

**REPUBLIQUE ALGERIENNE 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

## **THESE**

**Présentée pour l'obtention de diplôme de**

**DOCTORAT EN SCIENCES**

**Électronique**

**OPTION**

*CONTROLE DES SYSTEMES*

**Par**

**AHMED BELHANI**

***Conception des Contrôleurs Flous Multivariables***

**SOUTENUE LE 09/juillet /2007**

**Devant le jury :**

|              |                   |                       |                           |
|--------------|-------------------|-----------------------|---------------------------|
| Président :  | Hocine Benalla    | Professeur            | Université de Constantine |
| Rapporteur : | Khaled Belarbi    | Professeur            | Université de Constantine |
| Examineurs : | Faouzi Soltani    | Professeur            | Université de Constantine |
|              | Arbaoui Fayçal    | Maître de conférences | université de Skikda      |
|              | Mohamed Boumahraz | Maître de conférences | Université de Biskra      |

## ملخص

هذا العمل يمثل إحداه و إدخال لمسة على عالم التحكم في الأنظمة المتعددة المتغيرات غير خطية . من خلال الطريقة الأولى أردنا أنجاز ملاحظ يعمل بالمنطق الغامض من أجل الحصول على انفصال لساني بين إشارات التحكم للأنظمة ذات المتغير الواحد. الطريقة تعمل على مرحلتين ، في المرحلة الأولى و بعدم الأخذ بعين الإعتبار الروابط بين إشارات التحكم ، نقوم بإنشاء مراقبات للأنظمة ذات المتغير الواحد ثم نلجأ في المرحلة الثانية إلى انجاز ملاحظ لنزرع تلك الروابط حيث أن مداخله هي إشارات التحكم المحسوبة سالفاً و مخرجه هي إشارات تعويض تضاف إلى الإشارات السابقة للحصول على إشارات نهائية.

الطريقة الثانية تعمل على مبدأ اللامركزية من أجل إضعاف التداخل بين الوسائط الداخلية للنظام و أيضاً من أجل إضعاف حساسية الجهاز إتجاه تغير وسائطه. الطريقة تعمل على حساب نظرية التخماد غير الخطي . من أجل بناء مراقبات بوسائط جيدة قمنا بإدخال الخوارزميات الوراثة المتعددة المعايير

ثلاثة اجهزة أخذت من أجل تجارب و هي : جهاز تصفية المياه ، إنسان ألي بذرا عان مرتبطان و نواسان معكوسان مرتبطان

## Résumé

*Ce travail est une contribution à la commande des systèmes MIMO non linéaires. Dans une première contribution il s'agit de procéder par un découplage linguistique entre les signaux de commande calculés pour les systèmes monovariables à travers la théorie du backstepping L'approche procède en deux étapes, dans une première étape, les interactions sont ignorées et des contrôleurs sont conçus pour les systèmes monovariables résultants. Dans une deuxième étape un superviseur à logique floue, ayant comme entrées les sorties des contrôleurs monovariables, produit des signaux de compensations qui seront ajoutés aux commandes de boucles, avec effet de minimiser l'effet des interactions en agissant par anticipation.*

*La deuxième contribution repose sur le principe de décentralisation, elle tente de minimiser l'effet des interactions entre les modes internes ainsi que la sensibilité aux variations des paramètres. Elle repose sur la théorie de non linear damping. Afin de procéder par une conception optimale, les algorithmes génétiques multi-objectifs ont été introduits.*

*Trois procédés ont été considérés pour les tests tels que la station d'épuration des eaux usées, un bras de robot à deux segments et un double pendule inversé.*

## Abstract

*This work is a contribution for control of MIMO nonlinear systems, in the first contribution we consider a decoupling matrix based on linguistic reasoning between the laws control defined by the Backstepping SISO controllers. The approach proceeds by two steps, in the first step we built the backstepping SISO controllers by ignoring the interactions. In the second step we construct a fuzzy supervisor under the Takagi-Sugeno reasoning and working in feedforward. It has as inputs the outputs of the SISO controllers and produces compensations signals; the final control vector is the result of addition of the loop- control and compensations signals*

*In the second contribution we consider a decentralization approach in order to minimize the interactions effect coming from the internals states and the parameters variations sensibility. This approach uses the nonlinear damping theory. In order to proceed by optimal design we introduce the multi- objectives genetic algorithms.*

*Three processes are considered to these two approaches: the wastewater station, the two-linked robot and the two – linked pendulum.*



To Professor Khaled BELARBI  
Director of Robotic and Automatic laboratory  
LAR  
Constantine University

To Professor David Edward Goldberg  
Director of Illinois genetic algorithms laboratory  
ILLIGAL

## *Remerciements*

*J'adresse mes vifs remerciements à Mr Khaled BELARBI, professeur à l'université Mentouri de Constantine à qui je témoigne ma gratitude pour son soutien et enthousiasme qu'il m'a manifestés pour la réalisation de cette thèse et aussi pour sa collaboration ainsi que ses conseils qu'il n'a jamais manqué de m'apporter.*

*Egalement, mes remerciements à Monsieur BENALLA Hocine, professeur à l'université Mentouri de Constantine d'avoir honoré par sa présidence du jury ainsi que tous les membres du jury d'avoir accepté d'examiner cette thèse*

*Sans oublier le Professeur David Goldberg Directeur du laboratoire ILLIGAL pour sa permanente aide [codes source des AG], Egalement les Professeurs Born-Sen Chen, Department of Electrical Engineering and Computer Science, National Tsing Hua University, Taiwan et Romeo Ortega, LSS-Supelec, France pour l'envoi des articles*

*En fin je tiens à remercier tous ceux qui m'ont aidé de près et de loin pour la réalisation de ce travail.*

# *Dédicaces*

*A mes parents*

*A ma femme*

*A mes enfants Lina et Abd errahmane*

*A ma sœur et mes frères*

## INTRODUCTION GENERALE

|                          |   |
|--------------------------|---|
| I.Introduction -----     | 1 |
| II.Etat de l'art -----   | 1 |
| III.Problématique-----   | 2 |
| IV.Plan du travail ----- | 2 |

## CHAPITRE I

### LES SYSTEMES NON LINEAIRES : DESCRIPTION ET COMMANDE

|   |    |
|---|----|
| I.Introduction -----  | 4  |
| II.Propriétés des systèmes non linéaires-----                         | 5  |
| III.Analyse des systèmes non linéaires-----                           | 6  |
| III.1. Analyse par la seconde méthode de Lyapunov -----               | 7  |
| III.1.1. Définitions et théorèmes -----                               | 7  |
| III.1.2. Détermination de la fonction de Lyapunov-----                | 10 |
| III.1.2.1. Méthode de Krasovskii-----                                 | 11 |
| III.1.2.2. Méthode de Krasovskii généralisée-----                     | 11 |
| III.1.2.3. Méthode du gradient variable -----                         | 12 |
| IV.Commande des systèmes non linéaires-----                           | 12 |
| IV.1. Linéarisation par contre-réaction: Feedback linearization ----- | 13 |
| IV.2. La commande Backstepping -----                                  | 14 |
| IV.3. La commande adaptative -----                                    | 14 |
| IV.4. Le contrôle flou-----   | 15 |
| IV.5. Le contrôle neuronal -----                                      | 16 |
| V.Sommaire-----   | 17 |

## CHAPITRE II

### LA COMMANDE BACKSTEPPING

|  |    |
|--|----|
| I.Introduction -----                               | 18 |
| II.Préliminaires -----                             | 19 |
| III.Principes de base -----                        | 20 |
| IV.Conception en absence d'incertitudes -----      | 21 |
| IV.1. Intégrateur Backstepping-----                | 21 |
| IV.2. Backstepping avec un bloc linéaire -----     | 22 |
| IV.3. Backstepping avec un bloc non linéaire ----- | 22 |
| V.Exemple de calcul-----                           | 24 |



|   |    |
|---|----|
| VI.Backstepping avec incertitudes-----    | 26 |
| VI.1. Non linear Damping -----            | 26 |
| VI.2. Backstepping avec incertitudes----- | 27 |
| VII.Sommaire.-----                        | 28 |

## CHAPITRE III

### L'APPROCHE LINGUISTIQUE DE DECOUPLAGE

|  |    |
|--|----|
| I.Introduction -----   | 29 |
| II.Description de l'approche -----   | 29 |
| II.1. Le principe de coordination par superviseur à logique floue -----    | 30 |
| II.1.1. Fondement mathématique-----  | 30 |
| II.1.2. Construction des contrôleurs Backstepping monovariabes-----        | 32 |
| II.1.3. Fonctionnement du superviseur flou -----                           | 33 |
| II.2. Description de l'algorithme d'apprentissage -----                    | 34 |
| II.2.1. Codage des paramètres -----  | 35 |
| II.2.2. Mécanisme des algorithmes génétiques multi- objectifs-----         | 36 |
| II.3. L'algorithme de conception -----                                     | 37 |
| III.Applications-----  | 38 |
| III.1. Station d'épuration des eaux usées-----                             | 38 |
| III.1.1. Description du procédé. -----                                     | 38 |
| III.1.2. Modèle mathématique. -----  | 39 |
| III.1.3. Position du problème. -----                                       | 40 |
| III.1.4. L'approche directe : Contrôleur Backstepping multivariable -----  | 40 |
| III.1.4.1. Résultats de simulation -----                                   | 41 |
| III.1.4.2. Test de robustesse-----   | 46 |
| III.1.5. Découplage par un superviseur flou -----                          | 46 |
| III.1.5.1. Résultats de simulation -----                                   | 47 |
| III.1.5.2. Test de robustesse-----   | 50 |
| III.2. bras de robots a deux segments-----                                 | 51 |
| III.2.1. Modèle mathématique. -----  | 51 |
| III.2.2. L'approche directe : Contrôleur Backstepping multivariable. ----- | 52 |
| III.2.2.1. Résultats de simulation -----                                   | 54 |
| III.2.2.2. Test de robustesse-----   | 57 |
| III.2.3. Découplage par un superviseur flou. -----                         | 60 |
| III.2.3.1. Résultats de simulation -----                                   | 62 |
| III.2.3.2. Test de robustesse-----   | 65 |
| IV.Sommaire-----   | 69 |

## CHAPITRE IV

### L'APPROCHE DE DECENTRALISATION PAR BACKSTEPPING

|   |    |
|---|----|
| I.Introduction -----                          | 70 |
| II.Fondement mathématique de l'approche ----- | 71 |

|  |    |
|--|----|
| III.Applications   | 72 |
| III.1. Double pendules inversés.                                     | 72 |
| III.1.1. Dynamique du système  | 72 |
| III.1.2. L'approche directe : Contrôleur Backstepping multivariable. | 74 |
| III.1.2.1. Résultats de simulation                                   | 74 |
| III.1.2.2. Test de robustesse  | 76 |
| III.1.3. Commande par décentralisation.                              | 77 |
| III.1.3.1. Résultats de simulation                                   | 79 |
| III.1.3.2. Test de robustesse  | 81 |
| III.2. bras de robot A DEUX SEGMENT                                  | 82 |
| III.2.1. Commande par décentralisation                               | 82 |
| III.2.1.1. Résultats de simulation                                   | 85 |
| III.2.1.2. Test de robustesse  | 88 |
| IV.Sommaire  | 91 |

## CONCLUSION GENERALE

|                     |    |
|---------------------|----|
| Conclusion générale | 92 |
|---------------------|----|

## ANNEXE A

|  |    |
|--|----|
| Quelques notions sur la géométrie Différentielle | 93 |
|--|----|

## ANNEXE B

|                            |    |
|----------------------------|----|
| Contrôleur à logique floue | 95 |
|----------------------------|----|

## ANNEXE C

|                         |    |
|-------------------------|----|
| LA commande FeedForward | 97 |
|-------------------------|----|

## ANNEXE D

|   |    |
|---|----|
| Les Algorithmes génétiques multi- objectifs | 99 |
|---|----|

|               |     |
|---------------|-----|
| Bibliographie | 103 |
|---------------|-----|

# INTRODUCTION GENERALE

## I. Introduction

Les outils et techniques de conception pour la commande des systèmes monovariabiles non linéaires peuvent être classés en deux grandes catégories. Dans la première catégorie, la conception de la loi de commande passe essentiellement par deux étapes, dans une première étape il s'agit de transformer, par l'introduction d'une commande dite linéarisante, le système non linéaire en un système linéaire à travers des algorithmes de linéarisation spécifiques, la deuxième étape repose sur l'utilisation des techniques de commande linéaire pour le système ainsi linéarisé. Cette technique est connue sous le nom de *feedback linearization* [1]. La deuxième catégorie de commande utilise des algorithmes et des techniques plus adaptées aux systèmes non linéaires, il s'agit de concevoir des contrôleurs non linéaires sans passer par une linéarisation. Différentes approches existent, telles que l'approche adaptative, l'approche floue, l'approche neuronale, ... [2].

## II. Etat de l'art

Par ailleurs, le monde réel est riche de systèmes non linéaires multivariabiles, qui se caractérisent par la présence des interactions non linéaires et ce qui rend la conception des contrôleurs plus difficile. Dans ce contexte, plusieurs méthodes et approches ont été élaborées pour la commande de ce genre de systèmes, ces méthodes peuvent se diviser en deux principaux axes, les commandes directes et les commandes indirectes [3]; dans l'approche directe, il s'agit de concevoir des contrôleurs pour le système global. Dans l'approche indirecte il s'agit d'une conception à travers deux étapes, dans une première étape, une matrice de découplage est calculée afin de convertir le système global à un ensemble de sous-systèmes non linéaires monovariabiles, la deuxième étape consiste à concevoir des contrôleurs non linéaires pour chaque sous-système en appliquant les outils nécessaires.

Dans le cas des systèmes linéaires, plusieurs approches ont été développées pour le calcul de la matrice de découplage [4], [5],[6] pour éliminer les interactions. Cette matrice doit

être inversible durant l'adaptation des paramètres ce qui présente l'inconvénient principal de cette approche. D'autres approches ont été proposées pour éviter ce problème de singularité dont on peut citer l'algorithme de projection établi par *Sastry* et *Isodori* et à travers lequel les paramètres estimés n'engendrent pas un problème de singularités [7]. L'inconvénient de cet algorithme est qu'il nécessite une connaissance à priori de l'ensemble des paramètres pour lesquels le problème de singularité ne se pose pas. Une solution à ce problème a été élaborée à travers les réseaux de neurones, il s'agit de choisir un vecteur de poids initiaux du réseau de neurones proches des valeurs idéales pour lesquelles la matrice de découplage ne présente pas de singularité [8], [9]. Une autre solution a été proposée par *Slotine* afin d'éviter le problème de singularité, l'auteur a procédé par la construction des approximateurs de fonctions non linéaires afin d'éviter l'utilisation de la matrice de découplage [10].

Il est noter que les algorithmes et les approches cités précédemment reposent essentiellement sur le principe de découplage entrées/sorties. Il s'agit d'éviter l'influence de l'entrée de chaque sous-système sur le comportement désiré des autres sous-systèmes, cependant, les états internes du système global peuvent provoquer des phénomènes indésirables pour l'ensemble des sous-systèmes, ce qui nécessite d'autres méthodes de commande. Dans ce cas, une autre technique a été développée, il s'agit de la technique de décentralisation qui tente de décentraliser l'ensemble des sous-systèmes ce qui rend le système global un ensemble de sous systèmes non linéaires complètement séparés [11], [12].

### III. Problématique

La difficulté d'avoir une matrice de découplage et d'éviter les non singularités dans cette matrice pose un réel problème. Est-il alors possible de découpler le système sans passer par cette matrice? D'un autre côté est-il possible de procéder par la construction des contrôleurs- décentralisateur qui tente d'éviter le problème de l'incertitude sans passer par la commande adaptative?

### IV. Plan du travail

Le présent travail traite ces deux aspects. La première approche est basée sur un découplage à base d'un système d'inférence et règles flous. Pour remédier au problème de calcul de la matrice de découplage et au problème de singularité, on introduit un compensateur à logique floue dans la boucle de commande qui tente d'assurer un

découplage à travers des règles linguistiques basées sur le modèle de *Takagi- Sugeno*. Les signaux issus du superviseur sont générés en se basant sur l'information issue des contrôleurs monovariables. Cependant, toute variation des paramètres de la structure peut influencer sur les lois de commandes monovariables et donc sur l'information donnée au superviseur. La deuxième utilise un contrôleur – décentralisateur à base de la commande non linéaires backstepping. On considère que les interactions à travers les signaux de commandes et à travers les états internes se présentent comme des perturbations, ensuite se basant sur la théorie du backstepping, on construit des commandes pour chaque sous-système. Pour les deux approches proposées, les paramètres de conception sont optimisés par les algorithmes génétiques multi- objectifs

Cette thèse est organisée comme suit, après cette introduction, on présente brièvement une description des systèmes non linéaires ainsi que les méthodes les plus répandues pour la commande, le deuxième chapitre traite en détail la commande backstepping avec ses deux composantes : avec incertitudes et sans incertitudes. La méthode de découplage par un superviseur flou fait l'objet d'un troisième chapitre, alors que la décentralisation par le backstepping sera traitée dans le quatrième chapitre et en fin une conclusion générale est des perspectives.



**Alexandre Mikhailovitch Lyapunov**  
6 juin 1857 - 3 novembre 1918

*It is known that at a certain angular velocity ellipsoidal forms cease to be the forms of equilibrium of a rotating liquid. In this case, do they not shift into some new forms of equilibrium which differ little from ellipsoids for small increases in the angular velocity?*

**By Chebyshev**

## CHAPITRE I

# LES SYSTEMES NON LINEAIRES :

## DESCRIPTION ET COMMANDE

### I. Introduction

La présence des organes non linéaires dans les boucles de contrôle; nous incite à dépeindre le comportement non linéaire du système global. Il s'agit de nouveaux comportements, comparativement aux systèmes linéaires dont la mise en évidence des outils d'analyse et des théories de stabilité permettent la mise en place de nombreux outils et techniques de contrôle, cependant, le contrôle non linéaire nécessite d'autres outils d'analyse et de nouvelles théories de stabilité.

Dans ce chapitre, nous décrivons quelques propriétés des systèmes non linéaires, les éléments de base de la théorie de stabilité des systèmes non linéaire et nous évoquerons les principales méthodes de commande non linéaires.

## II. Propriétés des systèmes non linéaires

Les systèmes non linéaires sont gouvernés par des équations différentielles non linéaires, leur comportement est assez différent des systèmes linéaires, dû à un ensemble de caractéristiques. Parmi lesquelles on peut citer :

### 1. *Temps d'échappement fini- Finite escape-time-*

Dans les systèmes linéaires, les états instables, tendent vers l'infini, lorsque le temps tend vers l'infini, pour les systèmes non linéaires, ces états tendent vers l'infini pour un temps fini

### 2. *Points d'équilibre multiples*

Pour un système non linéaire décrit par

$$\dot{x} = f(x) \quad (1.1)$$

Le point d'équilibre est défini comme étant le point pour lequel la dynamique du système est nulle, analytiquement c'est les racines de l'équation  $f(x) = 0$

Pour un système linéaire, le point d'équilibre est unique, contrairement au système non linéaire qui présente plusieurs points d'équilibre. Il est à noter que la stabilité du système non linéaire, au voisinage de point d'équilibre, dépend des conditions initiales.

Le comportement du système au voisinage du point d'équilibre est étroitement lié aux modes de la matrice  $A$  du système linéarisé, différents états peuvent être distingués [13].

### 3. *Oscillations Subharmoniques, harmoniques où aperiodiques*

Un système non linéaire, sous une excitation sinusoïdale, peut présenter une sortie sinusoïdale avec des fréquences multiples, sub-fréquences et des oscillations aperiodiques.

### 4. *Cycles limites*

Les systèmes non linéaires peuvent exhiber des oscillations à amplitudes et fréquences fixes en l'absence d'une entrée exogène, ces oscillations sont appelées cycles limites.

### 5. *Bifurcation*

La théorie de la bifurcation a montré que le changement quantitatif des paramètres du système entraîne un changement qualitatif dans son comportement, ainsi la stabilité peut être affectée par ce changement quantitatif.

## 6. Chaos

Les systemes non lineaires presentent un phenomene chaos qui est defini par la sensibilite de la sortie vis-à-vis un changement des valeurs initiales

### III. Analyse des systemes non lineaires

L'analyse des systemes non lineaires revele une importance enorme dans l'etude des systemes, ce dévoilement est du a la puissance des outils theoriques d'analyse dans l'exploration des caracteristiques du systeme, contrairement a la simulation qui, lorsqu'elle n'est pas guidee par une etude theorique, risque de donner des resultats ne couvrant pas la realite du systeme. Outre, la synthese des lois de commande non lineaires repose, essentiellement, sur une analyse a priori [2].

