de stabilisation (équations 4.2 et 4.3), ce qui donne :

$$\begin{aligned} z_1 &= x_1 - y_r \\ z_2 &= x_2 - \alpha_1(x_1, \theta) - \dot{y}_r \\ \alpha_1(x_1, \theta) &= -c_1 z_1 - \varphi^T \hat{\theta} \end{aligned} \quad (4.13)$$

Dans ce cas, la loi de commande donnée par (4.7) va être renforcée par le terme  $v_2(x_1, x_2, \theta)$  qui va compenser les transitions des paramètres estimés.

$$u = \alpha_2(x_1, x_2, \hat{\theta}) = -c_2 \cdot z_2 - z_1 + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \varphi^T \cdot \hat{\theta}) + v_2(x_1, x_2, \hat{\theta}) + \ddot{y}_r + \frac{\partial \alpha_1}{\partial x_1} \cdot \dot{y}_r \quad (4.14)$$

Le système résultant dans les coordonnées  $z$  est :

$$\begin{aligned} \dot{z}_1 &= z_2 + \alpha_1 + \varphi^T \cdot \theta = -c_1 z_1 + z_2 + \varphi^T \cdot \tilde{\theta} \\ \dot{z}_2 &= \dot{x}_2 - \alpha_1 = u - \frac{\partial \alpha_1}{\partial x_1}(x_2 + \varphi^T \cdot \theta) - \frac{\partial \alpha_1}{\partial \hat{\theta}} \hat{\theta} - \ddot{y}_r - \frac{\partial \alpha_1}{\partial y_r} \cdot \dot{y}_r \\ &= -c_2 z_2 - z_1 - \frac{\partial \alpha_1}{\partial x_1} \varphi^T \cdot \tilde{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}} \hat{\theta} + v_2(x_1, x_2, \hat{\theta}) - \ddot{y}_r - \frac{\partial \alpha_1}{\partial y_r} \cdot \dot{y}_r \end{aligned} \quad (4.15)$$

Avec:  $\tilde{\theta} = \theta - \hat{\theta}$

En tenant compte de l'équation (4.15), le terme compensateur est choisi comme suit:

$$v_2(x_1, x_2, \hat{\theta}) = \frac{\partial \alpha_1}{\partial \hat{\theta}} \hat{\theta} \quad (4.16)$$

Si l'erreur  $\tilde{\theta}$  est nulle, le système devient asymptotiquement linéaire et stable (équation 4.10). Puisque ce n'est pas le cas, la tâche suivante consiste à choisir la loi de mise à jour  $\dot{\hat{\theta}} = \Gamma \tau_2(x, \hat{\theta})$

Considérons la fonction de Lyapunov:

$$V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 + \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta} \quad (4.17)$$

Puisque  $\dot{\hat{\theta}} = -\dot{\theta}$ , la dérivé de  $V_2$  s'écrit:

$$\begin{aligned} \dot{V}_2 &= -c_1 z_1^2 - c_2 z_2^2 + [z_1 \quad z_2] \cdot \begin{bmatrix} \varphi^T \\ -\frac{\partial \alpha_1}{\partial x_1} \varphi^T \end{bmatrix} \cdot \tilde{\theta} - \tilde{\theta}^T \Gamma^{-1} \dot{\hat{\theta}} \\ &= -c_1 z_1^2 - c_2 z_2^2 + \tilde{\theta} \left( \Gamma^{-1} \begin{bmatrix} \varphi & -\frac{\partial \alpha_1}{\partial x_1} \varphi \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} - \dot{\hat{\theta}} \right) \end{aligned} \quad (4.18)$$

La seule solution pour éliminer l'erreur paramétrique  $\tilde{\theta}$  est de choisir la loi de mise jour suivante :

$$\dot{\hat{\theta}} = \Gamma \tau_2(x, \hat{\theta}) = \Gamma \begin{bmatrix} \varphi & -\frac{\partial \alpha_1}{\partial x_1} \varphi \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (4.19)$$

ce qui permet d'écrire les expressions suivantes :

$$\begin{aligned} \tau_1(x_1) &= \varphi \cdot z_1 \\ \tau_2(x_1, x_2, \hat{\theta}) &= \tau_1(x_1) - \frac{\partial \alpha_1}{\partial x_1} \varphi \cdot z_2 \end{aligned} \quad (4.20)$$

Alors,  $\dot{V}_2$  est négative et la stabilité globale de  $z=0$  est réalisée.

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 \leq 0 \quad (4.21)$$

Enfin, il résulte que l'équilibre est globalement stable et  $\lim_{t \rightarrow \infty} x_1(t) = y_r$

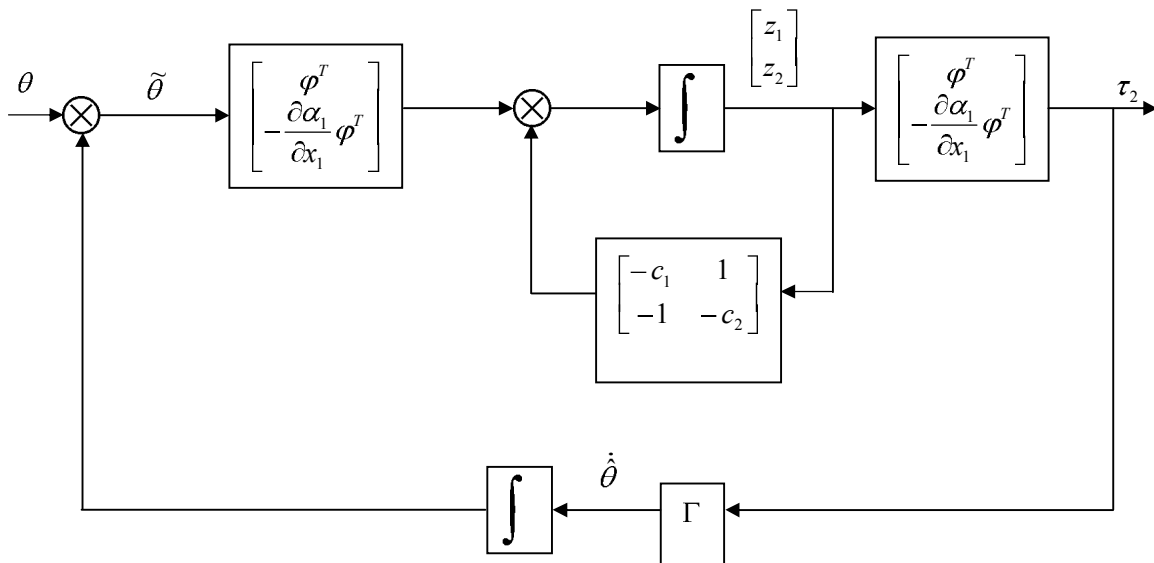


Figure 4.5 : Commande adaptative du système bouclé

### II.2.2 Résultats de Simulation

On prend le même exemple (4.1) avec :

$$\varphi(x_1) = \begin{bmatrix} \sin(x_1) \\ x_1^2 \end{bmatrix}, \quad \theta = \begin{bmatrix} 20 \\ 10 \end{bmatrix}, \quad c_1 = 50, \quad c_2 = 150, \quad y_r = 1,$$

$$x_1(0) = 0, \quad x_2(0) = 0, \quad T_e = 0.001, \quad \hat{x}_1(0) = 0, \quad \hat{x}_2(0) = 0,$$

$$\Gamma = \begin{bmatrix} 0.5 & 0.004 \\ 0.004 & 0.11 \end{bmatrix}, \quad \hat{\theta}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

Les résultats de simulation donnés par les figures 4.6, 4.7, 4.8 et 4.9. On remarque que la sortie suit bien la référence avec un dépassement assez important caractéristique d'une commande adaptative non linéaire. En effet durant les premières itérations le système est en phase d'apprentissage. Les paramètres convergent rapidement vers un état stationnaire,  $\hat{\theta}_1$  convergent vers la vraie valeur  $\theta_1$  mais  $\hat{\theta}_2$  présente un léger offset.