Maintes methodes d'analyse existent et peuvent etre divisees en deux grandes categories, graphique et analytique. Dans la premiere categorie, la methode du plan de phase applicable pour les systemes deuxieme ordre, tente de decrir le comportement du systeme par rapport aux points d'equilibre a travers un trace dans un plan a deux dimensions appele plan de phase.

Les deux methodes de Lyapunov, connues sous les noms de la premiere et la seconde methode, representent des outils analytiques puissants. Dans la premiere methode, appelee egalement methode de linearisation, il s'agit d'analyser la stabilite du systeme non lineaire au voisinage du point d'equilibre a travers une approximation lineaire, la mise en evidence de cette methode, permet l'utilisation des systemes de controle lineaire. Cependant, la seconde methode est plus generale et elle repose sur le principe d'energie pour analyser la stabilite, «*un systeme physique est stable, si son energie mecanique totale, diminue avec le temps*», Lyapunov a pu decrir une fonction similaire ayant les proprietes de l'energie et pour laquelle l'analyse de stabilite devient plus commode.

La methode des fonctions descriptives, appelee egalement methode de la transmittance equivalente, est une methode approximative, utilisant des equivalents lineaires pour approximer le comportement des organes non lineaires afin d'analyser le systeme resultant dans le domaine frequentiel. Cette methode ne se restreint pas aux systemes deuxieme ordre, contrairement a la methode plan de phase. Comparativement a la methode Lyapunov qui utilise des fonctions adequates, cette methode est tres simple et tres commode. Elle est utilisee principalement dans la prediction des cycles limites ainsi que dans la prediction de subharmonique et de la reponse, face a une excitation



sinusoïdale. Elle présente autant d'avantages, elle est applicable et pour les système d'ordre inférieur et pour des systèmes d'ordre supérieur, de plus, cette méthode permet aux automaticiens d'utiliser les notions acquises pour le contrôle linéaire, du fait qu'elle utilise le domaine fréquentiel, également elle peut travailler même pour des non linéarité dures, l'inconvénient majeur de cette méthode et de trouver l'approximation adéquate.

Dans ce chapitre nous nous limitons à la description en détail de la deuxième méthode de Lyapunov qui sera la base du calcul des contrôleurs non linéaires utilisés dans nos approches

### III.1. Analyse par la seconde méthode de Lyapunov

Pour les systèmes linéaires, la stabilité repose sur la nature des valeurs propres du système, les systèmes non linéaires nécessitent d'autres outils pour l'analyse de la stabilité autour d'un point d'équilibre. En 1892, le mathématicien russe *Alexander Mikhailovich Lyapunov*, développait une théorie de stabilité pour les systèmes non linéaires et établissait les méthodes d'analyse. La seconde méthode de Lyapunov est brièvement introduite dans cette section [2], [13], [14].

#### III.1.1. Définitions et théorèmes

##### *Définition 1.1*

Pour le système (1.1), on considère que  $f : D \rightarrow \mathbb{R}^n$ ,  $D \subset \mathbb{R}^n$ , est localement Lipschitz et soit  $x = 0$ , un point d'équilibre pour le système, ce point est :

- ❖ Stable, si pour tout  $\varepsilon > 0$ , il existe  $\delta = \delta(\varepsilon) > 0$  tel que :

$$\|x(0)\| < \delta \Rightarrow \|x(t)\| < \varepsilon, \forall t \geq 0$$

- ❖ Asymptotiquement stable, s'il est stable et si  $\delta$  peut être choisi tel que :

$$\|x(0)\| < \delta \Rightarrow \lim_{t \rightarrow \infty} x(t) = 0$$

- ❖ Instable, autrement

##### *Théorème 1.1 : Stabilité locale*

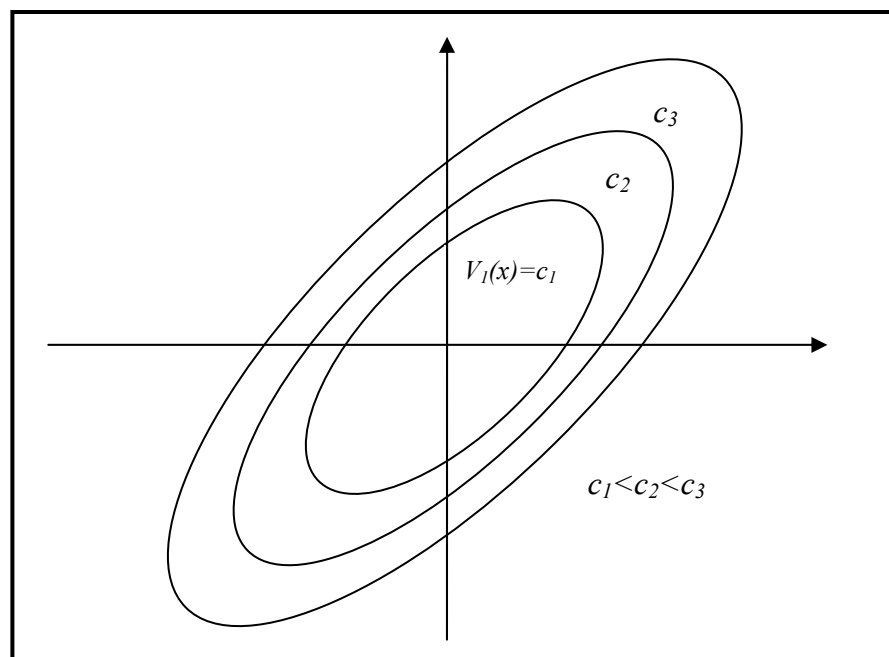
Soit  $x_0 = 0$  le point d'équilibre pour le système (1.1) et soit une fonction  $V : D \rightarrow \mathbb{R}$ , continûment différentiable dans un voisinage  $D$  de  $x_0$ . Le point d'équilibre est stable si :

- ❖  $V(0) = 0$ ;  $V(x) > 0$  sur  $D - \{0\}$  et  $\dot{V}(x) \leq 0$  sur  $D$

❖ Il est asymptotiquement stable si  $\dot{V}(x) < 0$  sur  $D - \{0\}$

La fonction  $V$  est appelée fonction de Lyapunov, les surfaces  $V(x) = c$  pour tout  $c > 0$ , sont appelées surfaces de Lyapunov ou surfaces de niveaux, La figure (I.1), montre l'interprétation graphique de ce théorème. La condition  $\dot{V}(x) \leq 0$  implique que la trajectoire de phase coupe la surface de Lyapunov  $V(x) = c$  et elle évolue à l'intérieur de l'ensemble  $\Omega_c = \{x \in \mathbb{R}^n | V(x) \leq c\}$ , cette trajectoire ne peut, en aucun cas, être à l'extérieur de cet ensemble, cependant, lorsque  $\dot{V}(x) < 0$ , la trajectoire de phase évolue d'une surface de Lyapunov à une autre intérieure, en décrémentant  $c$ , ce qui implique que la trajectoire de phase tend vers le point d'équilibre après un certain temps.

Il à noter que lorsque  $\dot{V}(x) \leq 0$ , le point d'équilibre est stable, d'après le théorème (1.1), cependant on ne pas conclure que la trajectoire de phase tend vers ce point d'équilibre, le théorème de Lasalle traite ce point.[14][15]



**Figure I.1: Niveaux de surfaces de Lyapunov**

En se basant sur la définition énergétique de la stabilité, une classe de fonctions de formes quadratiques définit les fonctions candidates de Lyapunov, cette classe a la forme suivante :

$$V(x) = x^T P x = \sum_{i=1}^n \sum_{j=1}^n p_{ij} x_i x_j \quad (1.2)$$

**Théorème 1.2 (théorème de Barbashin-Krasovskii) : stabilité globale**

Soit une fonction  $V : \mathfrak{R}^n \rightarrow \mathfrak{R}$ , continûment différentiable, est soit le système (1.1) avec un point d'équilibre situé à l'origine ; si :

$$\diamond V(0) = 0 \text{ et } V(x) > 0, \forall x \neq 0$$

$$\diamond \|x\| \rightarrow \infty \Rightarrow V(x) \rightarrow \infty$$

$$\diamond \dot{V}(x) < 0, \forall x \neq 0$$

Le point d'équilibre a une stabilité asymptotique globale.

Sous la deuxième condition, la fonction  $V$  est dite radialement non bornée (*radially unbounded*). La détermination de la stabilité asymptotique joue un rôle essentiel pour les systèmes non linéaires. Cette importance réside dans la dérivée de la fonction de Lyapunov qui est, dans la plus part des cas, semi-définie négative ce qui est interprété par une stabilité marginale du point d'équilibre. Dans ce sens il est nécessaire d'évoquer d'autres théorèmes qui permettent d'étudier la stabilité asymptotique.

**Définition 1.2**

Soit  $x(t)$  la solution du système (1.1). le point  $p$  est appelé limite positive pour  $x(t)$  s'il existe une séquence  $\{t_n\}$  avec  $t_n \rightarrow \infty$  pour  $n \rightarrow \infty$ , telle que  $x(t_n) \rightarrow p$  pour  $n \rightarrow \infty$ . L'ensemble de toutes les limites positives est appelé l'ensemble de limites positives.

**Définition 1.3**

Soit  $x(t)$  la solution du système (1.1). Un ensemble  $\Omega$  est appelé un ensemble invariant pour ce système, si toute trajectoire de phase commençant dans  $\Omega$ , reste dans  $\Omega$ , lorsque le temps évolue :

$$x(0) \in \Omega \Rightarrow x(t) \in \Omega, t \in \mathfrak{R}$$

L'ensemble  $\Omega$  est appelé ensemble positivement invariant, si :

$$x(0) \in \Omega \Rightarrow x(t) \in \Omega, t \geq 0$$

Le plus grand l'ensemble invariant, est l'union de tous les ensembles invariants

**Définition 1.4**

On dit que la solution  $x(t)$  s'approche de l'ensemble  $\Omega$  quand  $t$  s'approche de l'infini, si pour tout  $\varepsilon > 0$ , il existe  $T > 0$  tel que :

$$\text{dist}(x(t), M) = \inf_{m \in \Omega} \|x(t) - m\| < \varepsilon, \forall t > T$$

**Théorème 1.3 (Théorème de Lasalle)**

Soit un ensemble compact (fermé et borné)  $\Omega$ , tel que toute solution  $x(t)$  du système (1.1), commençant dans  $\Omega$ , reste dans  $\Omega$ , lorsque le temps évolue dans  $\Omega$ .

Soit une fonction continûment différentiable  $V : \Omega \rightarrow \mathfrak{R}$  avec  $\dot{V}(x) \leq 0$  dans  $\Omega$  et soit  $E = \{x \in \Omega \mid \dot{V}(x) = 0\}$ . Si  $M$  est le plus grand ensemble invariant dans  $E$ , alors toute solution commençant dans  $\Omega$  s'approche de l'ensemble  $M$  lorsque  $t \rightarrow \infty$ .

Contrairement au théorème de Lyapunov, le théorème (1.3) n'exige pas que la fonction  $V$  soit positive définie. Egalement la construction de l'ensemble  $\Omega$  n'est pas liée à la construction de la fonction  $V$ . Cependant dans certaines applications la construction de la fonction  $V$  peuvent assurer l'existence de l'ensemble  $\Omega$  ; si  $\Omega_c = \{x \in \mathfrak{R}^n \mid V(x) \leq c\}$  est bornée et si  $\dot{V}(x) \leq 0$  dans  $\Omega_c$  alors on peut prendre  $\Omega = \Omega_c$

Il est à noter que lorsque la fonction  $V$  est positive définie alors l'ensemble  $\Omega_c$  est borné pour des petites valeurs de  $c > 0$ , ce qu'est ne pas le cas lorsque  $V$  n'est pas définie positive, l'ensemble  $\Omega_c$  n'est pas nécessairement borné.

Lorsque le problème consiste à établir une stabilité asymptotique de l'origine (supposé point d'équilibre) on se met devant la démonstration que le plus grand ensemble invariant c'est l'origine [13].

**III.1.2. Détermination de la fonction de Lyapunov**

Après avoir donné quelques notions et théorèmes de stabilité, il est nécessaire de considérer le problème de la détermination de la fonction de Lyapunov. Notons que pour un système linéaire invariant dans le temps sous la représentation d'état  $\dot{x} = Ax$ , l'étude de la stabilité repose essentiellement sur les modes de la matrice  $A$ , pour le système non linéaire (1.1), ils existent quelques méthodes pour la détermination de la fonction de Lyapunov. Toutes ces méthodes s'articulent sur la fonction candidate (1.2).

Pour le système (1.1), on considère un point d'équilibre situé à l'origine. Soit la matrice jacobienne :

$$A = \left. \frac{\partial f(x)}{\partial x} \right|_{x=0} \quad (1.3)$$

Alors :

- ❖ Le point s'équilibre est asymptotiquement stable si  $\text{Re } \lambda_i < 0$  pour tous les valeurs propres de la matrice  $A$
- ❖ Le point s'équilibre est instable si  $\text{Re } \lambda_i > 0$  pour une ou tous les valeurs propres de la matrice  $A$

### III.1.2.1. Méthode de Krasovskii

#### *Théorème 1.5 Krasovskii*

Pour le système (1.1) ayant comme point d'équilibre, l'origine, on définit la matrice jacobienne :

$$A(x) = \frac{\partial f(x)}{\partial x} \quad (1.4)$$

Si la matrice  $F = A + A^T$  est définie négative, au voisinage d'une région  $\Omega$ , alors le point d'équilibre est asymptotiquement stable et la fonction de Lyapunov peut s'écrire sous la forme :

$$V(x) = f^T(x)f(x) \quad (1.5)$$

Si  $\Omega$  représente l'espace d'état entier et si  $V(x) \rightarrow \infty$  lorsque  $\|x\| \rightarrow \infty$ , alors le point d'équilibre est globalement asymptotiquement stable.

### III.1.2.2. Méthode de Krasovskii généralisée

#### *Théorème 1.6 Krasovskii*

Pour le système (1.1) ayant comme point d'équilibre, l'origine, avec sa matrice jacobienne  $A$ , le point d'équilibre est asymptotiquement stable ; s'il existe deux matrices symétriques définies positives  $P$  et  $Q$  telle que :  $\forall x \neq 0, F(x) = A^T P + PA + Q$ ,

est semi- définie négative au voisinage de point d'équilibre, la fonction de Lyapunov peut s'écrire sous la forme :

$$V(x) = f^T(x)Pf(x) \quad (1.6)$$

### III.1.2.3. Méthode du gradient variable

Elle consiste à calculer la fonction de Lyapunov par intégration d'un gradient supposé connu, l'algorithme de raisonnement est le suivant :

- ❖ Proposer un gradient sous la forme  $\nabla V_i = \sum_{j=1}^n a_{ij}x_j$

- ❖ Trouver les coefficients  $a_{ij}$  pour satisfaire la condition de boucle (curl

condition) telle que :  $\frac{\partial \nabla V_i}{\partial x_j} = \frac{\partial \nabla V_j}{\partial x_i}$ ,  $i, j = 1 : n$

- ❖ Remplacer les valeurs de  $a_{ij}$  dans la formule du gradient, assurer que  $\dot{V} \leq 0$

- ❖ Calculer

$$V = \int_0^{x_1} \nabla V_1(x_1, 0 \dots 0) dx_1 + \int_0^{x_2} \nabla V_2(x_1, x_2 \dots 0) dx_2 + \dots \int_0^{x_n} \nabla V_n(x_1, x_2 \dots x_n) dx_n$$

- ❖ S'assurer que  $V$  soit définie positive

## IV. Commande des systèmes non linéaires

Dans ce qui précède, nous avons décrit l'ensemble des outils mathématiques nécessaires pour l'analyse des systèmes non linéaires. Dans ce qui suit, différentes approches de commande sont présentées, ces approches peuvent se diviser en deux grandes catégories, la commande indirecte et la commande directe, dans la commande indirecte, l'objectif consiste à généraliser les différentes commandes linéaires, reposant sur la représentation d'état, pour les systèmes non linéaires, à travers une linéarisation, cette technique est dite linéarisation par contre-réaction (*Feedback linearization*), ces approches sont appliquées généralement pour les systèmes qui ne présentent aucune incertitude dans leurs dynamiques. L'approche directe consiste à élaborer des techniques plus appropriées aux systèmes non linéaires avec incertitudes, dans ce cas il s'agit généralement de la commande robuste et la commande adaptative [2], [16].

Dans ce paragraphe nous introduisons brièvement les principales méthodes de synthèse des contrôleurs non linéaires.

#### **IV.1. Linéarisation par contre-réaction: *Feedback linearization***

Ce type de commande a fait son apparition dans les années 1980 avec les travaux d'*Isidori* et les apports de la géométrie différentielle. Un grand nombre de systèmes non linéaires peuvent être partiellement ou complètement transformés en systèmes possédant un comportement entrée-sortie linéaire ou entrée-état linéaire à travers un choix approprié d'une loi de commande par retour d'état non linéaire. Lorsque la dynamique des zéros sont stables, il est possible de transformer le système non linéaire en une chaîne d'intégrateurs. Après linéarisation, les techniques classiques du linéaire peuvent être appliquées. Cette approche a souvent été employée pour résoudre des problèmes pratiques de commande mais elle impose que le vecteur d'état soit mesuré et demande un modèle précis du procédé à commander. De plus, les propriétés de robustesse ne sont pas garanties face aux incertitudes paramétriques du modèle. En effet, cette technique est basée sur l'annulation exacte des termes non linéaires. Par conséquent, la présence d'incertitudes de modélisation sur les termes non linéaires rend l'annulation inexacte et l'équation entrée-sortie résultante non linéaire [1],[2].

Dans ce contexte deux variantes existent, la première dite linéarisation entrée-états (*Input-states feedback linearization*) et qui utilise le principe de placement des pôles par retour d'état pour les systèmes linéaires, il s'agit de transformer le système non linéaire en système linéaire via des transformations entrée-état adéquates. Le problème est de définir les classes des systèmes non linéaires obéissant à une telle transformation et de trouver la transformation adéquate et appropriée et à travers cette transformation, après cette transformation on se met face à une dynamique linéaire dont les techniques de commande linéaires sont applicables facilement.

Dans la deuxième technique appelée linéarisation par entrée –sortie (*output-states feedback linearization*) il s'agit de décrire une relation linéaire directe entre la commande  $u$  et la sortie  $y$ , via une nouvelle variable de commande  $v$ . La synthèse passe trois étapes. Dans une première étape, on s'intéresse à la génération d'une relation entre  $u$  et  $y$  par le biais des dérivées successives de la sortie. Cependant cette étape permet de générer un ensemble d'états internes inobservables appelé dynamique interne, dans la deuxième étape on s'intéresse étudie le comportement de la dynamique interne à travers

l'étude de stabilité des zéros dynamiques définis en absence du signal de commande. Dans la dernière étape il s'agit de concevoir le contrôleur approprié et par conséquent, il est judicieux de concevoir un contrôleur qui, à la fois, assure la stabilité dans la boucle et dans la dynamique interne.

## IV.2. La commande Backstepping

Développée par *Kokotović*, la commande backstepping utilise le principe de stabilité de Lypunov pour le calcul de la loi commande récursivement, c'est une méthode multi-étapes opérant sur les systèmes ayant une forme d'état triangulaire (strict-feedback systems), elle utilise la notion de la commande virtuelle et l'ajout des intégrateurs à chaque étape. Le chapitre suivant présente une étude en détail de cette méthode de commande.

## IV.3. La commande adaptative

Selon la définition de webster : « *l'adaptation est un auto – changement, permettant un comportement conforme et approprié face à un changement des circonstances* ». La commande adaptative est une approche utilisée pour les systèmes ayant des paramètres incertains, provenant des incertitudes dans les paramètres du système (incertitudes structurés) ou des simplifications dans la dynamique du système (incertitudes non structurés), elle est basée sur le concept de l'estimation en ligne des paramètres pour l'ajustement d'une loi de commande.

Cette commande a été développée vers les débuts des années 1950, pour la conception des autopilotes assurant de hautes performances des avions travaillant avec de grande vitesses et grande altitudes [16], les automaticiens se sont vite aperçus en effet qu'un contrôleur avec des paramètres fixes n'était pas toujours capable d'assurer les performances voulues, par exemple dans le cas où les paramètres du système variaient avec le temps. En 1958, *Whittaker, Yamron et Kezer* élaborèrent la "MIT Rule" (algorithme du gradient).

Deux approches existent, une commande adaptative indirecte, dite également commande adaptative explicite, dans laquelle les paramètres du système sont estimés en ligne pour le calcul des paramètres du contrôleur et une approche directe, dite aussi approche implicite, qui repose sur l'estimation en ligne des paramètres de contrôleur afin de les utiliser dans



la paramétrisation du système, à travers une relation étroite liant les paramètres du contrôleur et les paramètres du système.

La conception d'une commande adaptative passe obligatoirement par trois étapes :

- ❖ Choix de la loi de contrôle
- ❖ Choix de la loi d'adaptation
- ❖ Analyse de la convergence

Deux principaux schémas peuvent être cités pour le contrôle adaptatif : la commande adaptative à modèle de référence et la commande auto-ajustable.

#### **IV.4. Le contrôle flou**

Depuis l'établissement de la théorie de la logique floue par *A.LOTFI .ZADEH* en 1965, [17] qui opère sur des variables linguistiques, le contrôle flou a connu une vaste utilisation et il est devenu un outil actif et fructueux pour le développement des contrôleurs fiables pour les systèmes non linéaires ainsi que dans la modélisation [18],[19]. La structure d'un contrôleur flou est basée essentiellement sur des informations heuristiques fournies par un opérateur expérimenté agissant sur le système, ces informations forment une base de connaissance pour le système à commander servant l'émulation de la décision humaine.

Les principaux avantages d'un contrôleur à logique floue sont :

- ❖ La possibilité d'extraire une stratégie de contrôle automatique à partir d'une stratégie de contrôle linguistique basée sur une expertise humaine.
- ❖ Une implémentation adéquate et facile des FLC dans la commande des systèmes.

Bien que ces contrôleurs ont connu une vaste utilisation grâce à leurs efficacité et fiabilité dans la commande des procédés complexes, des difficultés sont apparues telles que

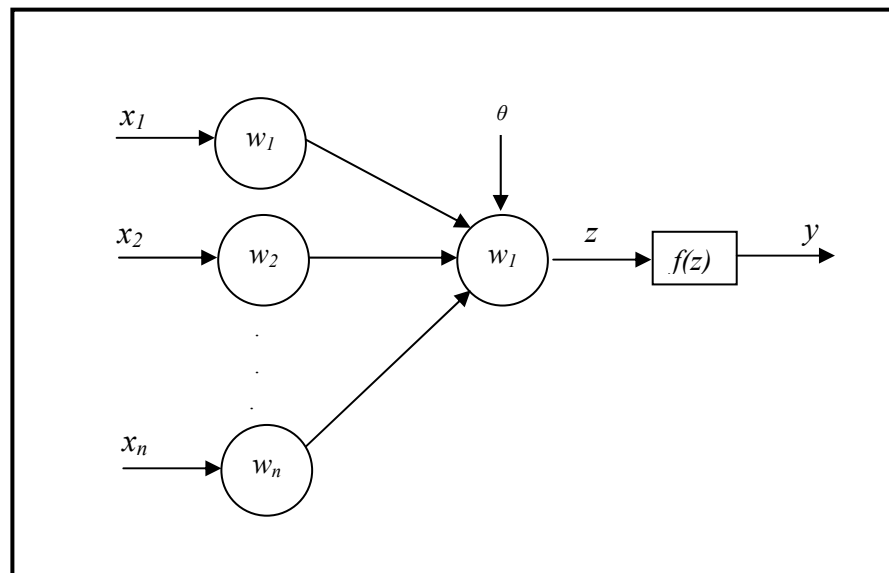
- ❖ Le domaine d'expertise n'est pas toujours disponible.
- ❖ Les opérateurs ne peuvent pas facilement transformer leurs connaissances en une caractérisation algorithmique nécessaire pour une extraction d'une stratégie de contrôle automatique.

Pour cela plusieurs approches systématiques de conception de FLC ont été développées ces dernières années [20],[21],[22],[23],[24],[25],[26]. L'étude en détail de cette technique de commande est présentée en annexe B.

#### IV.5. Le contrôle neuronal

Les réseaux de neurones artificiels sont des algorithmes et des représentations mathématiques imitant massivement le fonctionnement des réseaux de neurones biologiques. Ils ont montrés leurs efficacité et fiabilité dans de nombreux problème de contrôle, de traitement du signal, traitement d'image, reconnaissance de formes et dans l'estimation et l'identification. Leur origine vient de l'essai de modélisation du neurone biologique par *Mcculloch et Pitts*, ils supposent que l'impulsion nerveuse est le résultat d'un calcul simple effectué par chaque neurone et que la pensée née grâce à l'effet collectif d'un réseau de neurones interconnecté.

L'implantation de la théorie neuronale dans la commande se fait à travers un perceptron multicouches, le perceptron est un neurone artificiel développé par *Rosenblatt* [27], selon le schéma (I.2), la sortie du neurone  $y$  dépend des entrées  $x_i$  pondérées par les poids  $w_i$  appelés poids synaptique et d'une certaine fonction



**Figure I.2 : le neurone formel**

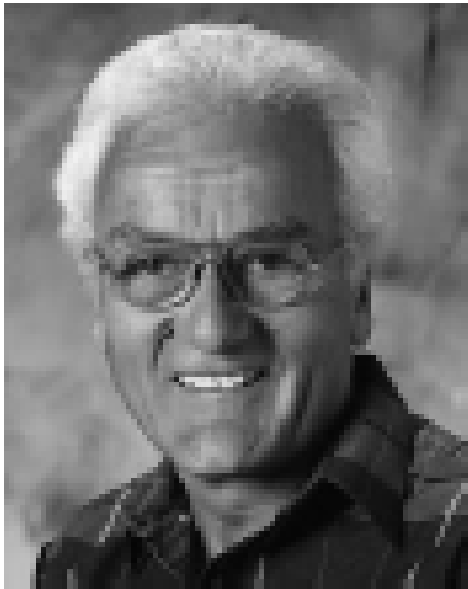
d'activation non linéaire  $f$  et généralement d'une certaine influence  $\theta$  elle prend la

$$\text{forme } y = f\left(\sum_{i=1}^n w_i x_i\right)$$

Ce neurone formel permet l'introduction d'un perceptron multicouches ou réseaux dont la propagation est directe et la valeur de la sortie présente ne dépend pas des sorties précédentes. Cette structure est très utilisée dans l'identification et le contrôle, elle constitue un approximateur universel : des recherches ont montré que le perceptron multicouche peut approximer n'importe quelle fonction entrée/sorties sous réserve de mettre suffisamment de neurones dans la couche cachées et d'utiliser des sigmoïdes pour la fonction d'activation.

## **V. Sommaire**

Dans ce chapitre nous avons présenté les systèmes non linéaires leurs propriétés, les méthodes d'analyse et les principales méthodes de commandes non linéaires. Notons qu'ils existent d'autres méthodes de commande et qui représentent des mécanismes hybrides entre les méthodes et approches citées [28]-[38].



**Petar Kokotovic**  
1936-

*It is a joy to feel this feedback from so many of you here today....*

**By Kokotovic**

## CHAPITRE II

# LA COMMANDE BACKSTEPPING

### I. Introduction

La commande Backstepping, développée par *Petar V. Kokotović*, est généralement reconnue plus intéressante que la commande par linéarisation du fait qu'elle évite le principe d'annulation des non linéarités et repose sur l'utilisation d'autres concepts pouvant améliorer les performances du régime transitoire.

C'est une approche récursive à retour d'état, basée sur la théorie de stabilité de Lyapunov applicable à une classe de systèmes non linéaires, dit triangulaire. Le principe de base consiste à considérer les états du système comme des «*commandes virtuelles*» servant à la détermination de la loi de commande exacte via différentes étapes.

Dans une première étape, on calcule une loi de commande virtuelle conduisant un état vers une position d'équilibre tout en assurant la négativité de la fonction de Lyapunov associée, cette commande virtuelle est considérée comme la valeur désirée du deuxième état, il s'agit alors, dans une seconde étape, de réaliser cet objectif, une fonction de

Lyapounov augmentée associée à l'erreur entre le deuxième état et la commande virtuelle calculée précédemment est mise en place, la négativité de cette fonction, incite la définition d'une deuxième loi de contrôle virtuelle. On suit le même raisonnement, la commande finale sera déterminée à partir des lois de commande virtuelles calculées précédemment. Ce raisonnement peut être interprété par l'ajout des intégrateurs à chaque étape [39], [40].

Dans ce chapitre nous introduisons les principes de base de cette méthode. Nous commençons par quelques définitions et théorèmes préliminaires ensuite nous décrivons la méthode backstepping.

## II. Préliminaires

Soit le système non linéaire suivant :

$$\begin{cases} \dot{x} = f(x, u) \\ x \in \mathfrak{R}^n, u \in \mathfrak{R}, f(0,0) = 0 \end{cases} \quad (2.1)$$

L'approche repose sur les concepts de la stabilité entrée-états (*Input-state stability ISS*) introduite par *Stonag*[41] et de la fonction Lyapunov de control (*Control Lyapunov function clf*) introduite par *Artstein* [42]. Dans ce qui suit, nous présentons des rappels sur ces deux notions.

### **Définition 2.1** [39]

Le système (2.1) possède une stabilité entrée-états (*ISS*) si pour toute valeur  $x(0)$  et pour tout contrôle  $u \in \mathfrak{R}$  continu et borné sur  $[0, \infty]$ , la solution existe  $\forall t \geq 0$  et satisfait la condition

$$|x(t)| \leq \beta(|x(0)|, t) + \gamma\left(\sup_{0 \leq \tau \leq t} |u(\tau)|\right), \forall t \geq 0$$

Avec  $\beta(s, t)$  et  $\gamma(s)$  des fonctions strictement croissantes de  $s \in \mathfrak{R}_+$  avec  $\beta(0, t) = 0$  et  $\gamma(0) = 0$ , de plus la fonction  $\beta$  est décroissante de  $t$  avec  $\lim_{t \rightarrow \infty} \beta(s, t) = 0, \forall s \in \mathfrak{R}_+$

### **Définition 2.2** [39]

Une fonction lisse, illimitée radialement et définie positive  $V : \mathfrak{R}^n \rightarrow \mathfrak{R}_+$  est dite fonction Lyapunov de control (clf) pour le système (2.1) si

$$\inf_{u \in \mathfrak{R}} \left\{ \frac{\partial V}{\partial x}(x) f(x, u) \right\} < 0, \forall x \neq 0$$

Ce concept vient de préciser le meilleur choix de la fonction de Lyapunov afin de réaliser une commande stabilisante, l'existence de la *CLF* est équivalente à une stabilité asymptotique globale. Pour les systèmes affines tels que:

$$\dot{x} = f(x) + g(x)u \text{ avec } f(0) = 0 \quad (2.2)$$

*Sontag* a proposé une formule appropriée pour déterminer la loi de commande stabilisante [39]. Cependant cette conception est commode pour des représentations scalaires. Pour des systèmes d'ordre élevé, la formule s'avère difficile et pour remédier à cette restriction, *Kokotović* a mis en évidence une structure de commande pour une classe de système non linéaires dite triangulaire, cette approche est gouvernée par un ensemble de lemmes et de définitions.

### III. Principes de base

#### *Supposition 2.1*

Considérons le système (2.2), avec  $x \in \mathfrak{R}^n$  son vecteur d'état et  $u$  le signal de commande, alors il existe une loi de contrôle par retour d'état, continûment différentiable :

$$\begin{cases} u = \alpha(x) \\ \alpha(0) = 0 \end{cases} \quad (2.3)$$

Et une fonction lisse, radialement illimitée et définie positive  $V : \mathfrak{R}^n \rightarrow \mathfrak{R}$  telle que :

$$\frac{\partial V(x)}{\partial x} (f(x) + g(x)\alpha(x)) \leq -W(x) \leq 0, \forall x \in \mathfrak{R}^n \quad (2.4)$$

Avec  $W : \mathfrak{R}^n \rightarrow \mathfrak{R}$  semi définie positive.

Sous cette supposition la loi de commande (2.3) appliquée au système (2.2) assure que la solution  $x(t)$  est globalement bornée. Il résulte que la convergence rapide peut être obtenue selon le théorème de *Lasalle-Yoschizawa*, avec  $\Omega = \mathfrak{R}^n$ ,  $x(t)$  converge vers le plus grand ensemble invariant  $M$  contenant dans  $E = \{x \in \Omega | W(x) = 0\}$ , si  $W(x)$  est semi-définie positive alors la loi(2.3) donne à l'origine une stabilité asymptotique globale.

### Supposition 2.2

Supposons que pour la supposition (2.1)  $V(x)$  est semi-définie positive et que le système (2.2) sous la loi de commande (2.3) présente la propriété que  $x(t)$  est borné si  $V(x)$  est bornée.

Sous cette supposition la loi de commande (2.3), outre la limite globale de  $x(t)$  elle assure aussi une régulation de  $W(x)$ .

Ces deux suppositions vont servir à l'établissement des lois de commande stabilisante backstepping pour deux cas: en absence d'incertitude et en présence d'incertitude.

## IV. Conception en absence d'incertitudes

### IV.1. Intégrateur Backstepping

#### Lemme 2.1 : [40]

Considérons le système (2.2) augmenté par un intégrateur tel que

$$\begin{cases} \dot{x} = f(x) + g(x)\xi & (2.5a) \\ \dot{\xi} = u & (2.5b) \end{cases}$$

On suppose que l'équation (2.5a) admet  $\xi$  comme sa loi de contrôle et qu'elle satisfait la supposition (2.1), alors:

- ❖ Si  $W(x)$  est définie positive, la fonction  $V_a(x, \xi) = V(x) + \frac{1}{2}(\xi - \alpha(x))^2$  est une CLF pour le système global (2.6) et il existe une loi de commande par retour d'état :

$$u = -c(\xi - \alpha(x)) + \frac{\partial \alpha}{\partial x}(x)(f(x) + g(x)u) - \frac{\partial V}{\partial x}(x)g(x), c > 0 \quad (2.6)$$

qui rend le point d'équilibre (0,0) globalement asymptotiquement stable.

- ❖ Si  $W(x)$  est, seulement, semi-définie positive, il existe une loi de commande par retour d'état, qui rend  $\dot{V}_a \leq -W_a(x, \xi) \leq 0$  telle que  $W_a(x, \xi) > 0$  quand  $W(x) > 0$  ou  $\xi \neq \alpha(x)$ . Ce qui garanti une limite globale et une convergence de  $[x(t) \ \xi(t)]^T \in \mathfrak{R}^{n+1}$  dans le plus grand ensemble invariant contenu dans l'ensemble :

$$E_a = \left\{ [x \ \xi]^T \in \mathfrak{R}^{n+1} \mid W(x) = 0, \xi = \alpha(x) \right\}$$

Pour généraliser ce théorème à plusieurs intégrateurs, deux cas ont été étudiés : backstepping avec un bloc linéaire et backstepping avec un bloc non linéaire.

## IV.2. Backstepping avec un bloc linéaire

### *Lemme 2.2 [40]*

Considérons le système sous une forme cascade:

$$\begin{cases} \dot{x} = f(x) + g(x)y, & f(0) = 0, x \in \mathfrak{R}^n, y \in \mathfrak{R} \end{cases} \quad (2.7a)$$

$$\begin{cases} \dot{\xi} = A\xi + bu, & y = h\xi, \xi \in \mathfrak{R}^q, u \in \mathfrak{R} \end{cases} \quad (2.7b)$$

Avec (2.7b) est un système à phase minimale avec un degré relatif égale à un ( $hb \neq 0$ ), si le système (2.7a) satisfait la supposition (2.2), en admettant  $y$  comme signal de contrôle (une loi de contrôle virtuelle), alors il existe une loi de commande par retour d'état garantissant une limite globale et une convergence de  $[x(t) \ \xi(t)]^T \in \mathfrak{R}^{n+q}$  vers le plus grand ensemble invariant contenu dans l'ensemble :

$$E_a = \left\{ [x \ \xi]^T \in \mathfrak{R}^{n+q} \mid W(x) = 0, y = \alpha(x) \right\},$$

Cette loi de commande est donnée par :

$$u = \frac{1}{hb} \left\{ -c(y - \alpha(x)) - hA\xi + \frac{\partial \alpha}{\partial x}(x)(f(x) + g(x)y) - \frac{\partial V}{\partial x}(x)g(x) \right\}, c > 0 \quad (2.8)$$

En outre, si  $V(x)$  et  $W(x)$  sont définies positives alors le point d'équilibre  $(0,0)$  est globalement asymptotiquement stable.

## IV.3. Backstepping avec un bloc non linéaire

### *Lemme 2.3 [40]*

Considérons le système en cascade de forme :

$$\begin{cases} \dot{x} = f(x) + g(x)y, & f(0) = 0, x \in \mathfrak{R}^n, y \in \mathfrak{R} \end{cases} \quad (2.9a)$$

$$\begin{cases} \dot{\xi} = m(x, \xi) + \beta(x, \xi)u, & \xi \in \mathfrak{R}^q, u \in \mathfrak{R} \\ y = h\xi, & h(0) = 0 \end{cases} \quad (2.9b)$$

Avec (2.9b) globalement défini avec un degré relatif égale à un uniformément dans  $x$ , de plus on suppose que son sous-système définissant les zéros dynamiques est ISS, en admettant  $y$  comme signal de contrôle, alors, sous la supposition(2.2), il existe une loi de commande par retour d'état garantissant une limite globale et une convergence de



$[x(t) \ \xi(t)]^T \in \mathfrak{R}^{n+q}$  dans le plus grand ensemble invariant contenu dans l'ensemble  $E_a = \left\{ [x \ \xi]^T \in \mathfrak{R}^{n+q} \mid W(x) = 0, y = \alpha(x) \right\}$ , cette loi de commande est donnée par :

$$u = \left( \frac{\partial h}{\partial \xi}(x, \xi) \beta(x, \xi) \right)^{-1} \left\{ -c(y - \alpha(x)) - \frac{\partial h}{\partial \xi}(x, \xi) m(x, \xi) + \frac{\partial \alpha}{\partial x}(x) (f(x) + g(x)y) - \frac{\partial V}{\partial x}(x) g(x) \right\}, c > 0 \quad (2.10)$$

En outre, si  $V(x)$  et  $W(x)$  sont définies positives alors le point d'équilibre  $(0,0)$  est globalement asymptotiquement stable.

Ce lemme est généralisable pour les deux classe de systèmes non linaires strict- feedback et pur- feedback, le lecteur peut se référer au [39], nous nous limitons ici à la définition de ces deux classes

**Définition 2.3 système strict – feedback**

Un tel système a pour représentation d'état sous cette forme :

$$\begin{cases} \dot{x} = f(x) + g(x)\xi_1 \\ \dot{\xi}_1 = f_1(x, \xi_1) + g_1(x, \xi_1)\xi_2 \\ \vdots \\ \dot{\xi}_{k-1} = f_{k-1}(x, \xi_1, \dots, \xi_{k-1}) + g_{k-1}(x, \xi_1, \dots, \xi_{k-1})\xi_k \\ \dot{\xi}_k = f_k(x, \xi_1, \dots, \xi_k) + g_k(x, \xi_1, \dots, \xi_k)u \end{cases}$$

**Définition 2.4 système pur – feedback**

Un tel système a pour représentation d'état sous cette forme :

$$\begin{cases} \dot{x} = f(x) + g(x)\xi_1 \\ \dot{\xi}_1 = f_1(x, \xi_1, \xi_2) \\ \vdots \\ \dot{\xi}_{k-1} = f_{k-1}(x, \xi_1, \dots, \xi_{k-1}, \xi_k) \\ \dot{\xi}_k = f_k(x, \xi_1, \dots, \xi_k, u) \end{cases}$$

Pour ces deux systèmes la loi de commande se détermine par une formulation récurrente [39].