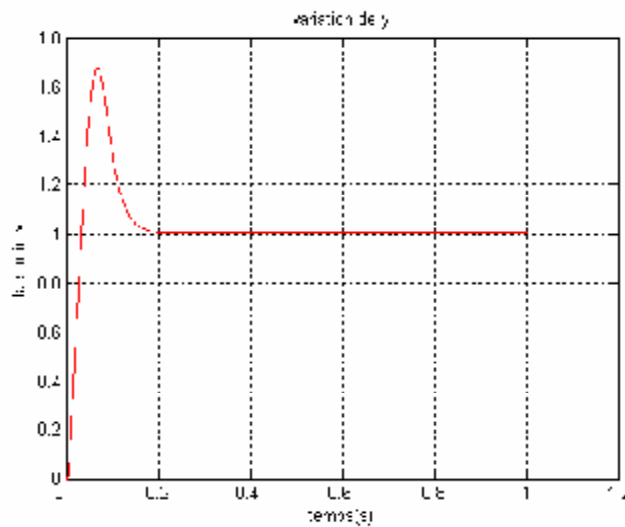


Figure 4.6 : la sortie du système



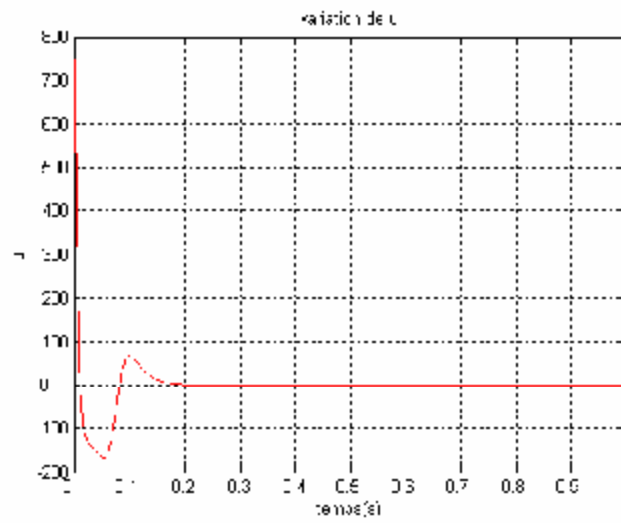


Figure 4.7 : la commande de système

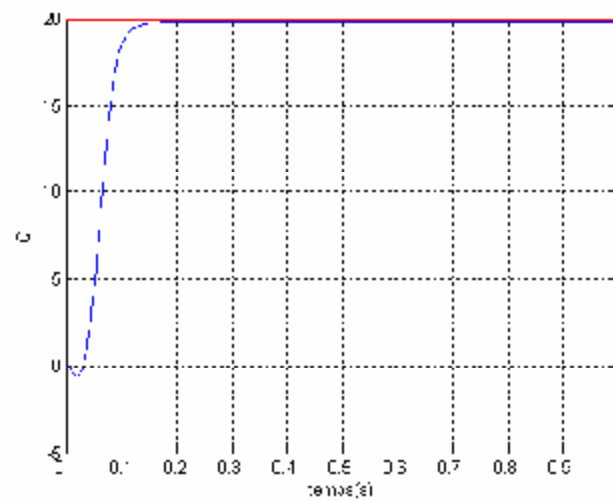


Figure 4.8 : variation de  $\hat{\theta}_1$  par rapport a  $\theta_1$

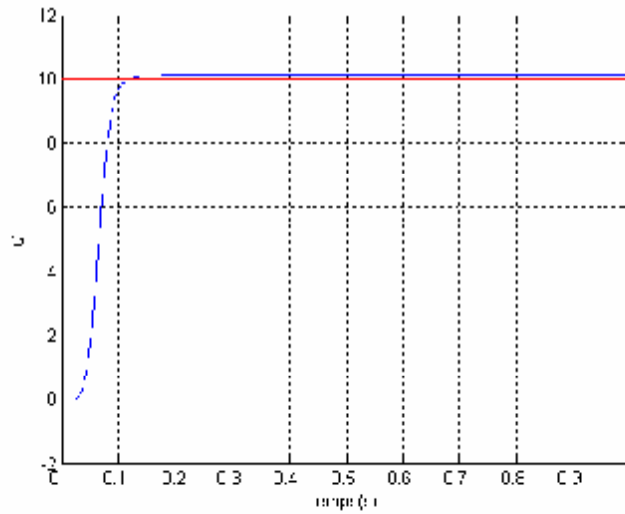


Figure 4.9 : variation de  $\hat{\theta}_2$  par rapport a  $\theta_2$

## II.3 Généralisation

### II. 3.1 Théorie

Afin de donner l'algorithme de la commande adaptative « backstepping » pour certaines classes de systèmes non-linéaires, le schéma (4.10) représente la procédure globale de cette technique. La forme générale du système est donnée par :

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \varphi_1(x_1)^T \theta \\
 \dot{x}_2 &= x_3 + \varphi_2(x_1, x_2)^T \theta \\
 &\cdot \\
 &\cdot \\
 &\cdot \\
 \dot{x}_{n-1} &= x_n + \varphi_{n-1}(x_1, \dots, x_{n-1})^T \theta \\
 \dot{x}_n &= \beta(x)u + \varphi_n(x)^T \theta \\
 y &= x_1,
 \end{aligned} \tag{4.22}$$

Telles que  $\beta(x) \neq 0 \quad \forall x \in \mathbb{R}^n$  et  $F(x) = [\varphi_1(x_1), \varphi_2(x_2), \dots, \varphi_n(x_n)]$  vecteur de fonctions non linéaires lisses.

- Algorithme de la procédure backstepping

Par convention, on définit  $z_0 \stackrel{\Delta}{=} 0, \alpha_0 \stackrel{\Delta}{=} 0, \tau_0 \stackrel{\Delta}{=} 0$

$$z_i = x_i - y_r^{i-1} - \alpha_{i-1}$$

$$\alpha_i(\bar{x}_i, \hat{\theta}, y_r^{i-1}) = -z_{i-1} - c_i z_i - w_i^T \cdot \hat{\theta} + \sum_{k=1}^{i-1} \left( \frac{\partial \alpha_{i-1}}{\partial x_k} x_{k+1} + \frac{\partial \alpha_{i-1}}{\partial y_r^{k-1}} y_r^{(k)} \right) + v_i$$

$$v_i(\bar{x}_i, \hat{\theta}, y_r^{i-1}) = \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \Gamma \tau_i + \sum_{k=2}^{i-1} \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \Gamma w_i z_k$$

$$\tau_i(\bar{x}_i, \hat{\theta}, \bar{y}_r^{(i-1)}) = \tau_{i-1} + w_i z_i$$

$$w_i(\bar{x}_i, \hat{\theta}, \bar{y}_r^{(i-2)}) = \varphi_i - \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k} \varphi_k$$

$$i = 1, \dots, n$$

$$\bar{x}_i = (x_1, x_2, \dots, x_i), \quad \bar{y}_r^{(i)} = (y_r, \dot{y}_r, \ddot{y}_r, \dots, y_r^{(i)}) \quad \text{connues}$$

La loi de commande adaptative:

$$u = \frac{1}{\beta(x)} \left[ \alpha_n(x, \hat{\theta}, \bar{y}_r^{(n-1)}) + y_r^{(n)} \right]$$

La loi de mise à jour:

$$\dot{\hat{\theta}} = \Gamma \tau_n(x, \hat{\theta}, \bar{y}_r^{(n-1)}) = \Gamma w \cdot z$$

Le système bouclé aura la forme:

$$\dot{z} = A_z(z, \hat{\theta}, t)z + w(z, \hat{\theta}, t)^T \tilde{\theta}$$

$$\dot{\hat{\theta}} = \Gamma w(z, \hat{\theta}, t)$$

Tel que:

$$A_z(z, \hat{\theta}, t) = \begin{bmatrix} -c_1 & 1 & 0 & \dots & 0 \\ -1 & -c_2 & 1 + \sigma_{23} & \dots & \sigma_{2n} \\ 0 & -1 - \sigma_{23} & \cdot & \dots & \cdot \\ \cdot & \cdot & \cdot & \cdot & 1 + \sigma_{n-1,n} \\ 0 & -\sigma_{2n} & \dots & -1 - \sigma_{n-1,n} & -c_n \end{bmatrix}$$

et

$$\sigma_{ij}(x, \hat{\theta}) = -\frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \Gamma w_k$$

La fonction de Lyapunov s'exprime par:

$$V_n = \frac{1}{2} Z^T Z + \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta}$$

La condition de stabilité est sous forme:

$$\dot{V}_n = -\sum_{k=1} C_k Z_k^2$$

L'équilibre du système s'exprime par:  $Z = 0, \lim_{t \rightarrow \infty} Z(t) = 0, \lim_{t \rightarrow \infty} [y(t) - y_r(t)] = 0.$

#### II.4.2 Résultats de simulation Application à un robot avec le joint flexible

Les équations dynamiques d'un bras de robot de lien avec un joint élastique tournant dans un plan vertical sont données par :

$$\begin{aligned} J_1 \ddot{q}_1 + F_1 \dot{q}_1 + k(q_1 - q_2) + Mgl \sin q_1 &= 0 \\ J_m \ddot{q}_2 + F_m \dot{q}_2 - k(q_1 - q_2) &= u \\ y &= q_1 \end{aligned} \quad (4.23)$$