## V. Exemple de calcul

Considérant un système non linéaire modélisé par la représentation d'état suivante :

$$\begin{cases} \dot{x}_1 = x_2 + f_1(x_1) \\ \dot{x}_2 = x_3 + f_2(x_1, x_2) \\ \dot{x}_3 = u + f_3(x_1, x_2, x_3) \end{cases} \quad (2.11)$$

Ce système est illustré par la figure (II.1)

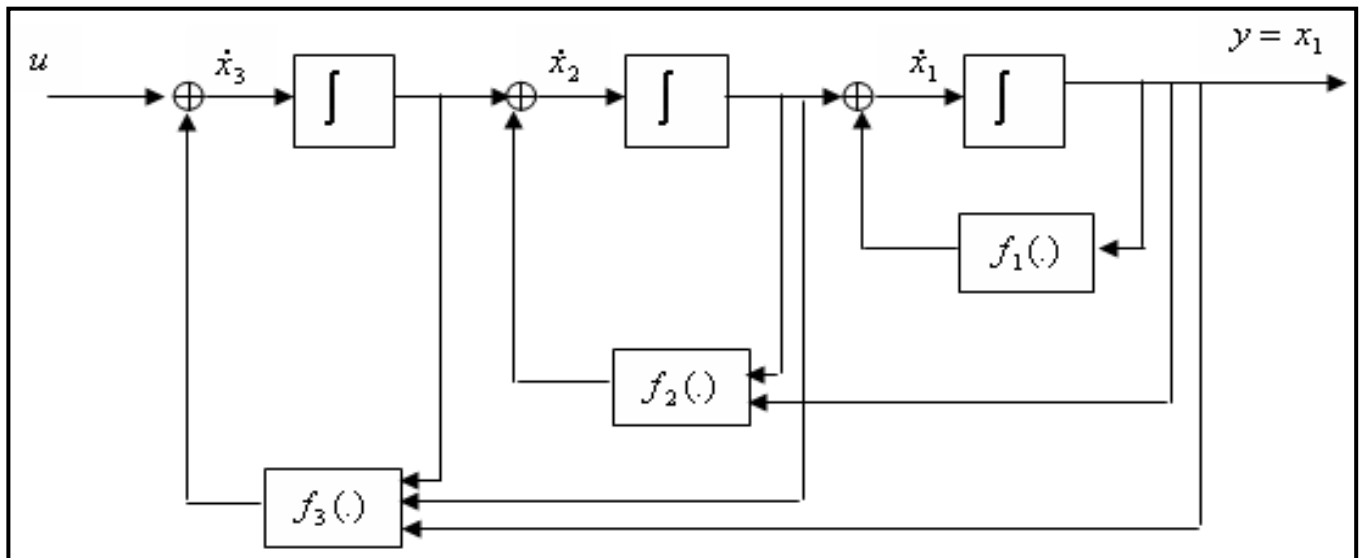


Figure II. 1 Bloc fonctionnel du système (2.11)

L'objectif est de ramener l'état  $x_1$  vers un état désiré  $x_{1d}$ .

Soit  $x_{1d}=0$ , pour simplifier, le calcul de la loi de commande passe par les étapes suivantes :

### 1ère Etape :

Dans cette étape, l'objectif est de faire conduire l'état  $x_1$  vers sa valeur désirée. Pour cela considérons la fonction de Lyapounov :

$$V_1(x_1) = 0.5x_1^2 \quad (2.12)$$

La condition de négativité de cette fonction candidate  $\dot{V}_1(x_1) \leq 0$ , permet d'écrire :

$$\frac{dV_1}{dt} = \frac{dV_1}{dx_1} \dot{x}_1 = \frac{dV_1}{dx_1} \cdot f_1 \leq 0 \Rightarrow x_2 = -c_1 x_1 - f_1(x_1) \quad (2.13)$$

L'équation (2.13) définit une loi de commande virtuelle pouvant assurer notre objectif, cette loi est notée :

$$x_2 = \alpha_1(x_1) \quad (2.14)$$

Un changement de variables adéquat de type :

$$\begin{cases} z_1 = x_1 \\ z_2 = x_2 - \alpha_1 \end{cases} \quad (2.15)$$

Ceci permet d'écrire une nouvelle représentation d'état :

$$\begin{cases} \dot{z}_1 = -c_1 z_1 + z_2 \\ \dot{z}_2 = x_3 + f_2(z_1, z_2 + \alpha_1(z_1)) - \dot{\alpha}_1(z_1) \\ \dot{x}_3 = u + f_3(z_1, z_2 + \alpha_1(z_1), x_3) \end{cases} \quad (2.16)$$

**2<sup>ème</sup> Etape :**

Dans cette étape, le but est de minimiser l'écart  $z_2$ , ce qui s'interprète par la réalisation de la loi de commande virtuelle  $\alpha_1(x_1)$ . Une fonction de Lyapounov augmentée est introduite :

$$V_2(z_1, z_2) = V_1(z_1) + 0.5z_2^2 = 0.5z_1^2 + 0.5z_2^2 \quad (2.17)$$

De la même façon, la négativité de  $V_2$  permet d'extraire une deuxième loi de commande virtuelle :

$$\begin{aligned} \frac{dV_2}{dt} \leq 0 &\Rightarrow \frac{dV_2}{dz_1} \dot{z}_1 + \frac{dV_2}{dz_2} \dot{z}_2 \leq 0 \Rightarrow z_1 \dot{z}_1 + z_2 \dot{z}_2 \leq 0 \Rightarrow -c_1 z_1^2 + z_2(z_1 + \dot{z}_2) \leq 0 \\ &\Rightarrow z_1 + \dot{z}_2 = -c_2 z_2 \end{aligned}$$

Ce qui est traduit par le choix de la commande virtuelle suivante:

$$x_3 = \alpha_2(z_1, z_2) = -c_2 z_2 - z_1 - f_2 + \dot{\alpha}_1 \quad (2.18)$$

**3<sup>ème</sup> Etape :**

Posons  $z_3 = x_3 - \alpha_2$ , alors le système (2.16) devient :

$$\begin{cases} \dot{z}_1 = -c_1 z_1 + z_2 \\ \dot{z}_2 = -z_1 - c_2 z_2 + z_3 \\ \dot{z}_3 = u + f_3(z_1, z_2 + \alpha_1(z_1), z_3 + \alpha_2(z_1, z_2)) - \dot{\alpha}_2 \end{cases} \quad (2.19)$$

A ce système on associé une fonction de Lyapounov :

$$V_3 = V_2(z_1, z_2) + 0.5z_3^2 = 0.5z_1^2 + 0.5z_2^2 + 0.5z_3^2 \quad (2.20)$$

Il vient alors :

$$\begin{aligned} \frac{dV_3}{dt} \leq 0 &\Rightarrow \frac{dV_3}{dz_1} \dot{z}_1 + \frac{dV_3}{dz_2} \dot{z}_2 + \frac{dV_3}{dz_3} \dot{z}_3 \leq 0 \Rightarrow z_1 \dot{z}_1 + z_2 \dot{z}_2 + z_3 \dot{z}_3 \leq 0 \\ &\Rightarrow -c_1 z_1^2 - c_2 z_2^2 + z_3 (z_2 + \dot{z}_3) \leq 0 \\ &\Rightarrow z_2 + \dot{z}_3 = -c_3 z_3 \end{aligned}$$

Ce qui conduit à l'extraction de la loi de commande finale :

$$u = -c_3 z_3 - z_2 - f_3 + \dot{\alpha}_2 \quad (2.21)$$

Notons que tout les  $c_i$ ,  $i=1,2,3$  sont des constantes positives

## VI. Backstepping avec incertitudes

Dans ce qui précède nous avons décrit l'approche dans le cas d'absence des incertitudes dans les non linéarités et avec des paramètres connus. Dans ce qui suit nous entamons l'approche dans le cas de présence des incertitudes. Il s'agit d'une stratégie établie pour les systèmes avec incertitudes couplées ; la loi de commande et les incertitudes apparaissent dans la même équation, cette stratégie est connue sous le nom *nonlinear damping*.

### VI.1. Non linear Damping

#### **Lemme 2.4 : [40]**

Le système (2.2) est perturbé comme suit :

$$\dot{x} = f(x) + g(x)[u + \varphi^T(x)\Delta(x, u, t)] \quad (2.22)$$

Avec  $\varphi(x) \in \mathfrak{R}^p$  représente un vecteur de fonctions non linéaires lisses connue et  $\Delta(x, u, t) \in \mathfrak{R}^p$  est un vecteur de non linéarités incertaines uniformément bornées pour toutes les valeurs de  $x, u, t$ .

Si la supposition (2.1) est réalisée alors  $W(x)$  définie positive et radialement bornée, alors le contrôle stabilisant le système en boucle fermé est donné par :

$$u = \alpha(x) - k \frac{\partial V(x)}{\partial x} g(x) | \varphi(x) |^2, k > 0 \quad (2.23)$$

Cette loi de commande rend le système en boucle fermé ISS à l'égard de l'entrée perturbatrice  $\Delta(x, u, t)$  et elle assure une borne uniforme globale de  $x(t)$  et une convergence.

Ce lemme traite la condition de stabilité du système perturbé, cependant, la combinaison de ce lemme avec celui qui traite l'intégrateur backstepping peut engendrer un outil puissant face à la présence des perturbations, cette combinaison est donnée par le lemme suivant.

## VI.2. Backstepping avec incertitudes

### *Lemme 2.5 [40]*

On considère le système suivant :

$$\dot{x} = f(x) + g(x)u + F(x)\Delta_1(x, u, t) \quad (2.24)$$

avec  $x \in \mathfrak{R}^n, u \in \mathfrak{R}, F(x) \in \mathfrak{R}^{n \times q}$  est une matrice de fonctions connues, non linéaires et lisses,  $\Delta_1(x, u, t) \in \mathfrak{R}^q$  est un vecteur de non linéarités incertaines uniformément borné pour toutes les valeurs de  $x, u, t$ . On suppose qu'il existe une loi de commande  $u = \alpha(x)$  qui rend  $x(t)$  globalement uniformément stable. Cette loi de commande est établie via deux fonctions illimités radialement et définies positive  $V(x), W(x)$  et une constante  $b$  tel que:

$$\frac{\partial V(x)}{\partial x} [f(x) + g(x)\alpha(x) + F(x)\Delta_1(x, u, t)] \leq -W(x) + b$$

Alors pour le système augmenté décrit par :

$$\begin{cases} \dot{x} = f(x) + g(x)\xi + F(x)\Delta_1(x, u, t) \\ \dot{\xi} = u + \varphi(x, \xi)^T \Delta_2(x, \xi, u, t) \end{cases} \quad (2.25)$$

Avec  $\varphi(x, \xi) \in \mathfrak{R}^p$  est un vecteur de fonctions connues non linéaires et lisses,  $\Delta_2(x, \xi, u, t) \in \mathfrak{R}^p$  est un vecteur de non linéarités incertain uniformément bornées pour toutes les valeurs de  $x, \xi, u, t$ . La loi de commande garantissant une borne uniforme de  $x(t)$  et  $\xi(t)$  est donnée par :

$$u = -c[\xi - \alpha(x)] + \frac{\partial \alpha(x)}{\partial x} [f(x) + g(x)\xi] - \frac{\partial V(x)}{\partial x} g(x) - k[\xi - \alpha(x)] \left\{ \left| \varphi(x, \xi) \right|^2 + \left| \frac{\partial \alpha(x)}{\partial x} F(x) \right|^2 \right\} \quad (2.26)$$

$$c > 0, k > 0$$

## **VII. Sommaire.**

Dans ce chapitre nous avons présenté les éléments de base de la technique backstepping. Après les définitions préliminaires nous avons introduit les lois de commande backstepping pour plusieurs cas : un système formé d'un bloc non linéaire en cascade avec un bloc linéaire, un système formé d'un bloc non linéaire en cascade avec un bloc non linéaire, enfin un système perturbé.



**Lotfi Zadeh**  
1921-

*Artificial intelligence is that field of computer usage which attempts to construct computational mechanisms for activities that are considered to require intelligence when performed by humans.*

**By Derek Partridge**

## CHAPITRE III

# L'APPROCHE LINGUISTIQUE DE DECOUPLAGE

### I. Introduction

Dans ce chapitre nous présentons une méthode de commande des systèmes non linéaires multivariables. Cette méthode est basée sur le concept de la commande par anticipation *feedforward control*. Elle est applicable aux systèmes décomposables en sous systèmes monovariables. Un contrôleur backstepping monovariables est construit pour chaque sous système. Les interactions entre les signaux de commandes sont considérées comme des perturbations mesurables. La compensation de ces interactions est réalisée à l'aide d'un superviseur à logique floue.

### II. Description de l'approche

L'approche proposée procède en deux étapes : dans une première étape, la commande stabilisante backstepping est introduite afin de réaliser les performances désirées de chaque sous-système résultant en négligeant les interactions, Dans une deuxième étape, les interactions sont considérées comme perturbations et un superviseur flou tente de les

compenser afin de garantir le fonctionnement adéquat des contrôleurs calculés dans la première étape. Le superviseur à logique floue est basé sur une commande *feedforward* [46], il a comme entrées les commandes issues des contrôleurs monvariables et fonctionne selon le type de Takagi – Sugeno. La figure (III.1) schématise le bloc fonctionnel pour un système de deux entrées et deux sorties. Les paramètres du superviseur sont déterminés de façon à minimiser les effets des interactions.

Le choix des paramètres de conception intervenant dans les fonctions de Lyapunov est fastidieux dans certains problèmes voire impossible pour les problèmes d'ordre élevés, pour remédier à ce problème, une recherche basée sur les algorithmes génétiques, est introduite, l'optimisation consiste à générer, les paramètres des contrôleurs backstepping et les paramètres du superviseur flou à savoir les univers de discours et les coefficients de l'algorithme de raisonnement.

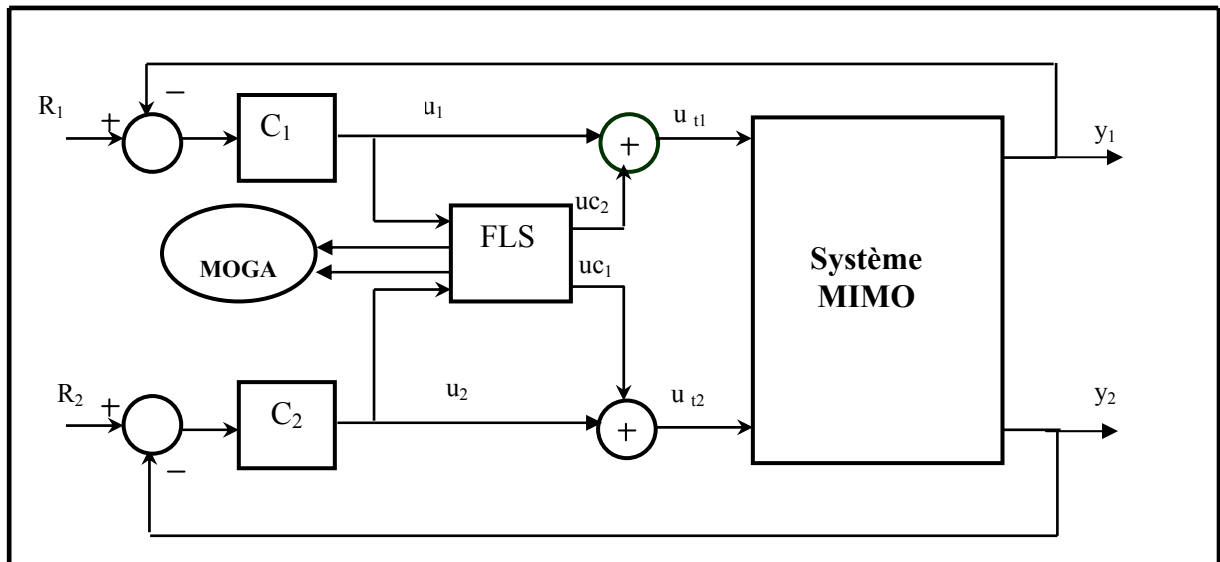


Figure III.1 Schéma descriptif de l'approche

## II.1. Le principe de coordination par superviseur à logique floue

### II.1.1. Fondement mathématique

Considérons un système multivariable non linéaire sous une forme affine, gouverné par une représentation d'état de type

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad (3.1)$$



Avec  $x \in \mathfrak{R}^n$  représente le vecteur d'état,  $f \in \mathfrak{R}^n$  est un vecteur de fonctions non linéaires,  $h \in \mathfrak{R}^m$  et  $g \in \mathfrak{R}^{n \times m}$  est une matrice des fonctions non linéaires,  $u \in \mathfrak{R}^m$ ,  $y \in \mathfrak{R}^m$  sont respectivement les signaux de contrôles et de sorties.

Dans sa construction, ce système comporte  $m$  sous-systèmes monovriables et dont les signaux de commandes sont  $u_k$ ,  $k=1:m$ , respectivement, les éléments de la matrice  $g$ , en exception de la diagonale, correspondent aux interactions et agissent d'une manière directe sur le comportement des sorties de ces sous-systèmes à travers les signaux de commandes.

Pour ce système on considère les hypothèses suivantes :

**Supposition 3.1**

1. Le système peut être divisé en  $m$  sous-systèmes ayant une forme monovriables triangulaire  $\sum_{s_1}, \sum_{s_2}, \dots, \sum_{s_m}$  et avec des blocs d'interactions tels que pour chaque sous-système on définit le triplet  $(x_i, u_i, y_i)$ , dont :

- ❖  $x_i$  vecteur d'état avec  $x = [x_1 \ x_2 \ \dots \ x_m]$ ,  $x_i \in \mathfrak{R}^{v_i}$ ,  $1 \leq i \leq m$  et  $\sum_{i=1}^m v_i = n$
- ❖  $u_i$  son signal de commande
- ❖  $y_i = h_i(x)$ ,  $1 \leq i \leq m$  représente sa sortie
- ❖  $h_i$  dépend uniquement de  $x_i$

2.  $f$  et  $g$  sont décomposables en  $m$  vecteurs et matrices respectivement, tels que:

$$f = [f_{v_1} \ f_{v_2} \ \dots \ f_{v_m}] \text{ et } g = [g_{v_1} \ g_{v_2} \ \dots \ g_{v_m}] \text{ tel que } f_{v_i} \in \mathfrak{R}^{v_i} \text{ et } g_{v_i} \in \mathfrak{R}^{v_i \times m}$$

Avec ces suppositions, la dynamique (3.1) s'écrit alors :

$$\sum_{s_i} \cdot \begin{cases} \dot{x}_i = f_{v_i}(x) + g_{v_i}(x)u \\ y_i = h_i(x) \end{cases}, 1 \leq i \leq m \tag{3.2}$$

Avec

$$f_{v_i} = [f_1 \ f_2 \ \dots \ f_{v_i}] \tag{3.3}$$

Et

$$g_{v_i} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \cdot & \cdot & \dots & \cdot \\ g_{v_i1} & g_{v_i2} & \dots & g_{v_im} \end{bmatrix} \quad (3.4)$$

Le système (3.1) est alors sous une forme triangulaire, il peut se mettre alors sous la représentation:

$$\sum_{si} : \begin{cases} \dot{x}_i = f_{v_i}(x) + \tilde{g}_{v_i}(x)u_i + \Delta_{ci}(x, u) \\ y_i = h_i(x) \end{cases}, 1 \leq i \leq m \quad (3.5)$$

Où

$$\begin{cases} \tilde{g}_{v_i} = [0 & \dots & 0 & g_{v_{ii}}] \\ \Delta_{ci}(x, u) = \begin{bmatrix} 0 & \dots & 0 & \sum_{\substack{j=1 \\ j \neq i}}^m g_{v_{ij}}u_j \end{bmatrix}, 1 \leq i \leq m \end{cases} \quad (3.6)$$

L'objectif alors est de minimiser l'effet du vecteur  $\Delta_{ci}$  sur chaque sous-système résultant afin de garantir les spécifications assurées par les contrôleurs backstepping monovariables.

Les sous-systèmes ont la forme suivante:

$$\sum_{si} : \begin{cases} \dot{x}_i = f_{v_i}(x) + \tilde{g}_{v_i}(x)u_i \\ y_i = h_i(x) \end{cases}, 1 \leq i \leq m \quad (3.7)$$

### II.1.2. Construction des contrôleurs Backstepping monovariables

Sous la supposition que  $y_i = x_{i,1}$  et qui représente la première composante du vecteur  $x_i$ , le problème consiste à conduire les sorties  $x_{i,1}$ ,  $i = 1:m$  vers des valeurs désirées  $x_{i,1d}$ .

On définit, alors, d'une manière récurrente, l'ensemble des fonctions de Lyapunov suivantes [47]:

$$\begin{cases} V_{i,1} = \frac{1}{2}(x_{i,1} - x_{i,1d})^2 = \frac{1}{2}e_{i,1}^2 \\ V_{i,k+1} = V_{i,k} + \frac{1}{2}(x_{i,k+1} - x_{i,(k+1)d})^2, k = 1 : v_i - 1 \end{cases} \quad (3.8)$$

La condition de négativité de ces fonctions de Lyapunov permet d'extraire les lois commandes suivantes:

$$u_i = \frac{-c_{v_i}(x_{i,v_i} - x_{i,v_id}) - (x_{i,v_i-1} - x_{i,(v_i-1)d}) - f_{v_i}(x) + \dot{\alpha}_{i,(v_i-1)}}{g_{v_i}}, \quad (3.9)$$

$$i = 1 : m$$

Avec

$C_{v_i} = [c_1 \quad c_2 \quad \dots \quad c_{v_i}]$  est un vecteur de composantes positives définissant les paramètres de conception,  $\alpha_{i,v_i} = [\alpha_{i,1} \quad \alpha_{i,2} \quad \dots \quad \alpha_{i,v_i-1}]$  est un vecteur contenant toutes les commandes virtuelles calculées à chaque étape

### II.1.3. Fonctionnement du superviseur flou

Le rôle du superviseur à logique flou (FLS : *fuzzy logic supervisor*) est de compenser les interactions selon la stratégie *feedforward* (voir annexe C) afin d'anticiper l'influence des interactions. Les sorties issues des contrôleurs monovariables sont injectées dans le FLS qui produit les commandes compensatrices et qui vont s'ajouter aux commandes issues des contrôleurs monovariables. Le compensateur d'interactions est basé sur le modèle de *TAKAGI- SUGENO* [48], le mécanisme d'inférence pour chaque signal de compensation est défini de la manière suivante :

$$R_{ik} : \text{if } u_k \text{ is } A_{ik} \text{ then } u_{ck} = p_{ik}u_k, k = 1 : m, i = 1 : n_k \quad (3.10)$$

$u_k$  représente le signal de commande issu de chaque contrôleur monovariable défini par la formule (3.9) ;

$A_{ik}$  représente les ensembles flous associés à chaque signal de commande ;

$u_{ck}$  définit le degré d'appartenance, de chaque signal de compensation  $u_{ck}$  à travers les paramètres de l'algorithme de raisonnement  $p_{ik}, k = 1 : m$  ;

Les valeurs numériques  $u_{ck}$  sont calculées selon la formule :

$$u_{ck} = \frac{\sum_{i=1}^{n_k} \mu_{R_{ik}} u_{ck}}{\sum_{i=1}^{n_k} \mu_{R_{ik}}} \quad (3.11)$$

$\sum_{i=1}^{n_k} \mu_{R_{ik}} = 1$  en cas de ensembles flous uniformément distribués.