Dans lesquels  $q_1$  et  $q_2$  sont le déplacement (ou la rotation) du lien et le déplacement de rotor, respectivement. L'inertie du lien  $J_1$ , l'inertie de rotor de moteur  $J_m$ , la constante élastique  $k$ , le masse de lien  $M$ , le constant de gravité  $g$ , le centre de masse  $l$  et les coefficients visqueux de frottement  $F_1, F_m$  sont des paramètres positifs. La commande  $u$  est le couple fourni par le moteur. On choisit comme variables d'états  $x_1, x_2, x_3, x_4$  telles que :

$$x_1 = q_1, \quad x_2 = \dot{q}_1, \quad x_3 = q_2, \quad x_4 = \dot{q}_2$$

Le modèle est réécrit sous la forme suivante :

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\left(\frac{F_1}{J_1}\right)x_2 - \left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -\left(\frac{F_m}{J_m}\right)x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) + \left(\frac{1}{J_m}\right)u \end{aligned} \quad (4.24)$$

Avec les paramètres  $F_1, F_m$  sont des paramètres inconnus, donc on peut écrire le système sous la forme suivante :

$$\dot{x} = f(x) + g(x)u + \sum_{i=1}^p \theta_i q_i(x) \quad (4.25)$$

Avec

$$f(x) = \begin{bmatrix} x_2 \\ -\left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \\ x_4 \\ \left(\frac{k}{J_m}\right)(x_1 - x_3) \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix}, \quad \theta = \begin{bmatrix} \frac{F_1}{J_1} \\ \frac{F_m}{J_m} \end{bmatrix} \quad (4.26)$$

$$q(x) = \begin{bmatrix} 0 \\ -x_2 \\ 0 \\ -x_4 \end{bmatrix}$$

Pour rendre le système (4.24) sous la forme (4.22) il faut vérifier les conditions suivantes :

- i) Le système nominal  $(f, g)$  est localement (globalement) linéarisable par retour d'état,
- ii) La supposition stricte de triangularité

$$ad_q G_i \subset G_i, \quad 0 \leq i \leq n-2$$

est satisfaite dans  $U_0$ , voisinage d'origine (dans  $R^n$ );

- i) On définit la fonction  $h(x) = z_1 = x_1$

$$g = \frac{1}{J_m} \frac{\partial}{\partial x_4},$$

$$f = x_2 \frac{\partial}{\partial x_1} + \left( -\left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \right) \frac{\partial}{\partial x_2}$$

$$+ x_4 \frac{\partial}{\partial x_3} + \left( \left(\frac{k}{J_m}\right)(x_1 - x_3) \right) \frac{\partial}{\partial x_4}, \quad (4.27)$$

$$q = -x_2 \frac{\partial}{\partial x_2} - x_4 \frac{\partial}{\partial x_4}$$

$$ad_f^1 g = L_f L_g - L_g L_f = -\frac{1}{J_m} \frac{\partial}{\partial x_3},$$

$$ad_f^2 g = L_f L_{ad_f^1 g} - L_{ad_f^1 g} L_f = \frac{1}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_2} - \frac{k}{J_m} \frac{\partial}{\partial x_4} \right), \quad (4.28)$$

$$ad_f^3 g = L_f L_{ad_f^2 g} - L_{ad_f^2 g} L_f = -\frac{1}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_2} - \frac{k}{J_m} \frac{\partial}{\partial x_3} \right),$$

Donc  $span\{g, \dots, ad_f^3 g\} = R^4$ .

La distribution  $G_2 = span\{g, \dots, ad_f^2 g\}$  est involutive et a un rang constant égal à 3,

$$\begin{aligned} [g, ad_f g] &= L_g L_{ad_f g} - L_{ad_f g} L_g = 0 \in G_2 = span\{g, \dots, ad_f^2 g\} \\ \text{Donc : } [g, ad_f^2 g] &= L_g L_{ad_f^2 g} - L_{ad_f^2 g} L_g = 0 \in G_2 = span\{g, \dots, ad_f^2 g\} \\ \text{aussi } [ad_f g, ad_f^2 g] &= 0 \in G_2 = span\{g, \dots, ad_f^2 g\}, \end{aligned}$$

On a donc la distribution  $G_2 = span\{g, \dots, ad_f^2 g\}$  est involutive et a un rang constant égal à 3,

$$ii) \text{ la 2eme supposition est : } ad_q G_i \subset G_i, \quad 0 \leq i \leq 2$$

$$\text{Avec: } G_0 = span\{g\} \quad G_1 = span\{g, ad_f^1 g\}.$$

Cette condition est satisfaite parce que :

$$ad_q G_i = 0 \subset G_i, \quad 0 \leq i \leq 1$$

$ad_q G_2$  se calcule comme suit

$$ad_q g = 0, \quad ad_q ad_f g = 0, \quad \text{et} \quad ad_q ad_f^2 g = ad_f^2 g \subset G_2$$

Les deux conditions sont vraies donc on peut faire la transformation de système sur la nouvelle forme qui est donnée par la formule (4.11) avec :

$$\begin{aligned} z_1 &= h(x) = x_1 \\ z_2 &= L_f h(x) = x_2 \\ z_3 &= L_f^2 h(x) = \left( -\left( \frac{Mgl}{J_1} \right) \sin(x_1) - \left( \frac{k}{J_1} \right) (x_1 - x_3) \right) \\ z_4 &= L_f^3 h(x) = -\left( \frac{Mgl}{J_1} \right) x_2 \cos(x_1) + \left( \frac{k}{J_1} \right) (x_4 - x_2) \\ \varphi_1(x_1, \dots, x_i) &= L_q L_f^{i-1} h, \quad 1 \leq i \leq n \\ \varphi_1 &= L_q h = 0 \\ \varphi_2 &= L_q L_f h = -x_2 \\ \varphi_3 &= L_q L_f^2 h = 0 \\ \varphi_4 &= L_q L_f^3 h = \left( \frac{Mgl}{J_1} \right) x_2 \cos(x_1) - \left( \frac{k}{J_1} \right) x_4, \quad \text{avec} \quad u = \frac{1}{L_g L_f^3 h} (-L_f^4 h + v) \end{aligned} \tag{4.29}$$

Les valeurs numériques :

$$M = 1, \quad g = 9.8, \quad l = 2 \quad \hat{\theta}_0 = \begin{bmatrix} 0.474 \\ 0.95 \end{bmatrix}, \quad x_1(0) = x_2(0) = x_3(0) = x_4(0) = 0$$

$$J_1 = 4, \quad J_m = 2, \quad k = 100, \quad \text{avec} \quad c_1 = 65, c_2 = 5.5, c_3 = 5, c_4 = 20,$$

$$T_e = 10^{-3} s, F_1 = 1, \quad F_m = 1 \quad z_1(0) = z_2(0) = z_3(0) = z_4(0) = 0, \quad \Gamma = 10^{-5} \begin{bmatrix} 8 & 0.4 \\ 0.4 & 0.4 \end{bmatrix},$$

Les résultats de simulation sont données par les figures 4.11, 4.12, 4.13, 4.14. La sortie suit

la référence sans dépassement, le temps de réponse est égal à 2 secondes. Comme dans le précédent exemple, le paramètre  $\hat{\theta}_1$  convergent vers la vraie valeur  $\theta_1$  mais  $\hat{\theta}_2$  présente un léger offset.

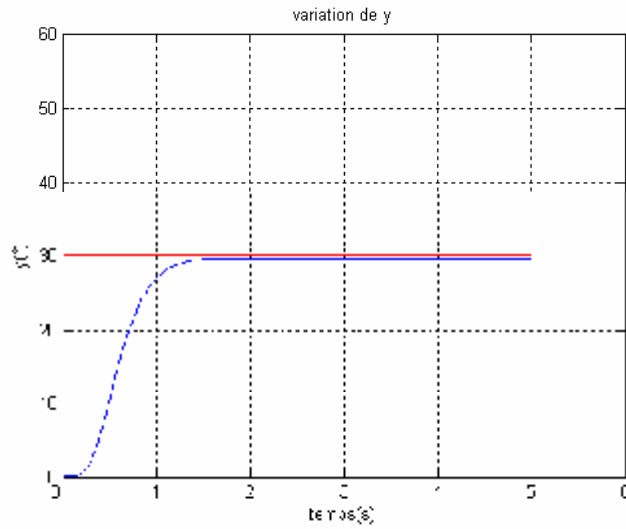


Figure 4.11 : la sortie du système  $y$  en degré

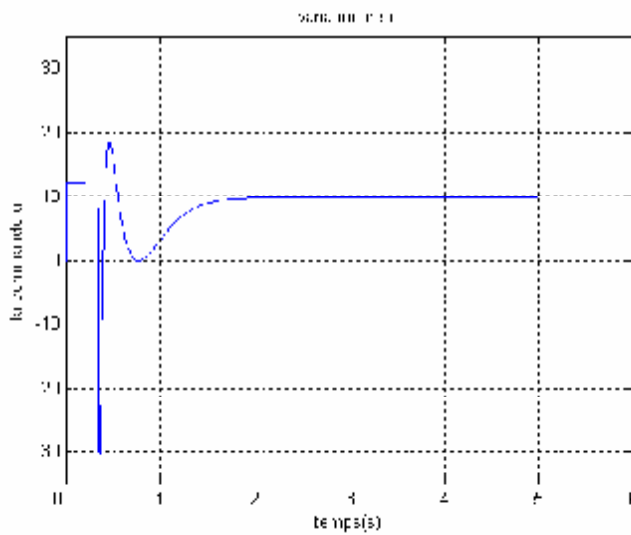


Figure 4.12 : variation de  $u$



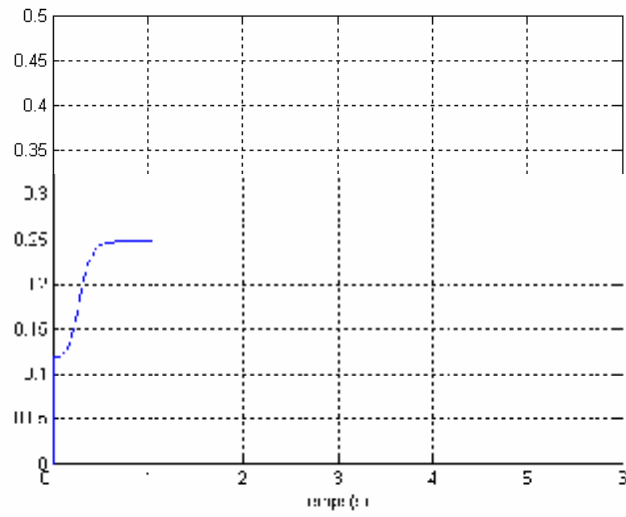


Figure 4.13 : variation de la valeur estimée  $\hat{\theta}(1)$

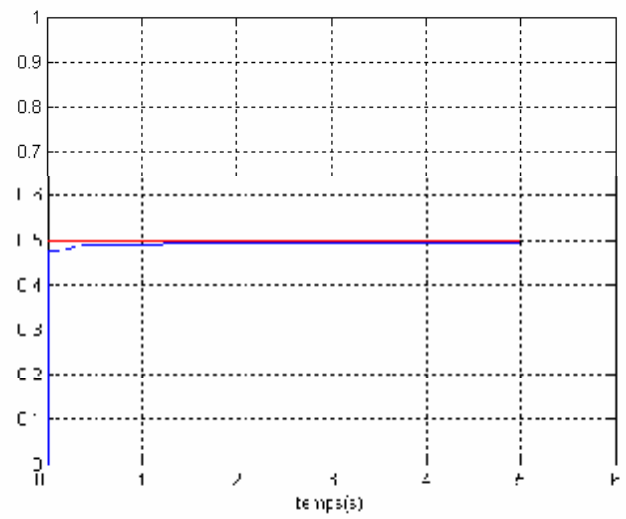


Figure 4.14 : variation de la valeur estimée  $\hat{\theta}(2)$

### III Backstepping avec observateur:

### III.1 Approche non adaptative avec observateur:

Certains systèmes ont des états inaccessibles à mesurer ou difficile ou coûteux à mesurer, soit par impossibilité physique d'introduire un capteur, soit pour des questions des coûts, on ne peut pas mesurer tous les états. La solution est d'utiliser la notion d'observateur basé sur les mesures faites sur l'entrée et la sortie du processus. Dans cette section, on introduit l'observateur sur la commande backstepping. La conception backstepping d'observateur peut être appliquée au système à retour de sortie, dans lequel les non linéarités dépendent seulement des sorties

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \varphi_1(y) \\
 \dot{x}_2 &= x_3 + \varphi_2(y) \\
 &\vdots \\
 \dot{x}_{\rho-1} &= x_\rho + \varphi_{\rho-1}(y) \\
 \dot{x}_\rho &= x_{\rho+1} + \varphi_\rho(y) + b_m \beta(y) u \\
 &\vdots \\
 \dot{x}_{n-1} &= x_n + \varphi_{n-1}(y) + b_1 \beta(y) u \\
 \dot{x}_n &= \varphi_n(y) + b_0 \beta(y) u \\
 y &= x_1
 \end{aligned} \tag{4.30}$$

On suppose que le système est minimum de phase, c.-à-d.  $b_m s^m + \dots + b_1 s + b_0$  est un polynôme de Hurwitz, et  $\beta(y) \neq 0 \quad \forall y \in R$ .

Pour dériver un observateur pour le système, nous réécrivons le système sous la forme :

$$\begin{aligned}
 \dot{x} &= Ax + \varphi(y) + b\beta(y)u, \quad x(0) = x_0 \\
 y &= C^T x
 \end{aligned} \tag{4.31}$$

Où

$$A = \begin{bmatrix} 0 & I \\ \vdots & \\ 0 & \dots & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_m \\ \vdots \\ b_0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \varphi(y) = \begin{bmatrix} \varphi_1(y) \\ \vdots \\ \varphi_n(y) \end{bmatrix}, \tag{4.32}$$

Alors un observateur est défini par

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + k(y - \hat{y}) + \varphi(y) + b\beta(y)u \quad \text{Avec } k = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}, \quad k_i \geq 0, \quad 0 \leq i \leq n \\ \hat{y} &= C^T \hat{x}, \end{aligned} \quad (4.33)$$

On choisit  $k$  de façon que  $A_0 = A - kC^T$  est une matrice de Hurwitz ,

En a L'erreur d'observateur :  $\tilde{x} = x - \hat{x}$

Alors

$$\begin{aligned} \dot{\tilde{x}} &= A_0 \tilde{x}, \quad \tilde{x}(0) = \tilde{x}_0 \\ \text{et } \tilde{x}_0 &= x_0 - \hat{x}_0 \end{aligned} \quad (4.34)$$

Supposons que le signal de référence  $y_r(t) \in C^\rho[0, \infty)$ , On définit les expressions récursives suivantes :

$$\begin{aligned} \xi_1 &= y - y_r \\ \alpha_1(y) &= -c_1 \xi_1 - d_1 \xi_1 - \varphi_1(y) \\ \xi_i &= \hat{x}_i - \alpha_{i-1}(y, \hat{x}_1, \dots, \hat{x}_{i-1}, y_r, \dots, y_r^{(i-2)}) - y_r^{(i-1)} \\ \alpha_i &= -c_i \xi_i - d_i \xi_i - d_i \left( \frac{\partial \alpha_{i-1}}{\partial y} \right)^2 \xi_i - k_i (y - \hat{x}_1) - \varphi_i(y) + \frac{\partial \alpha_{i-1}}{\partial y} [\hat{x}_2 + \varphi_1(y)] \\ &\quad + \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial \hat{x}_j} [\hat{x}_{j+1} + k_j (y - \hat{x}_1) + \varphi_j(y)] \\ &\quad + \sum_{j=1}^{i-2} \frac{\partial \alpha_{i-1}}{\partial y_r^j} y_r^{j+1}, \quad i = 1, \dots, \rho. \end{aligned} \quad (4.35)$$

Où  $c_i, d_i, 1 \leq i \leq n$  sont des constantes positives.