Le vecteur de commande finale agissant sur la structure à commander est sous la forme :

$$u_t = u + u'_c \quad (3.12)$$

Ou chaque composante de  $u'_c$  représente le signal de compensation généré par le *FLS* à l'issu de la commande influente ; pour un système à deux entrées on a :

$$u_t = \begin{bmatrix} u_{t1} \\ u_{t2} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} u_{c2} \\ u_{c1} \end{bmatrix}$$

Cette formule présente les deux avantages:

1. éviter le calcul mathématique de la matrice de découplage ce qui évite le problème de singularité.
2. le découplage par un raisonnement linguistique permet de couvrir toute la plage de variation du vecteur  $\Delta_c$  donné par (3.6).

## II.2. Description de l'algorithme d'apprentissage

Les paramètres de conception figurant dans les lois de commandes Backstepping ainsi que les paramètres de Sugeno et les univers de discours doivent être choisis soigneusement afin de satisfaire les spécifications désirées. Ce choix est basé sur les critères suivants :

- ❖ Amélioration du régime transitoire, ce critère est assuré par les paramètres de conception des contrôleurs backstepping.
- ❖ Réduire l'effet des interactions entre les signaux de commande à travers les paramètres de Sugeno.

Cependant, le choix empirique s'avère fastidieux voire impossible dans certains cas. Dans ce contexte, on fait recours à des algorithmes d'optimisation. Dans ce travail on introduit les algorithmes génétiques multi- objectifs, (*Multi- Objectif Genetic Algorithm : MOGA*) pour les trois objectifs suivants :

- ❖ Ajuster les paramètres du FLS, on s'intéresse ici par la longueur des univers du discours.
- ❖ Trouver les paramètres optimums  $p_{ik}$  qui assurent un fonctionnement adéquat du superviseur
- ❖ Générer les paramètres de conception

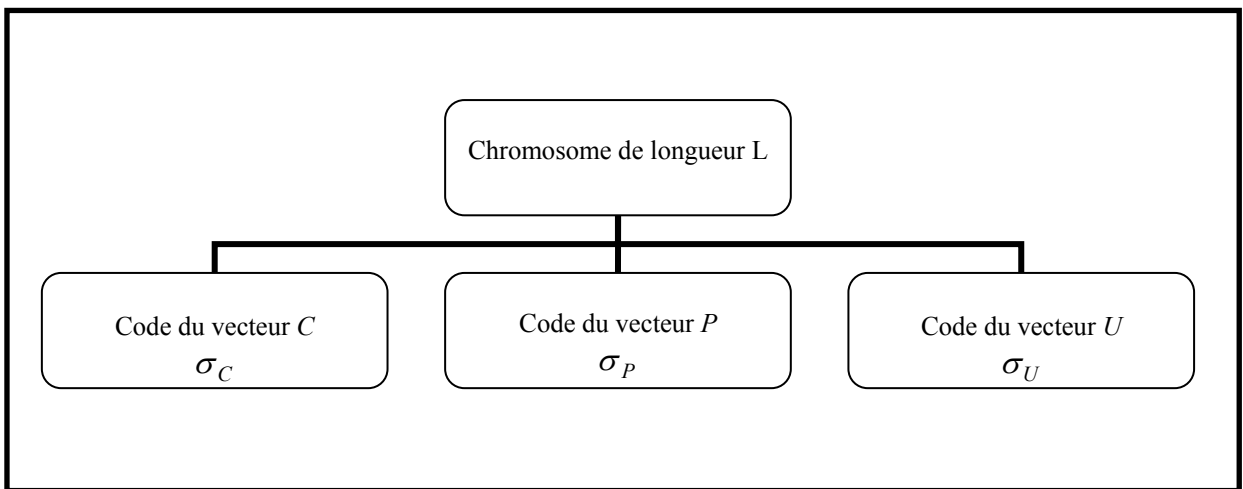
### II.2.1. Codage des paramètres

Soit :

$$C = [C_{v1} \ C_{v2} \ \dots \ C_{vm}] ;$$

$P_1 = [p_{11} \ p_{21} \ \dots \ p_{nk1}]$ ,  $P_2 = [p_{12} \ p_{22} \ \dots \ p_{nk2}]$ , ...,  $P_m = [p_{1m} \ p_{2m} \ \dots \ p_{nk m}]$  On forme alors le vecteur  $P = [P_1 \ P_2 \ \dots \ P_m]$ ;  $U_d = [u_{d1} \ u_{d2} \ \dots \ u_{dn}]$  tel que  $u_{di}$  représente l'univers de discours relatif au signal  $u_i$ .

On forme le vecteur des paramètres inconnus  $\sigma = [C \ P \ U_d]$ . Des chaînes binaires de longueur donnée sont utilisées pour coder ce vecteur, chaque chaîne est appelée chromosome et elle aura la forme suivant :



**Figure III.2 codage des paramètres**

$\sigma_C$  représente la taille du vecteur  $C$ ,  $\sigma_P$  représente la taille du vecteur  $P$ ,  $\sigma_u$  représente la taille du vecteur  $U$ . La chaîne a une taille  $L$  telle que  $L = \sigma_C + \sigma_P + \sigma_U$ .

Pour chaque paramètre notée  $\sigma_i$  du vecteur  $\sigma$ , il correspond une valeur entière naturelle  $N(\sigma_i)$  définie par l'opération du décodage binaire telle que:

$$N(\sigma_i) = \sum_{i=0}^{l_{\sigma_i}-1} 2^{l_{\sigma_i}-i-1} . b_i \quad (3.13)$$

Tel que  $l_{\sigma_i}$  représente sa taille

La valeur réelle comprise entre  $\sigma_{i\min}$  et  $\sigma_{i\max}$  :

$$\sigma_i = \sigma_{i\min} + \frac{\sigma_{i\max} - \sigma_{i\min}}{2^{l_{\sigma_i}} - 1} N(\sigma_i) \quad (3.14)$$

## II.2.2. Mécanisme des MOGA

Un algorithme génétique est un algorithme itératif de recherche d'optimum, manipulant une population de taille constante. Cette population est formée de points (candidats) appelés individu, chaque individu est symbolisé par un chromosome qui représente le codage d'une solution potentielle au problème à résoudre, il est constitué d'un ensemble d'éléments appelés gènes, pouvant prendre plusieurs valeurs appartenant à l'alphabet non forcément numérique. A chaque itération, appelée génération, est créée une nouvelle population avec le même nombre d'individu. Cette génération consiste en des chromosomes mieux "adaptés" à leur environnement tel qu'il est représenté par la fonction sélective. Au fur et à mesure des générations, les individus vont tendre vers l'optimum de la fonction sélective. La création d'une nouvelle population à partir de la précédente se fait par application des opérateurs génétiques que sont : la sélection, le croisement et la mutation.

Un algorithme génétique repose essentiellement sur trois opérateurs génétiques, la sélection des meilleurs chromosomes est la première opération. Au cours de cette opération l'algorithme sélectionne les éléments pertinents qui optimisent mieux la fonction d'adaptation. Le croisement permet de générer deux chromosomes nouveaux "enfants" à partir de deux chromosomes sélectionnés "parents", tandis que la mutation réalise l'inversion d'un ou plusieurs gènes d'un chromosome [49],[50].

Par ailleurs, beaucoup de problèmes nécessitent l'optimisation de deux critères ou plus, dans ce contexte, les algorithmes génétiques simples décrits précédemment s'avèrent vulnérables, dans le cadre où il n'existe aucune comparaison entre les objectifs. En 1984, *Schaffer* a procédé par l'établissement d'une méthode dite vecteur évalué génétiques algorithmes (VEGA), pour introduire les algorithmes génétiques dans l'optimisation

multi- objective. Il s'agit de mettre les deux objectifs dans deux sub-populations à taille égale et de ramener le problème d'optimisation multi- objective à deux problèmes séparés, cependant et après de nombreuses générations chaque sub-population converge vers un optimum sans tenir compte de l'autre [51]

Cette recherche introduite par Schaffer a encouragé les chercheurs de procéder par une recherche immédiate d'autres définitions d'optimalité, cette recherche a conduit de définir un ensemble dans lequel, toutes les solutions optimales, dites solution faisables, y sont groupées. Cet ensemble est appelé l'ensemble optimal pareto ou encore l'ensemble des solutions non- dominantes, son complément est appelé l'ensemble des solutions dominantes. Cette définition a permis de réaliser un compromis entre les objectifs lors de l'optimisation.

La combinaison de la procédure dite *nondominated sorting procedure* et la technique de partage a donné naissance à un algorithme d'optimisation multi- objectif dit *Nondominated Sorting Genetic Algorithm (NSGA)* [52],[53], [54],[55].

### II.3. L'algorithme de conception

Le calcul de la loi de commande se fait selon les étapes suivantes :

**Etape 0 :**

*Définir l'ensemble des sous-systèmes monovariabiles et les interactions ;*

**Etape 1 :**

*Définir la coordination :*

*Calcul des contrôleurs backstepping monovariabiles  $u_k$  ;*

*Etablir les règles du superviseur flou  $R_{ik}$  ;*

*Générer les signaux de compensation  $u_{ck}$  ;*

*Calcul du vecteur de commande  $u_t = u + u_c$*

**Etape 2 :**

*Déclenchement de l'algorithme d'apprentissage pour la génération d'un ensemble des solutions possibles du vecteur  $\sigma$*

**Etape 3 :**

*Application de chaque solution pendant un temps de simulation déterminé.*

**Etape 4 :**

*Mesurer l'indice de performance pour chaque solution*

**Etape 5 :**

*Refaire les étapes 2 et 3 jusqu'à un indice de performance minimal.*

**Figure III.3 Algorithme de commande pour le découplage linguistique**

### III. Applications

Deux systèmes à deux entrées et deux sorties seront considérés : une station d'épuration des eaux usées et un bras de robots à deux segments. Une étude comparative sera fait pour tester l'approche proposée, il s'agit d'une comparaison avec le backstepping multivariable dont les paramètres de contrôleur seront optimisés avec les algorithmes génétiques simples. Pour cela, l'entraînement se fait à l'aide des paramètres génétiques du tableau (III.1)

| <b>Tableau III.1<br/>Paramètres des MOGA</b> |                                    |
|--|------------------------------------|
| <i>Paramètre</i>                             | <i>Valeur</i>                      |
| <i>Génération</i>                            | 50                                 |
| <i>Population</i>                            | 100                                |
| <i>Taille du chromosome</i>                  | <i>N<sup>bre</sup> de param.10</i> |
| <i>Probabilité du croisement</i>             | 0.75                               |
| <i>Probabilité de mutation</i>               | 0.001                              |

Avec une fonction fitness telle que :

$$f = \frac{k}{\sum_{t=0}^{t_f} (1 + p_1 e_1^2(t) + p_2 e_2^2(t))} \quad (3.15)$$

$$\Gamma = [k \quad p_1 \quad p_2]$$

Dans le cas des algorithmes multiobjectifs, les deux critères à minimiser sont les erreurs de chaque sortie.

$$J = \min(e_1(t), e_2(t)) \quad (3.16)$$

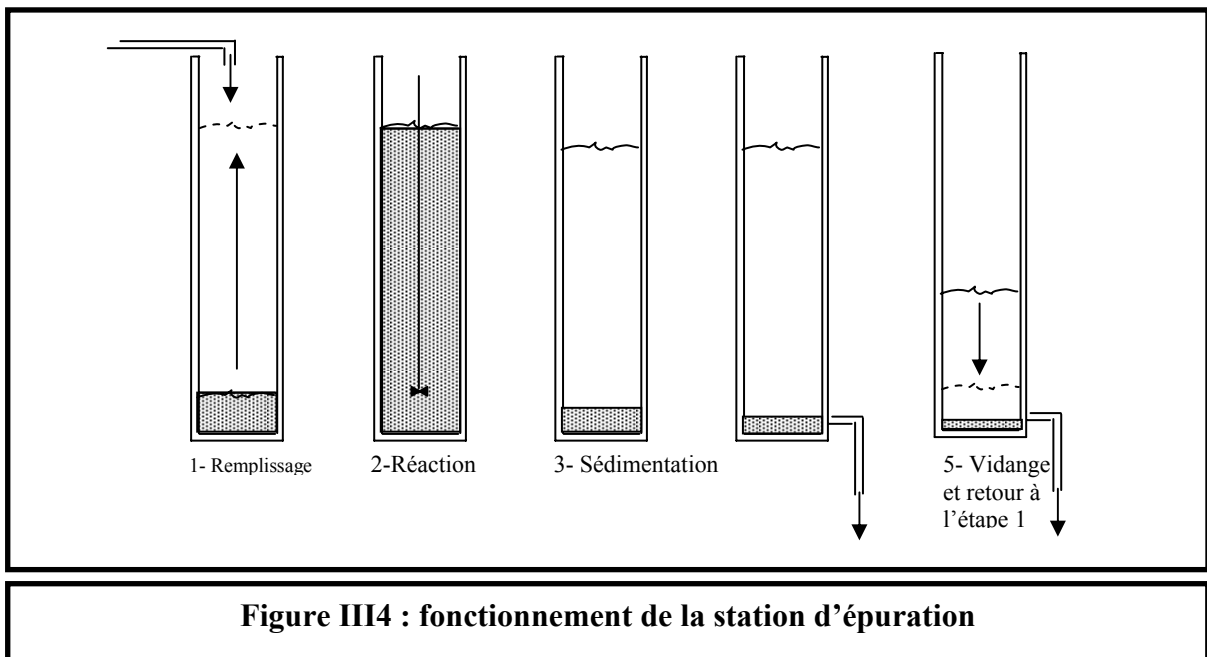
#### III.1. Station d'épuration des eaux usées

##### III.1.1. Description du procédé.

Le système étudié est un système biologique servant à la dépollution des eaux usées à travers le concept de la fermentation microbienne aidant à la consommation de substrat. Ce système présente un réacteur infiniment mélangé (*Stired Tank Reactor STR*) dans lequel le milieu réactionnel est homogène et la réaction est décrite par des équations différentielles ordinaires, il fonctionne selon un mode dit continu ou chemostat. Il est



composé de deux parties principales : le bioréacteur (aerator) est le sédiment (settler). Une réaction entre les micro-organismes, la matière organique située dans les eaux usées et l'oxygène injecté par l'air comprimée pour la production du  $CO_2$ , de l'eau et de masses cellulaires. Après cette phase, les micro-organismes sont complètement séparés dans le sédiment, une portion de la biomasse concentrée est recyclée alors que la quantité restante est évacuée afin de garantir un certain niveau de concentration, la figure (III.2) montre les étapes de la dépollution.[56],[57]



### III.1.2. Modèle mathématique.

Sous certains principes physiques et chimiques et sous certaines considérations liées au schéma réactionnel dans les bioréacteurs, le modèle mathématique est donné par [56] :

$$\begin{cases} \dot{x}(t) = \mu(t)x(t) - D(t)(1+r)x(t) + rD(t)x_r(t) \\ \dot{s}(t) = -\frac{\mu(t)x(t)}{Y} - D(t)(1+r)s(t) + D(t)S_{in} \\ \dot{c}(t) = -\frac{K_0\mu(t)x(t)}{Y} - D(t)(1+r)c(t) + K_{La}(c_s - c(t)) + D(t)C_{in} \\ \dot{x}_r(t) = D(t)(1+r)x(t) - D(t)(\beta+r)x_r(t) \end{cases} \quad (3.17)$$

$x(t)$ ,  $s(t)$ ,  $x_r(t)$ , et  $c(t)$  sont des variables d'état qui représentent, respectivement, la biomasse, le substrat, la biomasse recyclée et la concentration d'oxygène diluée,  $D(t)$  le taux de dilution,  $r$  et  $\beta$  représentent respectivement, le rapport du flux recyclé au flux influant et le rapport du flux de la quantité évacuée au flux influant,  $S_{in}$  et  $C_{in}$

correspondent aux concentrations de l'oxygène dilué et de substrat dans le flux consommé. La cinétique de la production des masses cellulaires est définie en terme d'un taux spécifique de croissance  $\mu$  et le rendement  $Y$ ;  $K_0$  est une constante,  $c_s$  est la concentration maximale diluée d'oxygène et  $KL_a$  représente le coefficient transfert de masse d'oxygène. Selon le modèle d'*Olsson*, le taux spécifique de croissance est donné par [58] :

$$\mu(t) = \mu_{\max} \frac{s(t)}{K_s + s(t)} \cdot \frac{c(t)}{K_c + c(t)} \quad (3.18)$$

### III.1.3. Position du problème.

Le problème posé est de régler le substrat  $s(t)$  et la concentration diluée d'oxygène  $c(t)$  en manipulant le taux de dilution  $D(t)$  et le débit d'air  $W$  avec  $K_{La} = \alpha W$ ,  $\alpha$  est une constante. Les paramètres cinétiques et les conditions initiales sont donnés dans les tableaux (III.2) et (III.3) respectivement [57].

| Tableau III.2<br>Paramètres cinétiques |                        |
|--|------------------------|
| Paramètre                              | Valeur                 |
| $Y$                                    | 0.65                   |
| $R$                                    | 0.6                    |
| $\beta$                                | 0.2                    |
| $\alpha$                               | 0.018                  |
| $K_c$                                  | 2 mg.l <sup>-1</sup>   |
| $\mu_{\max}$                           | 0.15 h <sup>-1</sup>   |
| $K_s$                                  | 100 mg.l <sup>-1</sup> |
| $K_0$                                  | 0.5                    |
| $C_s$                                  | 10 mg.l <sup>-1</sup>  |

| Tableau III.3<br>Conditions initiales |                        |
|---------------------------------------|------------------------|
| $x_0$                                 | 215 mg.l <sup>-1</sup> |
| $s_0$                                 | 55 mg.l <sup>-1</sup>  |
| $x_{r0}$                              | 400 mg.l <sup>-1</sup> |
| $c_0$                                 | 6 mg.l <sup>-1</sup>   |
| $s_{in}$                              | 200 mg.l <sup>-1</sup> |
| $c_{in}$                              | 0.5 mg.l <sup>-1</sup> |

Les trajectoires de références considérées sont variables tels que :

$$\begin{cases} S^* = 50 \text{ mg.l}^{-1} & \text{pour } 0 < t \leq 100h \\ S^* = 30 \text{ mg.l}^{-1} & \text{pour } 100 < t \leq 200h \end{cases} \text{ et } \begin{cases} C^* = 5 \text{ mg.l}^{-1} & \text{pour } 0 < t \leq 100h \\ C^* = 6.5 \text{ mg.l}^{-1} & \text{pour } 100 < t \leq 200h \end{cases} \quad (3.19)$$

### III.1.4. L'approche directe : Contrôleur Backstepping multivariable

Soit le système réduit suivant :

$$\begin{cases} \dot{s}(t) = -\frac{\mu(t)x(t)}{Y} - D(t)(1+r)s(t) + D(t)S_{in} \\ \dot{c}(t) = -\frac{K_0\mu(t)x(t)}{Y} - D(t)(1-r)c(t) + \alpha W(c_s - c(t)) + d(t)c_{in} \end{cases} \quad (3.20)$$

Pour le système (3.20), la conception est faite en une seule étape. Soit les états des erreurs:

$$\begin{cases} z_1 = s(t) - s^* \\ z_2 = c(t) - c^* \end{cases} \quad (3.21)$$

Pour ces états une fonction de Lyapunov est définie comme suit :

$$V = 0.5Z_1^2 + 0.5Z_2^2$$

La stabilité des lois de commandes est assurée par la négativité de cette fonction de Lyapunov, soit alors:

$$\dot{V} \leq 0 \Rightarrow z_1\dot{z}_1 + z_2\dot{z}_2 \leq 0$$

Un choix est donné par :

$$\begin{cases} \dot{z}_1 = -c_1 z_1 \\ \dot{z}_2 = -c_2 z_2 \end{cases}$$

Sous ces conditions, les lois de commandes ont les formes suivantes:

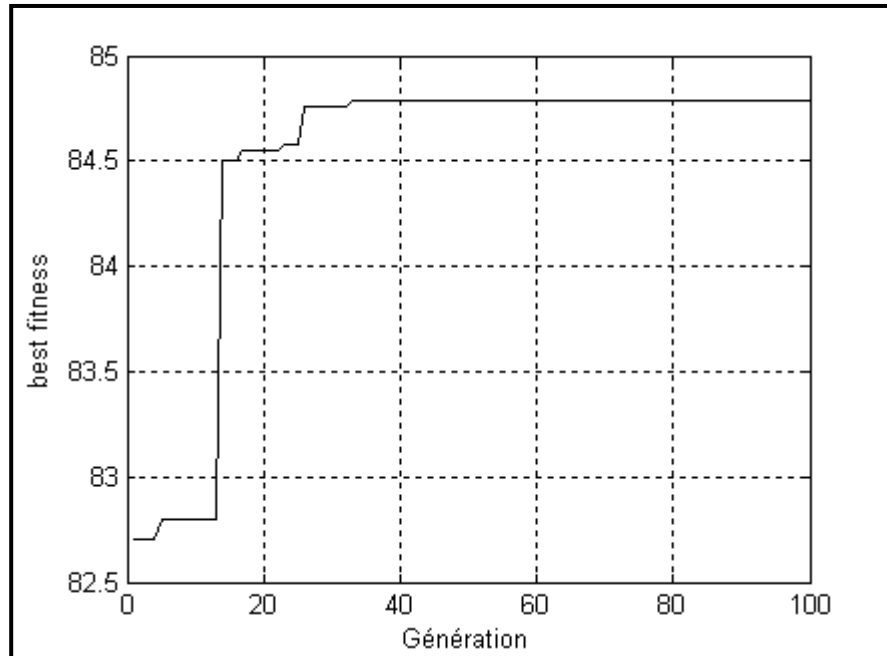
$$\begin{cases} D(t) = -c_1 z_1 + \frac{\mu(t)}{Y(S_{in} - (1+r)s(t))} x(t) \\ W(t) = -c_2 z_2 + \frac{k_0\mu(t)}{Y\alpha(c(t) - c_s)} x(t) + \frac{D(t)(1+r)c(t) - D(t)c_{in}}{\alpha(c(t) - c_s)} \end{cases} \quad (3.22)$$

Les paramètres  $c_i$  sont déterminés par les algorithmes génétiques simples et pour un  $\Gamma = [10^4 \quad 1 \quad 10]$  avec l'intervalle de recherche tel que  $c_i \in [0.01 \quad 1]$

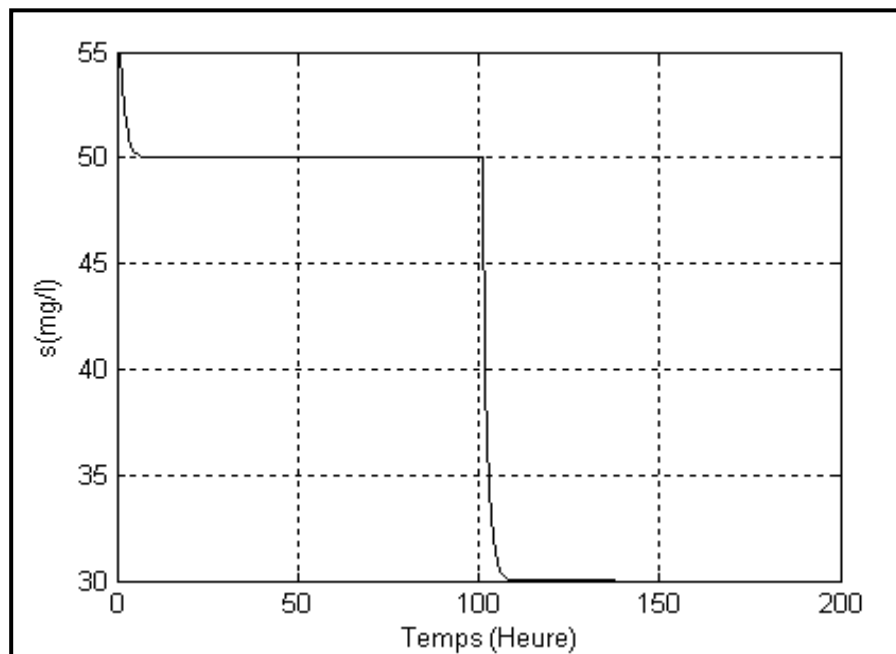
### III.1.4.1. Résultats de simulation

La figure (III.5) montre l'évolution de la fonction d'adaptation, d'après cette figure on constate que l'algorithme génétique simple converge après 40 générations. L'optimisation par les algorithmes génétiques simples nous a permis d'extraire les paramètres de conception  $C = [0.53 \quad 0.99]$ . Les figures (III.6) et (III.7) montrent le substrat et la concentration d'oxygène dans le réacteur. Ces deux grandeurs suivent leurs références sans dépassement et sans erreur statique. Le taux de dilution et le débit d'air sont illustrés par

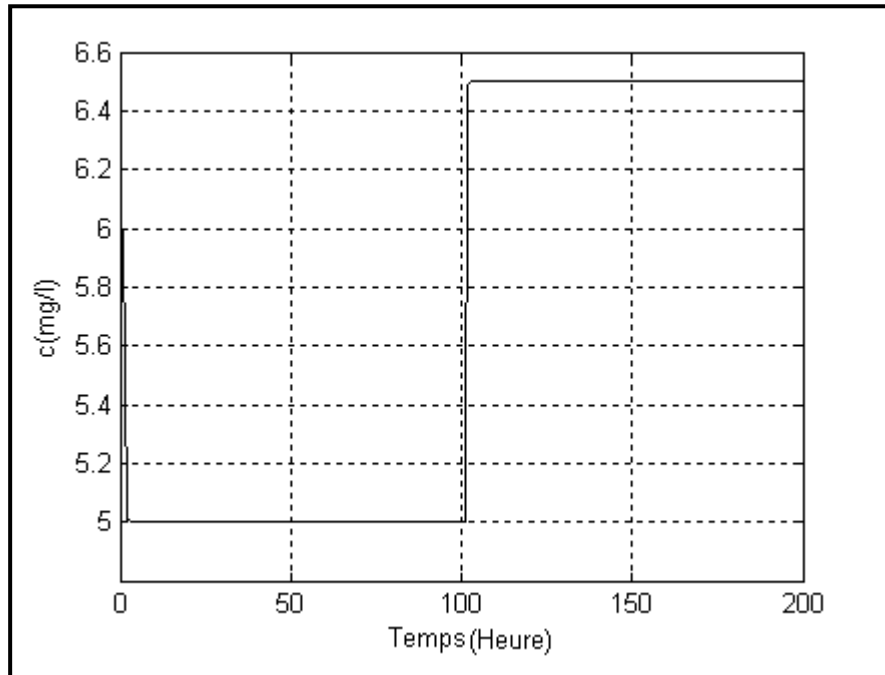
les figures (III.8) et (III.9) respectivement. Les figures (III.10) et (III.11) montrent l'évolution de la biomasse et la biomasse recyclée respectivement, le taux de croissance est donné par la figure (III.12).



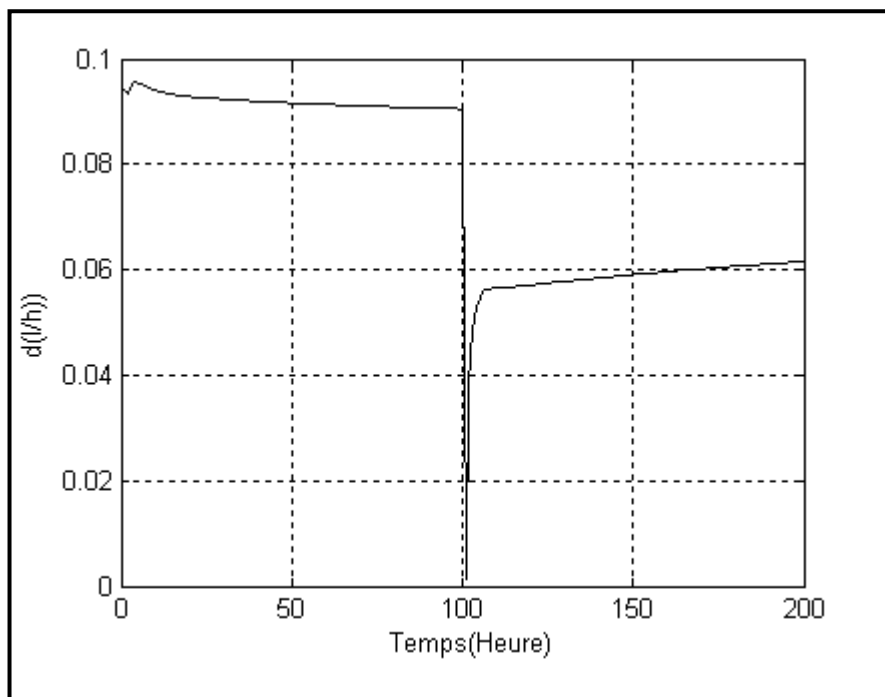
**Figure III.5 : Evolution de la fonction d'adaptation**



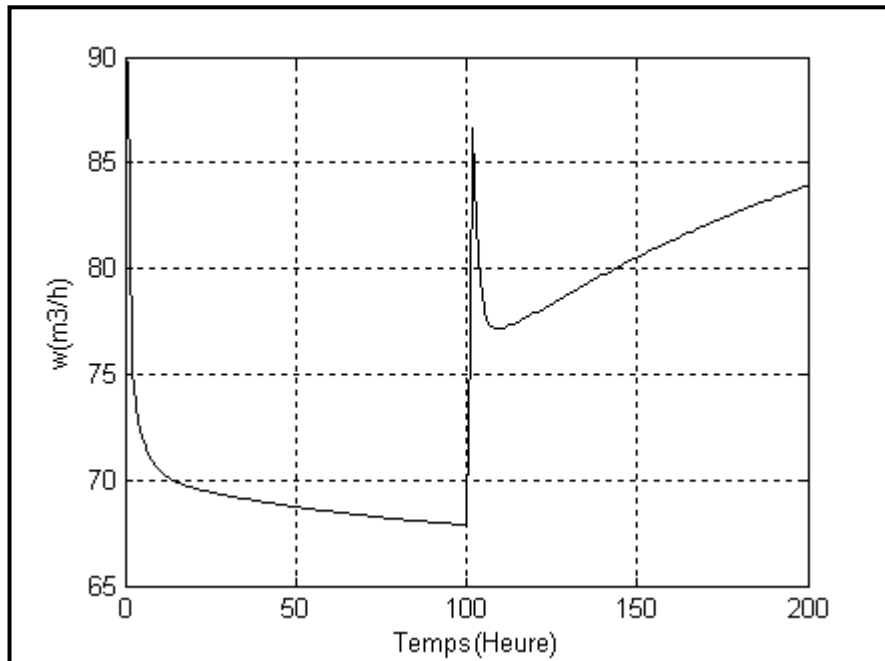
**Figure III.6 : Le substrat**



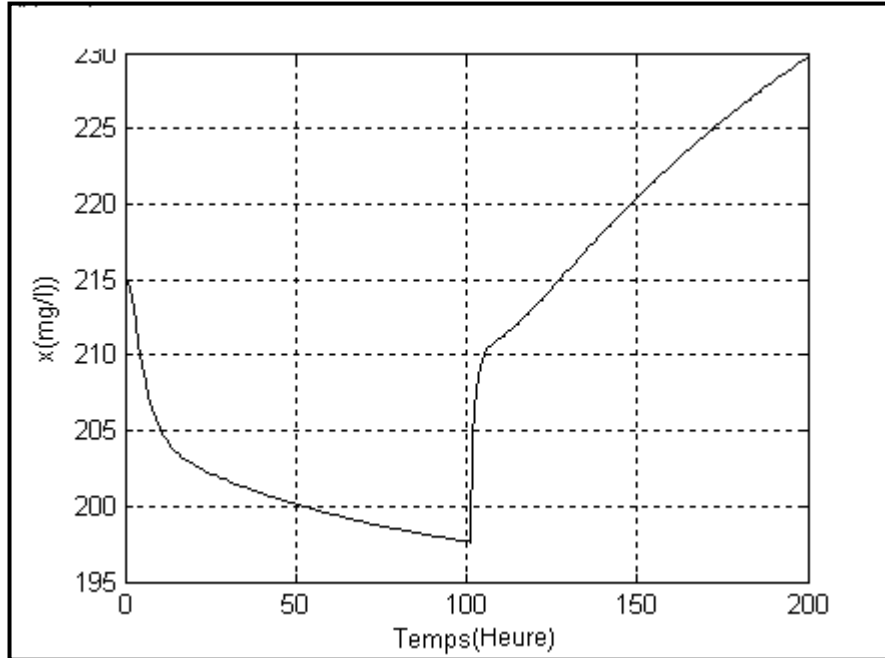
**Figure III.7 : La concentration d'oxygène**



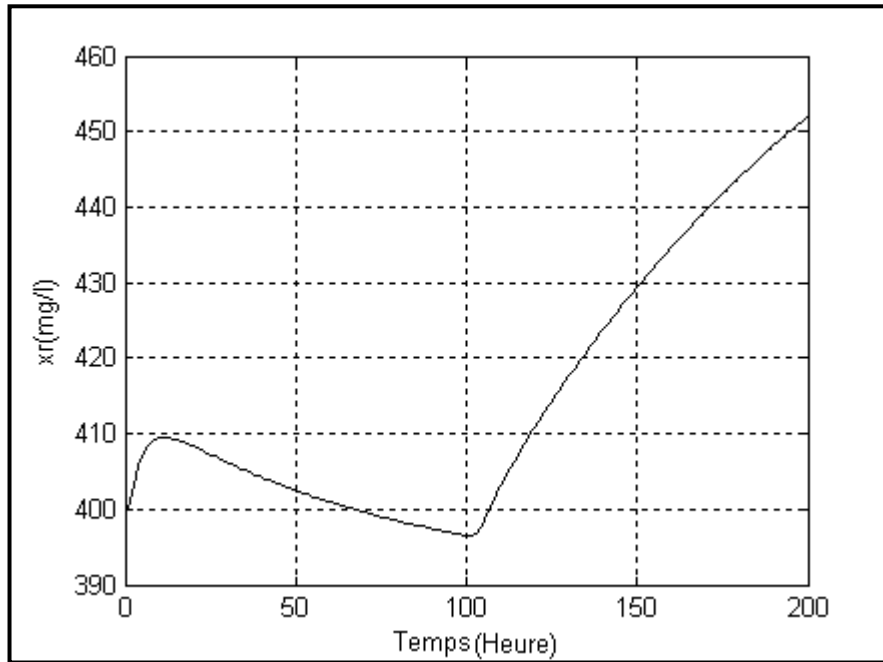
**Figure III.8 : Le taux de dilution**



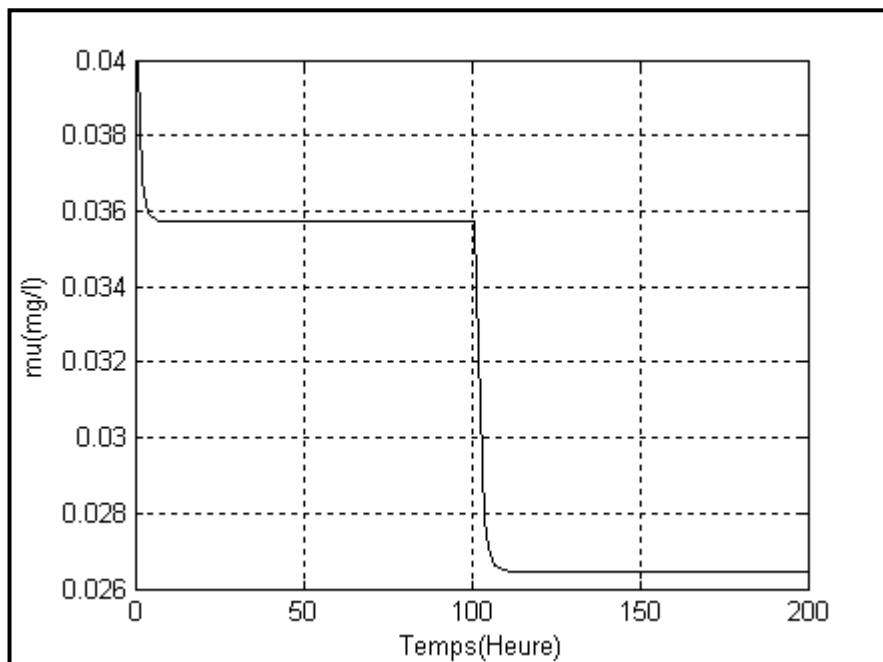
**Figure III.9 : Le débit d'air**



**Figure III.10 : La biomasse**



**Figure III.11 : La biomasse recyclée**



**Figure III.12 : Le taux de croissance**

### III.1.4.2. Test de robustesse

Pour tester la robustesse du contrôleur multivariable on introduit une perturbation sur la constante  $c_{in}$ . Dans la figure (III.13) on peut constater que cette augmentation perturbe le régime permanent pour le premier intervalle de temps : une erreur statique relativement importante apparaît dans la concentration d'oxygène.