Le contrôleur est alors défini par :

$$u = \frac{1}{b_m \beta(y)} (\alpha_\rho - \hat{x}_{\rho+1} - y_r^{(\rho)}) \quad (4.36)$$

### III.2 Résultats de simulation :

Dans le but d'illustrer la méthode de commande avec observateur, on propose la simulation du système d'ordre deux

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1(y) \\ \dot{x}_2 &= \theta_u u \quad \text{Avec } \varphi_1(y) = -\sin(y), \quad \theta_u = 5, \\ y &= x_1 \end{aligned} \quad (4.37)$$

Le signal de référence est :  $y_r = 1$  avec

$$c_1 = 1, \quad c_2 = 1, \quad d_1 = d_2 = 1, \quad k_1 = 30, \quad k_2 = 40, \quad T_e = 0.01$$

$$A0 = A - kC^T, \quad x_1(0) = x_2(0) = 0, \quad \hat{x}_1(0) = \hat{x}_2(0) = 0$$

$$P = \begin{bmatrix} 0.0171 & 0.0125 \\ 0.0125 & 1.0583 \end{bmatrix}$$

Les résultats sont donnés par la figure 4.15

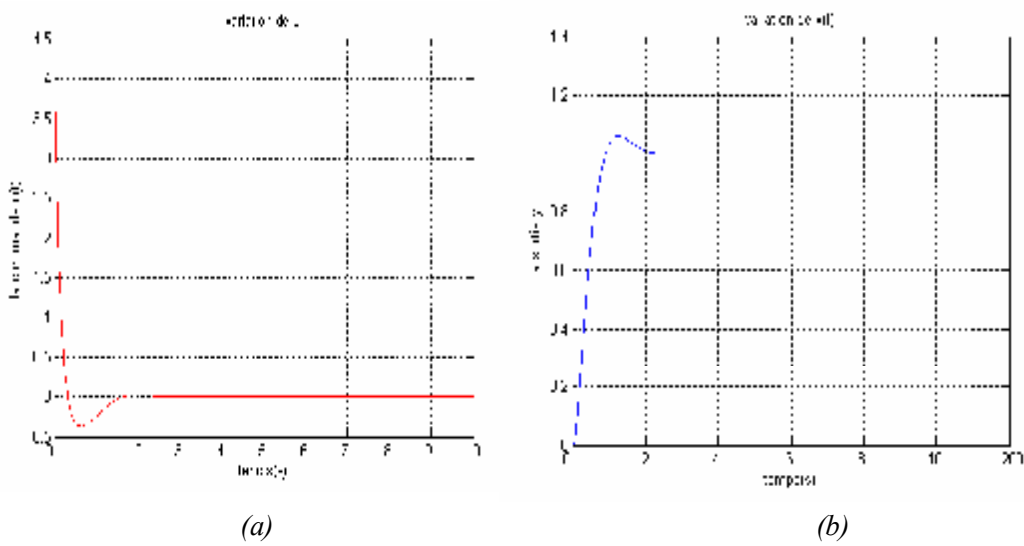


Figure 4.15 Exemple de commande backstepping non adaptative avec observateur

(a) la commande, (b) la sortie

## CHAPITRE 5 :

### CONCLUSION GENERALE

Dans ce travail, nous avons introduit une technique de commande prédictive basée sur un modèle non-linéaire reposant sur l'utilisation d'une linéarisation par retour d'état pour la commande MBPC ou retour de sortie pour la GPC. Cette technique permet de réaliser une commande non linéaire valable seulement pour les systèmes linéarisable par retour d'état dans le cas de MBPC et linéarisable par retour de sortie dans le cas de la GPC),

L'avantage de cette méthode est :

- i) On évite la linéarisation du système pour augmenter la robustesse du système commandé
- ii) Pour le cas sans contraintes on obtient une solution analytique. Pour le cas avec contraintes, le problème est plus complexe. Les nouvelles contraintes ont été obtenues par simulation.

Par ailleurs nous avons étudié deux approches de la commande adaptative et non adaptative en utilisant la technique backstepping. Dans le cas non adaptatif, tous les paramètres du système sont connus ce qui a permis d'avoir des résultats globalement acceptables. Lorsque ces paramètres sont inconnus, nous avons adoptés une commande adaptative indirecte basée sur une estimation en ligne des paramètres du système. Les résultats obtenus sont satisfaisants.

# Bibliographie

- [1] Isidori A. (1989). *Non linear control systems*. (2<sup>nd</sup> edition) Berlin: Springer.
- [2] Kritic, M., Kanellakopoulos, I. & Kokotovic, P. V. (1995). *Nonlinear and adaptive control design*. New York: Wiley.
- [3] M. Arcak and P. Kokotovi'c, *Redesign of backstepping for robustness against unmodelled dynamics*, International Journal of Robust and Nonlinear Control **11** (2001), no. 7, 633–643.
- [4] Clarke, D. W., C. Mohtadi and P. S. Tuffs , Generalised predictive control, Parts 1 and 2, *Automatica*, 23, pp. 137-160, 1987.
- [5] Maciejowski, J. M., *Predictive Control*, Prentice Hall, 2002.
- [6] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice-a survey," *Automation*, vol. 25, no. 3, pp. 335-348, 1989.

## Annexe-A

### **A.1 La commande Prédictive Généralisée (GPC: Generalized Predictive Control):**

Dans le cas du GPC,  $x(t)$  le modèle de prédiction est donné par

$$A(z^{-1}) \cdot y(t) = B(z^{-1}) \cdot u(t-1) + C(z^{-1}) \cdot e(t) / \Delta \quad (A.1)$$

où :  $C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{nc} z^{-nc}$  ( $nc$  est le degré de  $C$ ).

$\Delta = 1 - z^{-1}$ , appelé opérateur de différence, son rôle est d'assurer une action intégrale sur le contrôleur à fin d'annuler l'effet des perturbations.

$e(t)$  : est un bruit gaussien blanc à moyenne nulle.

Le modèle décrit par l'équation (A.1) représentatif du système à contrôler est appelé modèle *CARIMA* (*Controlled Auto Regressive Integrated and Moving Average*), ce modèle constitue le modèle de base de la méthode GPC à partir duquel sera dérivée l'expression de la loi de commande.

#### **La fonction de coût :**

Pour déterminer la commande à appliquer au système à l'instant  $t$ , la méthode GPC minimise le critère suivant :

$$J(t) = \sum_{j=N1}^{N2} [y(t+j) - w(t+j)]^2 + \sum_{j=N1}^{Nu} \lambda^* [\Delta u(t+j-1)]^2 \quad (A.2)$$

$$\text{avec : } \Delta u(t+j) = 0 \quad \text{pour } j \geq Nu$$

où :

$N1$  : est l'horizon minimal de prédiction.

$N2$  : est l'horizon maximal de prédiction.

$Nu$  : est l'horizon de commande.

$\lambda$  : est un facteur de pondération du signal de commande ( $\lambda > 0$ ).

$w(t+j)$  : est la trajectoire future de référence connue à l'avance.

$\Delta u(t+j)$  : est l'incrément de contrôle.

Cette fonction (A.2) est minimisée à chaque échantillon et seulement le premier incrément de contrôle sera appliqué au système à contrôler.

**Le prédicteur à j-pas :**

Pour simplifier le calcul de la loi de commande GPC, on va poser :

$$C(z^{-1}) = 1, \text{ on a : } C(z^{-1}) = 1 = E_j(z^{-1}) + \Delta A(z^{-1}) + Z^j \cdot F_j(z^{-1}) \quad (A.3)$$

où  $E_j$  et  $F_j$  sont des polynôme en  $z^{-1}$ .

Cette dernière équation (A.3) est appelée équation diophantine.

Après calcul l'expression du prédicteur s'écrit :

$$y(t+j) = G_j \cdot \Delta u(t+j-1) + F_j \cdot y(t) \quad (A.4)$$

avec  $G_j = B \cdot E_j$ ,

L'équation précédente peut être après développement divisée en deux parties (*partie des signaux connus et partie des signaux inconnus*) qui s'écrit sous forme matricielle :

$$\hat{Y} = G \cdot \tilde{U} + F \quad (A.5)$$

où :  $\hat{Y}$ ,  $U$  et  $F$  sont des vecteurs de dimension  $N \times 1$  avec :

$$\hat{Y} = [y(t+1), y(t+2), \dots, y(t+N)]^t$$

$$\tilde{U} = [\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+N-1)]^t$$

$$F = [f(t+1), f(t+2), \dots, f(t+N)]^t$$

$$\text{et } f(t+j) = [g(j,j) \cdot z^{-j} + g(j,j+1) \cdot z^{-(j+1)} + \dots + g(j, nb+j-1) \cdot z^{-(nb+j-1)}] \cdot \Delta u(t+j-1) + F_j \cdot y(t).$$

la matrice  $G$  définie dans l'équation (A.5) serait une matrice de dimension  $N \times N$  tel que :

$$G = \begin{bmatrix} g(1,0) & 0 & 0 & \dots & 0 \\ g(2,1) & g(2,0) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ g(N,N-1) & g(N,N-2) & g(N,N-3) & \dots & g(N,0) \end{bmatrix} \quad (A.6)$$

**La loi de la commande prédictive :**

La loi de commande est calculée de façon à minimiser le critère quadratique suivant :

$$J = \sum_{j=1}^{N_2} [(y(t+j) - w(t+j))]^2 + \sum_{j=1}^{N_u} \lambda \cdot [\Delta u(t+j-1)]^2 \quad (A.7)$$

avec :  $\Delta u(t+j-1) = 0$  pour  $j > N_u$

Le terme  $[(y(t+j) - w(t+j))]^2$  représente la somme pondérée des erreurs futures entre les sorties futures et les signaux de consignes  $w(t+j)$  ( $j=1..N_2$ ).

Le terme  $[\lambda \cdot \Delta u(t+j-1)]^2$  représente le coût de l'effort de commande

Le critère (A.10) peut être écrit sous forme matricielle comme suit :

$$J = (Y - W)^T \cdot (Y - W) + \lambda \cdot U^T \cdot U \quad (A.8)$$

La relation du prédicteur est:  $Y = G \cdot U + F$  en remplaçant cette dernière dans l'équation

$$(A.8) \text{ ont trouve : } J = (G \cdot U + F - W)^T \cdot (G \cdot U + F - W) + \lambda \cdot U^T \cdot U \quad (A.9)$$



$$J \text{ est minimale si } \delta J / \delta \ddot{U} = 0 \Rightarrow U = [(G^T \cdot G) + \lambda \cdot I]^{-1} \cdot G^T \cdot (W - F) \quad (A.10)$$

où  $I$  représente la matrice identité.

Cette dernière équation fournit les incréments futurs de contrôle pour les instants 't' à  $(t+N_u-1)$ , basés sur les informations disponibles à l'instant 't'. Seulement  $\Delta u(t)$  sera appliqué au système et la commande est alors telle que :  $u(t) = \Delta u(t) + u(t-1)$

La minimisation du critère sera reprise à chaque échantillon pour le calcul de la nouvelle commande à appliquer au système.

## ***A.2 la commande prédictive généralisée avec contraintes :***

Dans le cas de la *GPC* avec contraintes le problème est de minimiser  $J$  en tenant

$$\text{compte de: } \Delta u_{\min} < \Delta u < \Delta u_{\max} \quad (A.11)$$

$$\text{Donc on peut écrire } J \text{ sous la forme : } J = \frac{1}{2} U^T Q U + d^T U + c \quad (A.12)$$

avec :

$$Q = G^T G + \lambda I \quad (A.13)$$

$$d = 2(f_{\text{conu}} - W)^T G \quad (A.14)$$

$$c = (f_{\text{conu}} - W)^T (f_{\text{conu}} - W) \quad (A.15)$$

Pour résoudre ce problème on utilise l'instruction *QP* (Quadratic programming) le programme quadratique de *Matlab*.

## ***A.3 La commande prédictive à base d'un modèle d'état (MBPC: Model Based Predictive Control):***

### ***A.3.1 Cas sans contraintes:***

#### ***Calcul des prédictions :***

Le modèle de base pour calculer les prédictions est un modèle d'état discret donné par :

$$x(k+1) = Ax(k) + Bu(k)$$

$$y(k) = C_y x(k)$$

$$z(k) = C_z x(k) \quad (A.16)$$

$x(k)$  variable d'état,  $y(k)$  sorties du système,  $z(k)$  sorties à commander,  $k$  l'instant d'échantillonnage. Souvent on a  $z(k) = y(k)$  Aussi on suppose que nous ne savons rien au

sujet des perturbations ou bruit de mesure, alors tout ce que nous pouvons faire est de prédire à partir des équations ci-dessus, nous obtenons:

$$\begin{aligned}
\hat{x}(k+1/k) &= Ax(k) + B\hat{u}(k/k) \\
\hat{x}(k+2/k) &= A\hat{x}(k+1/k) + B\hat{u}(k+1/k) \\
&= A^2x(k) + AB\hat{u}(k/k) + B\hat{u}(k+1/k) \\
&\vdots \\
\hat{x}(k+N_2 | k) &= A\hat{x}(k+N_2-1 | k) + B\hat{u}(k+N_2-1 | k) \\
&= A^{N_2}x(k) + A^{N_2-1}B\hat{u}(k | k) + \dots + B\hat{u}(k+N_2-1 | k)
\end{aligned}$$

Dans la première ligne nous avons utilisé  $\hat{u}(k|k)$  plutôt que  $u(k)$ , parce qu'au moment où nous avons besoin de calculer les prédictions nous ne savons pas quel est  $u(k)$ . Maintenant rappelons que nous avons supposé que la commande changera aux temps  $k, k+1, \dots, k+N_u-1$  seulement, et restera constante après cela. Donc nous avons  $\hat{u}(k+N_u-1)$  pour:  $N_u \leq i \leq N_2-1$ . En fait, nous voudrions avoir les prédictions exprimées quant à  $\Delta\hat{u}(k+i|k)$  plutôt que  $\hat{u}(k+i|k)$ . Le prédicteur s'écrit sous forme matricielle :

$$\begin{bmatrix} \hat{x}(k+1 | k) \\ \vdots \\ \hat{x}(k+N_u | k) \\ \hat{x}(k+N_u+1 | k) \\ \vdots \\ \hat{x}(k+N_2 | k) \end{bmatrix} = \underbrace{\begin{bmatrix} A \\ \vdots \\ A^{N_u} \\ A^{N_u+1} \\ \vdots \\ A^{N_2} \end{bmatrix} x(k) + \underbrace{\begin{bmatrix} B \\ \sum_{i=0}^{N_u-1} A^i B \\ \sum_{i=0}^N A^i B \\ \vdots \\ \sum_{i=0}^{N_2-1} A^i B \end{bmatrix} u(k-1)}_{\text{passé}} + \underbrace{\begin{bmatrix} B & \dots & 0 \\ Ab+B & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N_u-1} A^i B & \dots & B \\ \sum_{i=0}^{N_u} A^i B & \dots & AB+B \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N_2-1} A^i B & \dots & \sum_{i=0}^{N_2-N_u} A^i B \end{bmatrix} \begin{bmatrix} \Delta\hat{u}(k | k) \\ \vdots \\ \Delta\hat{u}(k+N_u-1 | k) \end{bmatrix}}_{\text{futur}} \quad (A.18)$$

Les prédictions de  $z$  sont maintenant obtenues simplement

$$\begin{aligned}
\hat{z}(k+1 | k) &= C_z x(k+1 | k) \\
\hat{z}(k+2 | k) &= C_z \hat{x}(k+2 | k) \\
&\vdots \\
\hat{z}(k+N_2 | k) &= C_z \hat{x}(k+N_2 | k)
\end{aligned} \quad (A.19)$$

où:

$$\begin{bmatrix} \hat{z}(k+1 | k) \\ \vdots \\ \hat{z}(k+N_2 | k) \end{bmatrix} = \begin{bmatrix} C_z & 0 & \dots & 0 \\ 0 & C_z & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & C_z \end{bmatrix} \begin{bmatrix} \hat{x}(k+1 | k) \\ \vdots \\ \hat{x}(k+N_2 | k) \end{bmatrix}$$

### Calcul de la commande

La fonction coût s'écrit :

$$V(k) = \sum_{i=N_1}^{N_2} \|\hat{z}(k+i|k) - r(k+i)\|_{Q(i)}^2 + \sum_{i=0}^{N_u-1} \|\Delta\hat{u}(k+i|k)\|_{R(i)}^2 \quad (A.20)$$

avec :  $r(k)$  référence,  $Q(i)$  et  $R(i)$  sont des matrices des pondérations.

Nous pouvons récrire ceci sous forme condensée :

$$V(k) = \|Z(k) - T(k)\|_Q^2 + \|\Delta U(k)\|_R^2$$

$$\text{Où } Z(k) = \begin{bmatrix} \hat{z}(k+N_1|k) \\ \vdots \\ \hat{z}(k+N_2|k) \end{bmatrix} \quad T(k) = \begin{bmatrix} \hat{r}(k+N_1|k) \\ \vdots \\ \hat{r}(k+N_2|k) \end{bmatrix} \quad \Delta U(k) = \begin{bmatrix} \Delta\hat{u}(k|k) \\ \vdots \\ \Delta\hat{u}(k+N_u-1|k) \end{bmatrix}$$

et les matrices de pondérations Q et R sont données par :

$$Q = \begin{bmatrix} Q(N_1) & 0 & \dots & 0 \\ 0 & Q(N_1+1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q(N_2) \end{bmatrix}, R = \begin{bmatrix} R(0) & 0 & \dots & 0 \\ 0 & R(1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R(N_u-1) \end{bmatrix}$$

Les prédictions sont données par l'équation (A.20) :

$$Z(k) = \Psi x(k) + Yu(k-1) + \Theta \Delta U(k) \quad (A.21)$$

$$\text{En posant : } \varepsilon(k) = T(k) - \Psi x(k) - Yu(k-1) \quad (A.22)$$

On peut écrire  $V(k) = \text{const} - \Delta U(k)^T G + \Delta U(k)^T H \Delta U(k)$

$$\text{où : } G = 2\Theta^T Q \Theta \varepsilon(k) \text{ et } H = \Theta^T Q \Theta + R \quad (A.23)$$

et ni F ni H dépendent de  $\Delta U(k)$ .