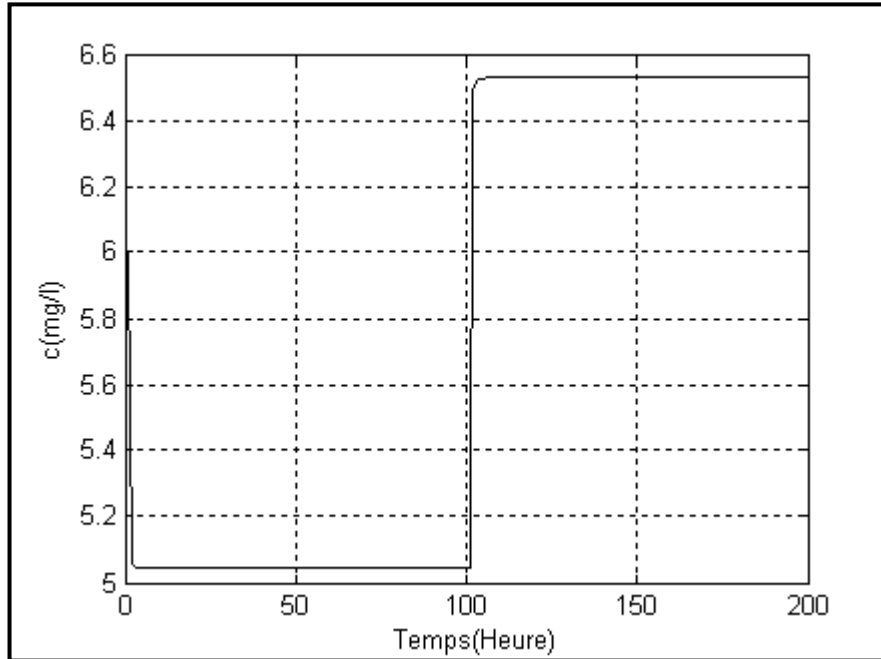


Figure III.13 : La concentration d'oxygène pour  $c_{in} = 1$

### III.1.5. Découplage par un superviseur flou

#### a). Les contrôleurs monovariables

Dans ce cas on considère les deux sous- systèmes monovariables suivants :

$$\begin{cases} \dot{s}(t) = -\frac{\mu(t)x(t)}{Y} - D(t)(1+r)s(t) + D(t)S_{in} \\ \dot{c}(t) = -\frac{K_0\mu(t)x(t)}{Y} + \alpha W(c_s - c(t)) \end{cases} \quad (3.23)$$

Pour les états définis par les équations (3.21) on considère les deux fonctions de Lyapunov suivantes :

$$\begin{cases} V_1(z_1) = 0.5z_1^2 \\ V_2(z_2) = 0.5z_2^2 \end{cases}$$



La condition de négativité s'écrit alors:

$$\begin{cases} \dot{V}_1 \leq 0 \Rightarrow z_1 \dot{z}_1 \leq 0 \\ \dot{V}_2 \leq 0 \Rightarrow z_2 \dot{z}_2 \leq 0 \end{cases}$$

Alors:

$$\begin{cases} \dot{z}_1 = -c_1 z_1 \\ \dot{z}_2 = -c_2 z_2 \end{cases}$$

Ce qui conduit à la réalisation des deux lois de commande telles que:

$$\begin{cases} D(t) = -c_1 z_1 + \frac{\mu(t)}{Y(S_{in} - (1+r)s(t))} x(t) \\ W = -c_2 z_2 + \frac{k_0 \mu(t)}{Y\alpha(c(t) - c_s)} x(t) \end{cases} \quad (3.24)$$

### ***b) Le superviseur flou***

Les deux lois données par (3.24) définissent les entrées principales d'un superviseur flou, néanmoins, la deuxième commande n'affecte pas l'évolution du substrat, ce qui mène à un système flou réduit ayant une seule entrée. L'algorithme de raisonnement est basé sur trois variables linguistiques,  $n_k = 3$  et il est défini par les trois règles suivantes:

$$\begin{cases} R_1 : \text{if } D(t) \text{ is } A_{11} \text{ then } u_{c1} = p_{11} D(t) \\ R_2 : \text{if } D(t) \text{ is } A_{21} \text{ then } u_{c1} = p_{21} D(t) \\ R_3 : \text{if } D(t) \text{ is } A_{31} \text{ then } u_{c1} = p_{31} D(t) \end{cases} \quad (3.25)$$

Les intervalles de recherche sont:

$$c_i \in [0.01 \quad 1], p_{i1} \in [-400 \quad +400] \text{ et } u_{d1} \in [0.01 \quad 1]$$

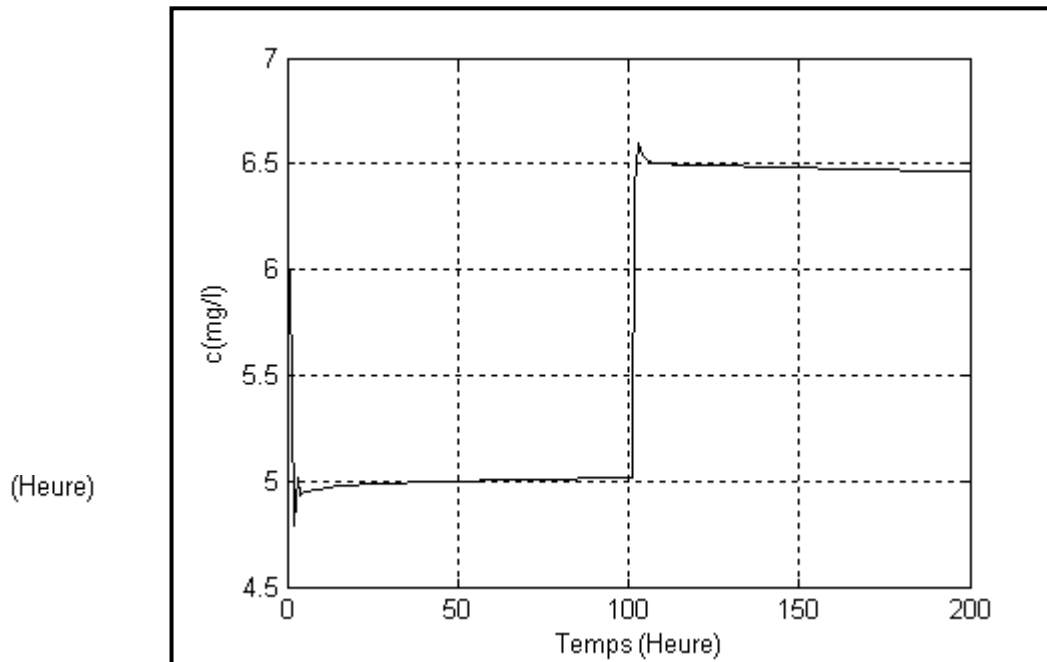
### **III.1.5.1. Résultats de simulation**

Les paramètres de conceptions obtenus par les MOGA sont:

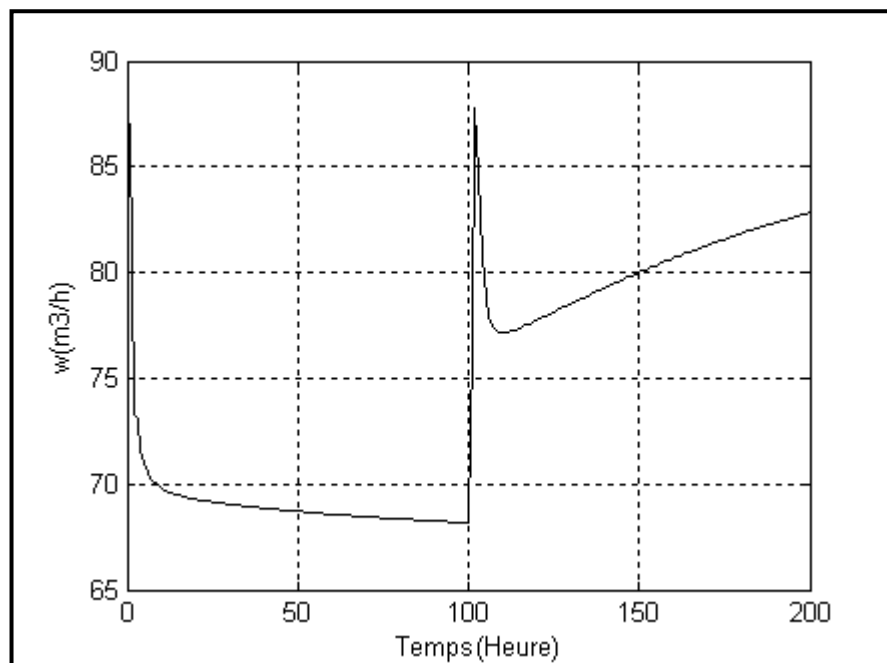
$$C = [0.5 \quad 0.86], P_1 = [-341.35 \quad 274.88 \quad -250.64] \text{ et } u_{d1} = 0.5$$

La figure (III.14) montre la concentration d'oxygène, la figure (III.15) montre la commande. Selon la figure (III.14), on peut constater que le FLS tente de compenser l'effet du taux de dilution  $D(t)$  dans les premières heures (régime transitoire), cette compensation est assurée pour que le système puisse atteindre l'objectif voulu. Les

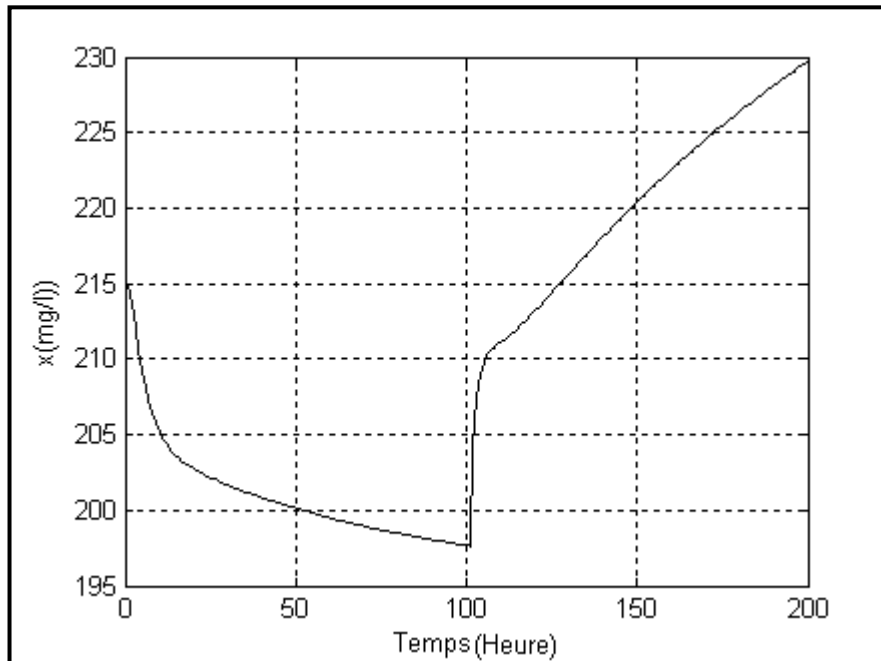
figures (III.16), (II.17) et (III.18) schématisent le taux de croissance, la biomasse et la biomasse recyclée respectivement.



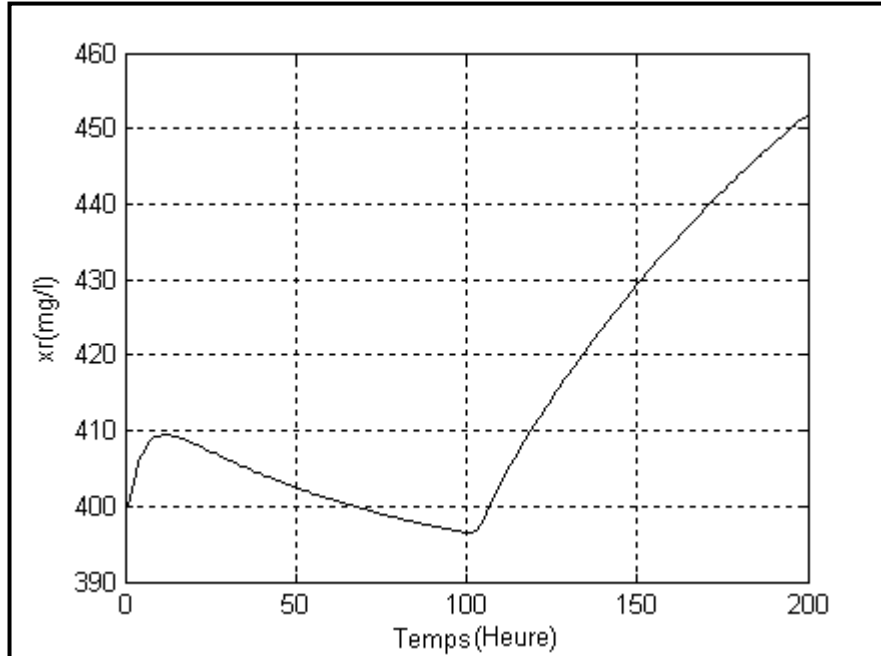
**Figure III.14 : La concentration d'oxygène**



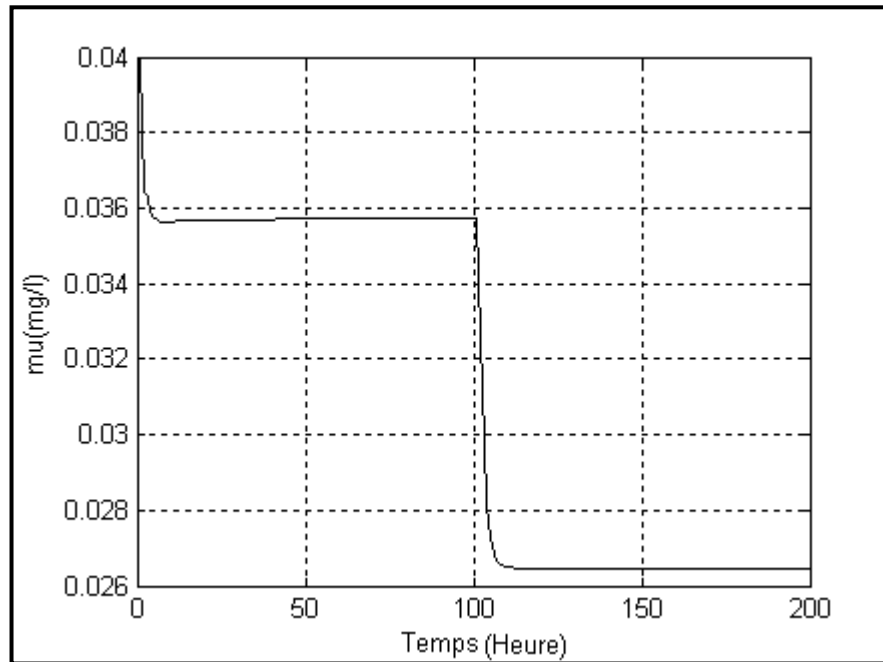
**Figure III.15 : Le débit d'air**



**Figure III.16 : La biomasse**



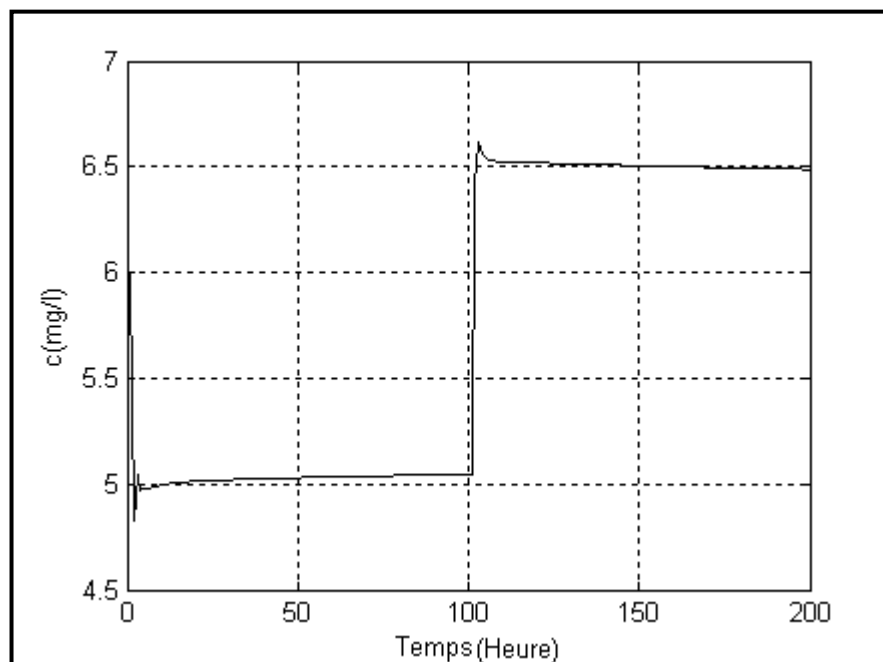
**Figure III.17: La biomasse recyclée**



**Figure III.18: Le taux de croissance**

### III.1.5.2. Test de robustesse

Pour montrer l'efficacité de cette méthode, on introduit la même perturbation sur le paramètre  $c_{in}$ . Le résultat est donné par la figure (III.19) où on constate que le comportement de la concentration d'oxygène n'est pas dégradé ;

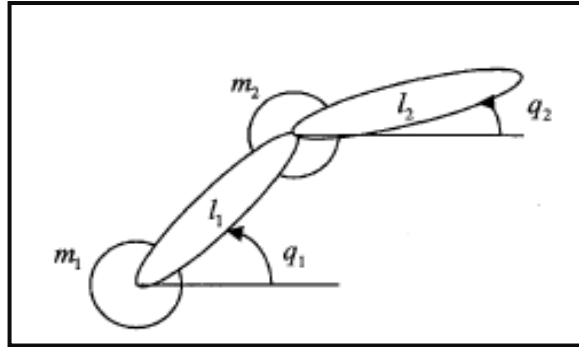


**Figure III.19 : La concentration d'oxygène pour  $c_{in} = 1$**

## III.2. Bras de robots à deux segments

### III.2.1. Modèle mathématique.

Le second procédé est un double bras de robots [60],[61],[62], le mécanisme est schématisé par la figure (III.20)



**Figure III.20: Double bras de robots**

La dynamique de ce système est donnée par

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (3.26)$$

avec :

$$M(q) = \begin{bmatrix} (m_1 + m_2)l_1^2 & m_2l_1l_2(s_1s_2 + c_1c_2) \\ m_2l_1l_2(s_1s_2 + c_1c_2) & m_2l_2^2 \end{bmatrix}$$

$$C(q, \dot{q}) = m_1l_1l_2(c_1s_2 - s_1c_2) \begin{bmatrix} 0 & -\dot{q}_2 \\ -\dot{q}_1 & 0 \end{bmatrix}$$

$$G(q) = \begin{bmatrix} -(m_1 + m_2)l_1gs_1 \\ -m_2l_2gs_2 \end{bmatrix}$$

$q_1, q_2$  représentent les positions angulaires (rad),  $M(q)$  représente le moment des inerties,  $C(q, \dot{q})$  contient les forces coriolis et centrifuges et  $G(q)$  indique les forces gravitationnelle,  $c_1 = \cos(q_1)$ ,  $c_2 = \cos(q_2)$ ,  $s_1 = \sin(q_1)$  et  $s_2 = \sin(q_2)$ ,  $\tau_1$  et  $\tau_2$  (Nm) indiquent les couples appliqués. Le tableau (III.4) donne les paramètres cinétiques.

Les références sont :

$$\begin{cases} y_{r1} = \frac{\pi}{6}(1 - \cos(2\pi t)) \\ y_{r2} = \frac{\pi}{4}(1 - \cos(2\pi t)) \end{cases}$$

| Tableau III.4<br>Paramètres cinétiques |                        |
|--|------------------------|
| Paramètre                              | Valeur                 |
| $m_1$                                  | 1 kg                   |
| $m_2$                                  | 10 kg                  |
| $l_1$                                  | 1 m                    |
| $l_2$                                  | 1 m                    |
| $g$                                    | $9.8 \text{ m/s}^{-2}$ |

Sous la représentation d'état, la dynamique peut s'écrire comme suit :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1 + h_{11}\tau_1 + h_{12}\tau_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2 + h_{21}\tau_1 + h_{22}\tau_2 \\ y_1 = x_1 \\ y_2 = x_3 \end{cases} \quad (3.27)$$

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

$$H = M^{-1}(q) \text{ et } F = M^{-1}(q)(-C(q, \dot{q})\dot{q} - G(q))$$

### III.2.2. L'approche directe : Contrôleur Backstepping multivariable.

L'établissement des lois de commandes passe par deux étapes ;

#### Etape 1 :

L'objectif étant de ramener les positions angulaires vers les signaux de références  $y_{r1}$  et  $y_{r2}$ . Pour cela, on définit les signaux d'erreur suivants

$$c \quad (3.28)$$

À ces deux états, on associe la fonction de Lyapunov

$$V_1 = 0.5Z_{11}^2 + 0.5Z_{21}^2$$

La condition de négativité s'écrit alors :

$$\dot{V}_1 \leq 0 \Rightarrow z_{11}\dot{z}_{11} + z_{21}\dot{z}_{21} \leq 0$$

On pose alors :

$$\begin{cases} \dot{z}_{11} = -c_1 z_{11} \\ \dot{z}_{21} = -c_2 z_{21} \end{cases}$$

Selon (3.27) et (3.28) on peut extraire les commandes virtuelles suivantes:

$$\begin{cases} x_2 = -c_1 z_{11} + \dot{y}_{r1} = \alpha_{11}(z_{11}) \\ x_4 = -c_2 z_{21} + \dot{y}_{r2} = \alpha_{21}(z_{21}) \end{cases}$$

**Etape 2 :**

On introduit de nouvelles variables telles que :

$$\begin{cases} z_{12} = x_2 - \alpha_{11}(z_{11}) \\ z_{22} = x_4 - \alpha_{21}(z_{21}) \end{cases} \quad (3.29)$$

Alors, le système (3.27) se met sous la forme :

$$\begin{cases} \dot{z}_{11} = -c_1 z_{11} + z_{12} \\ \dot{z}_{12} = f_1 + h_{11}\tau_1 + h_{12}\tau_2 - \dot{\alpha}_{11} \\ \dot{z}_{21} = -c_2 z_{21} + z_{22} \\ \dot{z}_{22} = f_2 + h_{21}\tau_1 + h_{22}\tau_2 - \dot{\alpha}_{21} \end{cases} \quad (3.30)$$

Pour ce nouveau système, on définit la fonction de Lyapunov suivante

$$V_2 = V_1(z_{11}, z_{21}) + 0.5z_{12}^2 + 0.5z_{22}^2$$

La condition de négativité s'écrit alors :

$$\begin{aligned} \dot{V}_2 \leq 0 &\Rightarrow \frac{dV_1}{dz_{11}} \dot{z}_{11} + \frac{dV_1}{dz_{21}} \dot{z}_{21} \leq 0 \\ &\Rightarrow z_{11}\dot{z}_{11} + z_{21}\dot{z}_{21} + z_{12}\dot{z}_{12} + z_{22}\dot{z}_{22} \leq 0 \\ &\Rightarrow -c_1 z_{11}^2 - c_2 z_{21}^2 + z_{12}(z_{11} + \dot{z}_{12}) + z_{22}(z_{21} + \dot{z}_{22}) \leq 0 \end{aligned}$$

Soit alors, pour réaliser cette condition :

$$\begin{cases} \dot{z}_{12} + z_{11} = -c_3 z_{12} \\ \dot{z}_{22} + z_{21} = -c_4 z_{22} \end{cases}$$

Selon (3.30) on a :

$$\begin{cases} h_{11}\tau_1 + h_{12}\tau_2 = -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} - f_1 \\ h_{21}\tau_1 + h_{22}\tau_2 = -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} - f_2 \end{cases}$$

Ce qui permet de déduire les deux couples

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = M(\tilde{Z} - F) \quad (3.31)$$

Telle que :

$$\begin{cases} \tilde{Z} = \begin{bmatrix} -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} \\ -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} \end{bmatrix} \\ \dot{\alpha}_{11} = \frac{d\alpha_{11}}{dz_{11}} \dot{z}_{11} + \ddot{y}_{r1} = c_1^2 z_{11} - c_1 z_{12} + \ddot{y}_{r1} \\ \dot{\alpha}_{21} = \frac{d\alpha_{21}}{dz_{21}} \dot{z}_{21} + \ddot{y}_{r2} = c_2^2 z_{21} - c_2 z_{22} + \ddot{y}_{r2} \end{cases} \quad (3.32)$$

Avec un vecteur  $\Gamma = [10^4 \quad 1 \quad 1]$  et avec un intervalle de recherche tel que  $c_i \in [1 \quad 50]$ ,

### III.2.2.1. Résultats de simulation

Les paramètres de conception générés par les algorithmes génétiques sont:  $c = [18.53 \quad 17.29 \quad 20.40 \quad 26.58]$ . Les positions angulaires et les vitesses angulaires sont illustrées par les figures (III.21) et (III.22) respectivement. On constate que la poursuite est réalisée dans un temps inférieur à 0.3 sec et avec des couples convenables donnés par les figures (III.23) et (III.24). Les vitesses angulaires sont illustrées par les figures (III.25) et (III.26).