Pour calculer  $\Delta U(k)$  optimal, il faut calculer le gradient de  $V(k)$  la solution

$$\Delta U(k)_{opt} = \frac{1}{2} H^{-1} G \quad (A.24)$$

Nous utilisons seulement la partie de cette solution qui correspond au premier élément du vecteur, nous pouvons représenter ceci comme :

$$\Delta u(k)_{opt} = \begin{bmatrix} I_l \\ \underbrace{0_l, \dots, 0_l}_{(N_U-1) \text{ times}} \end{bmatrix} \Delta U(k)_{opt} \quad (A.25)$$

où :  $I_l$  est la  $l \times l$  identité matrice, et  $0_l$  est la  $l \times l$  zéro matrice.

### ***A.3.2 Cas avec contraintes:***

Le problème de la commande prédictive avec contraintes est de minimiser

$$\Delta U(k)^T H \Delta U(k) - G^T \Delta U(k) \quad (A.26)$$

avec les contraintes. Ce problème est de la forme:

$$\min_{\theta} \frac{1}{2} \theta^T \Phi \theta + \Phi' \theta \quad (A.27)$$

qui est un problème standard appelé programme quadratique : *QP* qu'on peut résoudre numériquement en utilisant des algorithmes efficaces tels que la méthode ' *active set*'

## Annexe B

### ***A.1 La linéarisation par retour d'état du système (robot avec le joint flexible):***

Le modèle s'écrit sous la forme suivante :

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\left(\frac{F_1}{J_1}\right)x_2 - \left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -\left(\frac{F_m}{J_m}\right)x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) + \left(\frac{1}{J_m}\right)u \end{aligned} \tag{B.1}$$

Pour mettre le système sous la forme (2.24 - 2.25) on applique le théorème (2.3),

Donc il faut que le système vérifie les conditions suivantes :

**i)**  $\text{span}\{g, \dots, \text{ad}_f^{n-1}g\} = R^n$ ,

**ii)** la distribution  $G_{n-2} = \text{span}\{g, \dots, \text{ad}_f^{n-2}g\}$  est involutive et a un rang constant égal à  $n-1$ ,

Ou, d'une manière équivalente, si et seulement si :

**iii)** Les distributions  $G_i = \text{span}\{g, \dots, \text{ad}_f^i g\}$ ,  $0 \leq i \leq n-2$  sont involutives et de rang constant égal à  $i+1$

Donc nous avons :

$$f(x) = \begin{bmatrix} x_2 \\ -\frac{F_1}{J_1}x_2 - \left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \\ x_4 \\ -\frac{F_m}{J_m}x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix}, \quad (B.2)$$

On peut aussi écrire (B.2) sous la forme :

$$\begin{aligned} f(x) &= x_2 \frac{\partial}{\partial x_1} + \left( -\frac{F_1}{J_1}x_2 - \left(\frac{Mgl}{J_1}\right)\sin(x_1) - \left(\frac{k}{J_1}\right)(x_1 - x_3) \right) \frac{\partial}{\partial x_2} + x_4 \frac{\partial}{\partial x_3} \\ &\quad + \left( -\frac{F_m}{J_m}x_4 + \left(\frac{k}{J_m}\right)(x_1 - x_3) \right) \frac{\partial}{\partial x_4}. \\ g &= \frac{1}{J_m} \frac{\partial}{\partial x_4}. \end{aligned} \quad (B.3)$$

Pour vérifier *i)* il faut faire les calculs suivants :

$$\begin{aligned} ad_f g &= L_f g - L_g f = 0 - \frac{1}{J_m} \left( \frac{\partial}{\partial x_3} - \frac{F_m}{J_m} \frac{\partial}{\partial x_4} \right) \\ ad_f^2 g &= L_f ad_f g - L_{ad_f g} f = 0 + \frac{1}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_2} - \frac{k}{J_m} \frac{\partial}{\partial x_4} - \frac{F_m}{J_m} \left( \frac{\partial}{\partial x_3} - \frac{F_m}{J_m} \frac{\partial}{\partial x_4} \right) \right) \\ &= \frac{1}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_2} - \frac{F_m}{J_m} \frac{\partial}{\partial x_3} + \left( \frac{F_m^2}{J_m^2} - \frac{k}{J_m} \right) \frac{\partial}{\partial x_4} \right), \\ ad_f^3 g &= L_f ad_f^2 g - L_{ad_f^2 g} f = -\frac{1}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_1} - \frac{F_m}{J_m} \left( \frac{k}{J_1} \frac{\partial}{\partial x_2} - \frac{k}{J_m} \frac{\partial}{\partial x_4} \right) \right) \\ &\quad + \left( -\frac{1}{J_m} \right) \left( \left( \frac{F_m^2}{J_m^2} - \frac{k}{J_m} \right) \left( \frac{\partial}{\partial x_3} - \frac{F_m}{J_m} \frac{\partial}{\partial x_4} \right) \right), \end{aligned}$$

Donc on a :

$span\{g, ad_f g, ad_f^2 g, ad_f^3 g\} \in R^4$ , la condition *i)* satisfait,

$$\begin{aligned} [g, ad_f g] &= L_g L_{ad_f g} - L_{ad_f g} L_g = 0 \in G_2 \\ [g, ad_f^2 g] &= L_g L_{ad_f^2 g} - L_{ad_f^2 g} L_g = 0 \in G_2 \\ [ad_f g, ad_f^2 g] &= L_{ad_f g} L_{ad_f^2 g} - L_{ad_f^2 g} L_{ad_f g} = 0 \in G_2 \end{aligned}$$

$G_2$  Involutive et a un rang constant égale à 3. La condition ii) est satisfaite. L'équation (B.1) devient :

$$\begin{aligned} \dot{z}_i &= z_{i+1}, & 1 \leq i \leq 3 \\ \dot{z}_4 &= L_f^4 h(x) + L_g L_f^3 h(x) u. \end{aligned} \quad (B.4)$$

On définit le retour d'état  $u = -\frac{L_f^4 h(x)}{L_g L_f^3 h(x)} + \frac{1}{L_g L_f^3 h(x)} v$  (B.5)

on choisit la fonction  $h(x) = x_1$

on définit la fonction  $h = x_1$  avec  $h(0) = 0$ . et  $z = T(x)$  et un difféomorphisme local défini par :

$$z = T(x) = [h, L_f h, L_f^2 h, L_f^3 h] = [z_1, z_2, z_3, z_4] \quad (B.6)$$

$$\begin{aligned} L_f^4 h &= \left( \frac{Mgl}{J_1} \sin(x_1) + \frac{F_1 Mgl}{J_1^2} \cos(x_1) + \frac{kF_1}{J_1^2} \right) x_2 \\ &+ \left( \frac{Mgl}{J_1} \cos(x_1) + \frac{k}{J_1} - \frac{F_1^2}{J_1^2} \right) \left[ \frac{F_1}{J_1} x_2 + \left( \frac{Mgl}{J_1} \right) \sin(x_1) + \left( \frac{k}{J_1} \right) (x_1 - x_3) \right] \\ &- \frac{kF_1}{J_1^2} x_4 + \frac{k}{J_1} \left[ \frac{k}{J_m} (x_1 - x_3) - \frac{F_m}{J_m} x_4 \right] \\ L_g L_f^3 &= \frac{k}{J_1 J_m}. \end{aligned} \quad (B.7)$$

Avec :

$$z_1 = h(x) = x_1$$

$$z_2 = L_f h = x_2$$

$$z_3 = L_f^2 h = -\frac{F_1}{J_1} x_2 - \left( \frac{Mgl}{J_1} \right) \sin(x_1) - \left( \frac{k}{J_1} \right) (x_1 - x_3) \quad (B.8)$$

$$z_4 = L_f^3 h = -\left( \frac{Mgl}{J_1} \cos(x_1) + \frac{k}{J_1} \right) x_2 + \frac{F_1}{J_1} \left[ \frac{F_1}{J_1} x_2 + \frac{Mgl}{J_1} \sin(x_1) + \frac{k}{J_1} (x_1 - x_3) \right] + \frac{k}{J_1} x_4.$$

### **A.2 Linéarisation en entrée-sortie par retour d'état (système robot avec le joint flexible):**

Le système est localement linéarisable en entrée-sortie par retour d'état c.-à-d.

Localement entrée-sortie équivalente à l'équation (2.44) parce que  $\rho = 4$ .

Donc on peut écrire le système (B.1) sous la forme :

$$y^{(4)} = v, \quad \text{ou} \quad F(s) = \frac{1}{s^4} \quad (B.9)$$

avec la même expression de  $v$ .

### A.3 Calcul des contraintes pour le système (robot avec le joint flexible):

Nous avons l'équation suivante :

$$u = -\frac{L_f^4 h(x)}{L_g L_f^3 h(x)} + \frac{1}{L_g L_f^3 h(x)} v \Rightarrow v = L_f^4 h(x) + L_g L_f^3 h(x) u$$

Aussi  $x_i$  est borné  $\Rightarrow$  les fonctions  $L_f^4 h(x)$  et  $L_g L_f^3 h(x)$  sont aussi bornées

$$\text{Donc } u = u_{\max} = \frac{v_{\max} - L_f^4 h_{\min}}{L_g L_f^3 h_{\min}} \Rightarrow v_{\max} = L_g L_f^3 h_{\min} u_{\max} + L_f^4 h_{\min} \quad (B.10)$$

Pour une petite simulation ou un petit calcul (pour les systèmes simples) on peut trouver la valeur max de  $v$  équivalente à la valeur max de  $u$ , et en fait l'optimisation pour les valeurs max et min de  $v$ . On obtient les contraintes sur l'incrément de commande par :

$$\text{Pour } t = k \Rightarrow v(k) = L_f^4 h(x(k)) + L_g L_f^3 h(x(k)) u(k) \quad (B.11)$$

$$t = k + 1 \Rightarrow v(k + 1) = L_f^4 h(x(k + 1)) + L_g L_f^3 h(x(k + 1)) u(k + 1),$$

Donc

$$\Delta v = v(k + 1) - v(k) = (L_f^4 h(x(k + 1)) - L_f^4 h(x(k))) + (L_g L_f^3 h(x(k + 1)) - L_g L_f^3 h(x(k))) u(k) + L_g L_f^3 h(x(k + 1)) \Delta u$$

$$\Delta v = \Delta L_f^4 h(x) + \Delta L_g L_f^3 h(x) u + L_g L_f^3 h(x) \Delta u \quad (B.12)$$

Pour notre exemple  $L_g L_f^3 h(x) = Cst$  donc on a

$$\Delta v = \Delta L_f^4 h(x) + L_g L_f^3 h(x) \Delta u \quad (B.13)$$

On fait les mêmes opérations que pour l'équation (B.10), on calcule les contraintes pour la commande GPC et la commande MBPC avec la même méthode que l'annexe C



#### A.4 La linéarisation par retour d'état du système (moteur synchrone):

Les équations dynamiques d'un moteur synchrone à aimant permanent avec la distribution sinusoïdale de flux sont données par:

$$\begin{aligned}
 \frac{\partial \delta}{\partial t} &= w \\
 \frac{\partial w}{\partial t} &= -\frac{K_m}{J} i_a \sin(p\delta) + \frac{K_m}{J} i_b \cos(p\delta) - \frac{F}{J} w - \frac{T_L}{J} \\
 \frac{\partial i_a}{\partial t} &= -\frac{R}{L} i_a + \frac{K_m}{L} w \sin(p\delta) + \frac{u_a}{L} \\
 \frac{\partial i_b}{\partial t} &= -\frac{R}{L} i_b - \frac{K_m}{L} w \cos(p\delta) + \frac{u_b}{L}
 \end{aligned} \tag{B.14}$$

On choisit comme variables d'états  $x_1, x_2, x_3, x_4$  telles que :

$$x_1 = \delta, \quad x_2 = w, \quad x_3 = i_a, \quad x_4 = i_b.$$

on pose  $p=1$ , on obtient

$$\begin{aligned}
 \frac{\partial x_1}{\partial t} &= w \\
 \frac{\partial x_2}{\partial t} &= -\frac{K_m}{J} x_3 \sin(x_1) + \frac{K_m}{J} x_4 \cos(x_1) - \frac{F}{J} x_2 - \frac{T_L}{J} \\
 \frac{\partial x_3}{\partial t} &= -\frac{R}{L} x_3 + \frac{K_m}{L} x_2 \sin(x_1) + \frac{u_a}{L} \\
 \frac{\partial x_4}{\partial t} &= -\frac{R}{L} x_4 - \frac{K_m}{L} x_2 \cos(x_1) + \frac{u_b}{L}
 \end{aligned}$$

Donc nous avons :

$$\begin{aligned}
 f &= x_2 \frac{\partial}{\partial x_1} + \left( -\frac{K_m}{J} x_3 \sin(x_1) + \frac{K_m}{J} x_4 \cos(x_1) - \frac{F}{J} x_2 - \frac{T_L}{J} \right) \frac{\partial}{\partial x_2} \\
 &\quad + \left( -\frac{R}{L} x_3 + \frac{K_m}{L} x_2 \sin(x_1) \right) \frac{\partial}{\partial x_3} + \left( -\frac{R}{L} x_4 - \frac{K_m}{L} x_2 \cos(x_1) \right) \frac{\partial}{\partial x_4} \\
 g_1 &= \frac{1}{L} \frac{\partial}{\partial x_3}, \quad g_2 = \frac{1}{L} \frac{\partial}{\partial x_4}
 \end{aligned} \tag{B.15}$$

Pour obtenir notre linéarisation par retour d'état, on suit la procédure suivante :

- a) calcul les  $k_i$ ,  $1 \leq i \leq 2$

$$m_0 = \text{rank}G_0, \quad G_0 = \text{span}\{g_1, g_2\} \Rightarrow m_0 = \text{rank} \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{L} & 0 \\ 0 & \frac{1}{L} \end{bmatrix},$$

Donc  $m_0 = 2$ , de la même façon on obtient

$$m_1 = \text{rank}G_1 - m_0, \quad G_1 = \text{span}\{G_0, \text{ad}_f g\}, \quad g = [g_1, g_2].$$

$$m_1 = 2 - 2 = 0.$$

De la même façon on trouve  $m_2 = 4 - 0 = 4$ , et

$$k_1 = \text{card}\{m_i \geq 1\}, \quad 0 \leq i \leq 2 \Rightarrow k_1 = 2$$

$$k_2 = 2.$$

b) On trouve par calcul que  $\text{rank}\{\text{span}\{\text{ad}_f^j g_i : 0 \leq j \leq 3, 1 \leq i \leq 2\}\} = 4$ .

$$\text{c) } \text{rank}G_3 = 4,$$

Donc le système (B.10) linéarisable par un retour d'état avec  $h(x) = x_1$ , les nouveaux états donnés par :

$$z_1 = x_1$$

$$z_2 = x_2$$

$$z_3 = -\frac{K_m}{J} x_3 \sin(x_1) + \frac{K_m}{J} x_4 \cos(x_1) - \frac{F}{J} x_2 - \frac{T_L}{J} \tag{B.16}$$

$$z_4 = K_m x_3 \cos(x_1) + K_m x_4 \sin(x_1)$$

$$\begin{aligned}
\phi_a &= \frac{K_m}{JL} \sin(x_1) [-Rx_3 + K_m x_2 \sin(x_1)] + \frac{K_m}{J} x_3 x_2 \cos(x_1) \\
&\quad - \frac{K_m}{JL} \cos(x_1) [-Rx_4 - K_m x_2 \cos(x_1)] + \frac{K_m}{J} x_4 x_2 \sin(x_1) + \frac{F}{J} z_3 \\
\phi_b &= -\frac{K_m}{J} \cos(x_1) [-Rx_3 + K_m x_2 \sin(x_1)] + K_m x_3 x_2 \sin(x_1) \\
&\quad - \frac{K_m}{J} \sin(x_1) [-Rx_4 - K_m x_2 \cos(x_1)] - K_m x_4 x_2 \cos(x_1).
\end{aligned} \tag{B.17}$$

Le modèle est réécrit sous la forme suivante :

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
\dot{z}_3 &= v_1 \\
\dot{z}_4 &= v_2
\end{aligned}
\quad \text{Avec} \quad \frac{K_m}{L} \begin{bmatrix} -\frac{\sin(\delta)}{J} & \frac{\cos(\delta)}{J} \\ \cos(\delta) & \sin(\delta) \end{bmatrix} \begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} \phi_a \\ \phi_b \end{bmatrix} \tag{B.18}$$

#### ***A.5 Calcul des contraintes pour le système (moteur synchrone):***

Tous les calculs des contraintes sont fait exactement comme le calcul précédent, dans la partie (B.3).