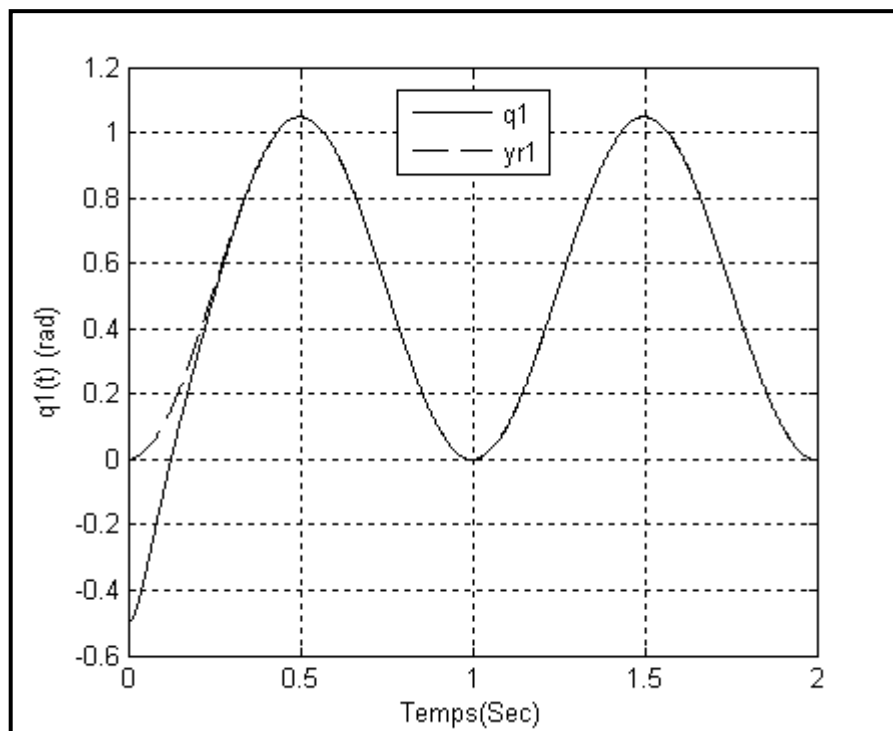
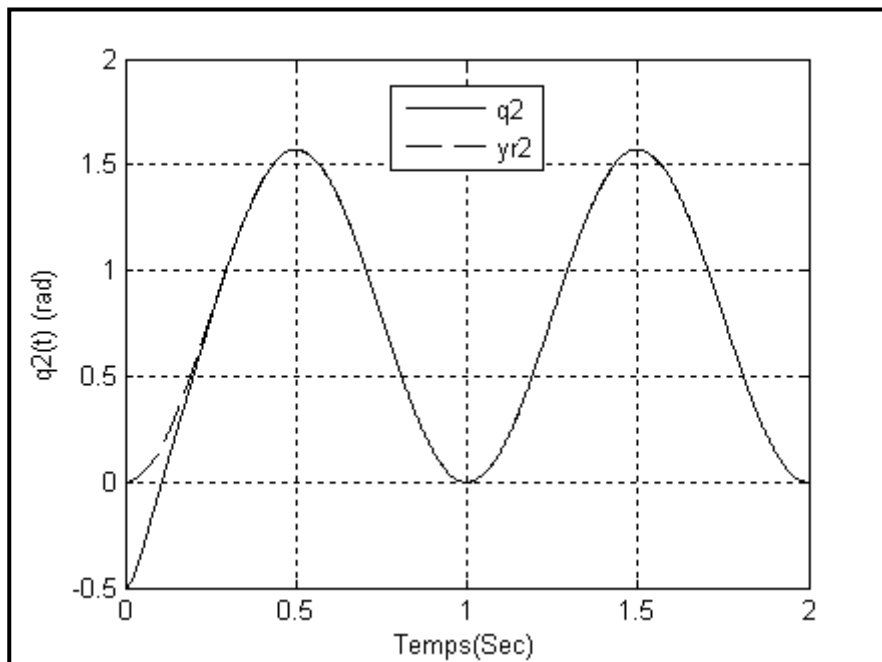
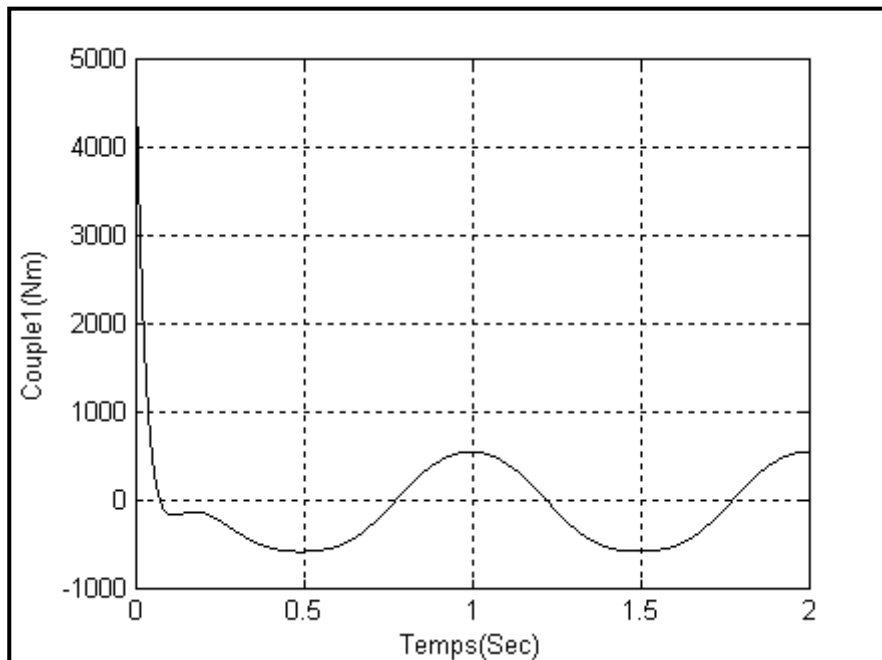


Figure III.21 : La position angulaire  $q_1$

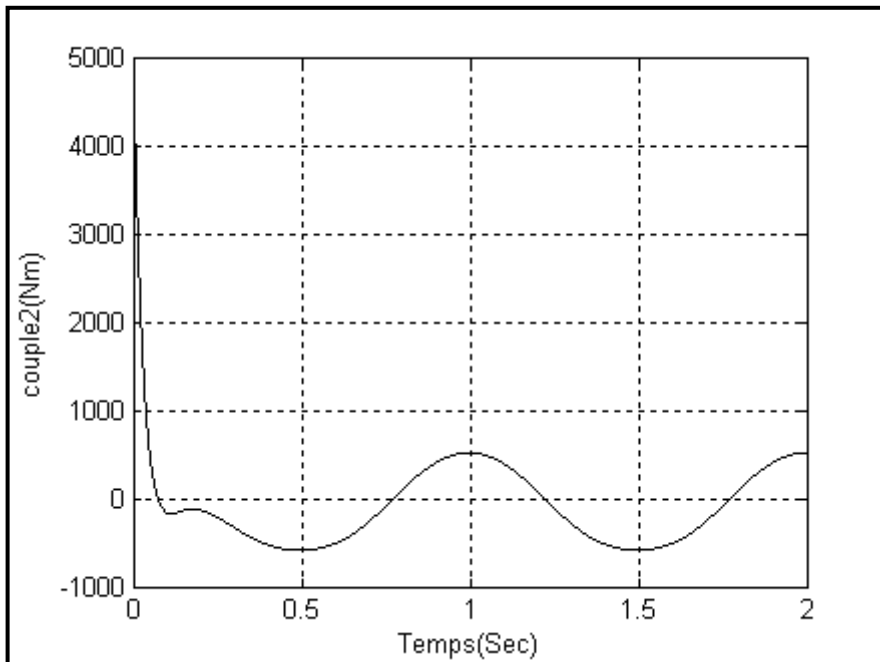




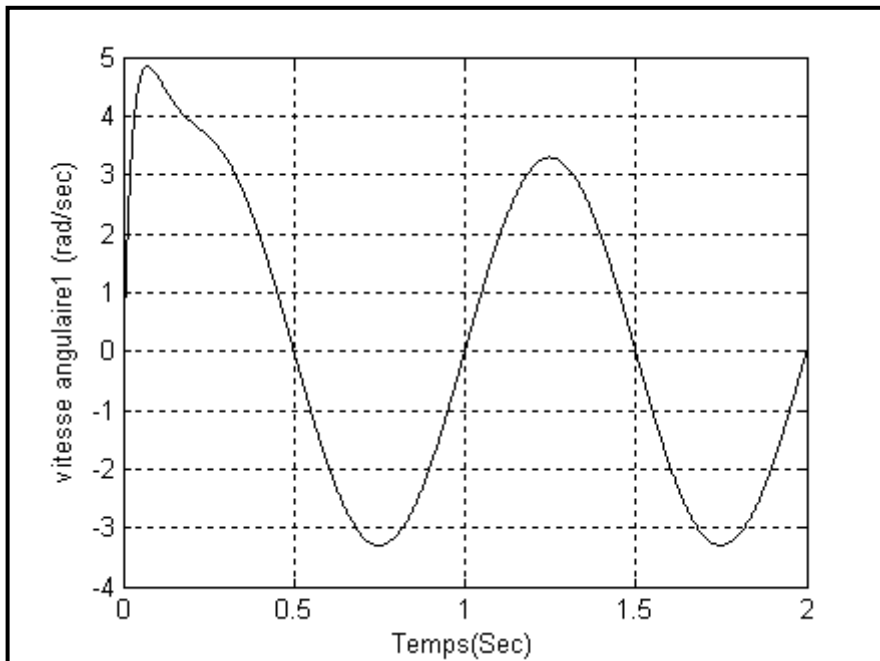
**Figure III.22 : La position angulaire  $q_2$**



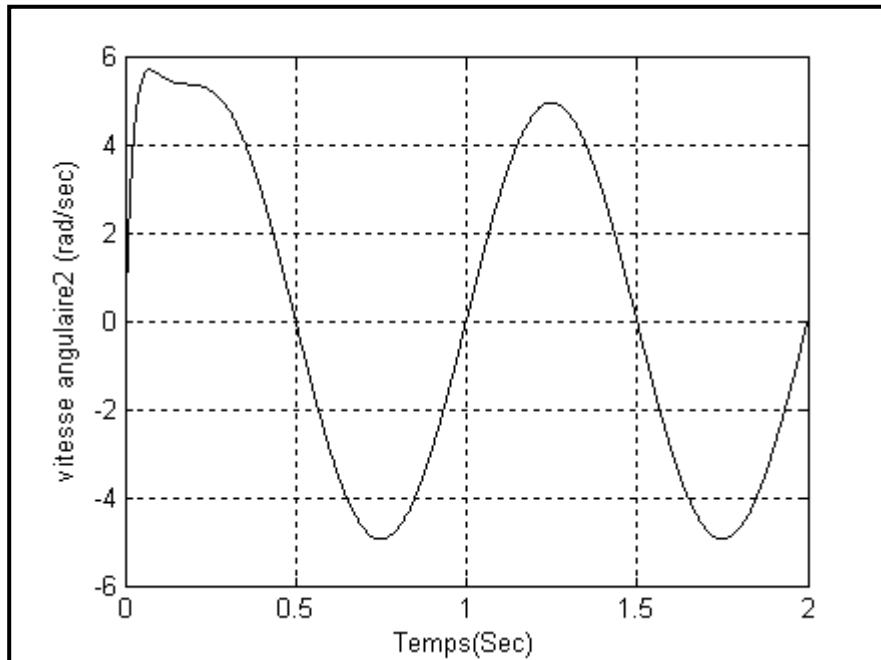
**Figure III.23 : Le couple appliqué  $\tau_1$**



**Figure III.24: Le couple appliqué  $\tau_2$**



**Figure III.25 : La vitesse angulaire  $\dot{q}_1$**

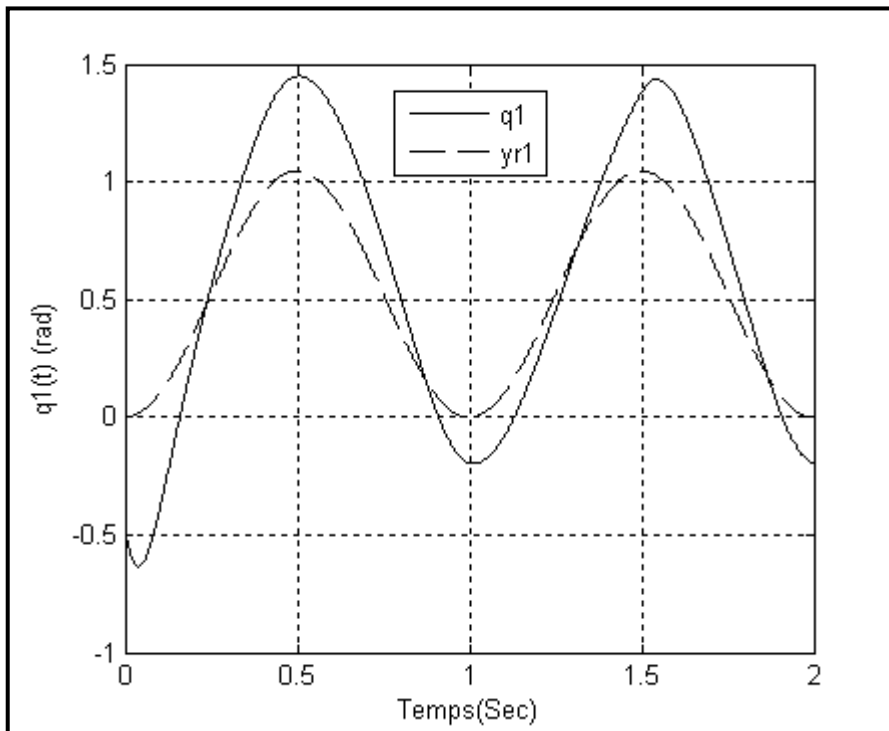


**Figure III.26 : La vitesse angulaire  $\dot{q}_2$**

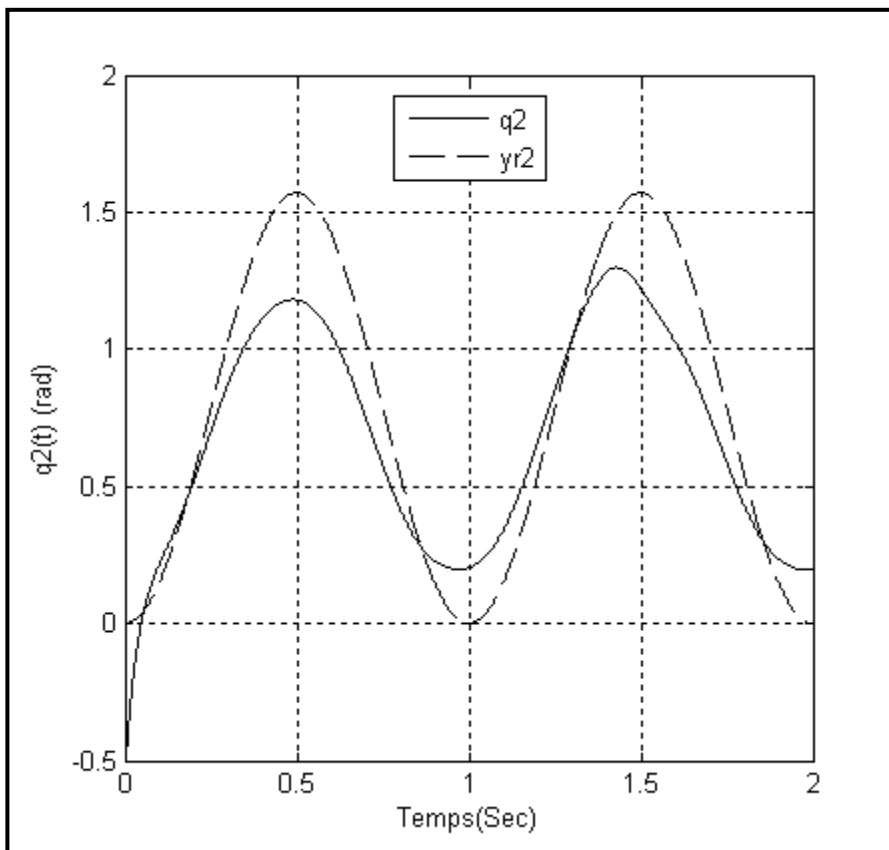
### III.2.2.2. Test de robustesse

Pour tester la robustesse de la commande backstepping multivariable on considère d’abord un changement dans la longueur du deuxième bras, soit alors  $l_2 = 0.5$ , ce paramètre intervient dans les formules d’interactions et donc on procède par l’augmentation de leurs effet.

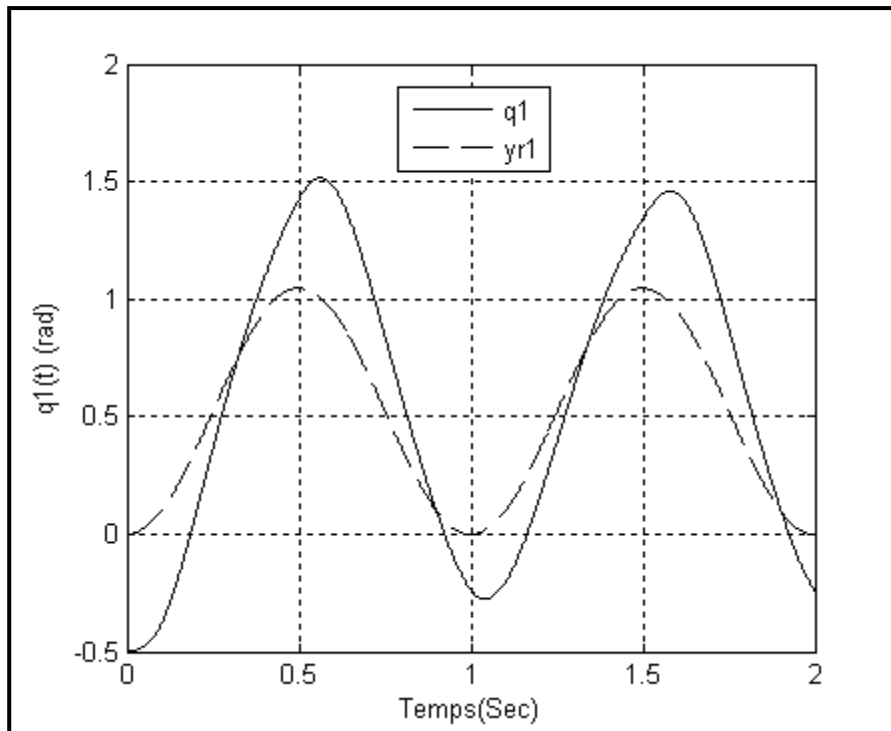
D’après les figures (III.27) et (III.28), ce changement affecte les deux bras simultanément. Cependant, un changement de masse du premier bras, soit  $m_1 = 10$ , affecte directement la réponse du deuxième bras d’après les figures (III.29) et (III.30)



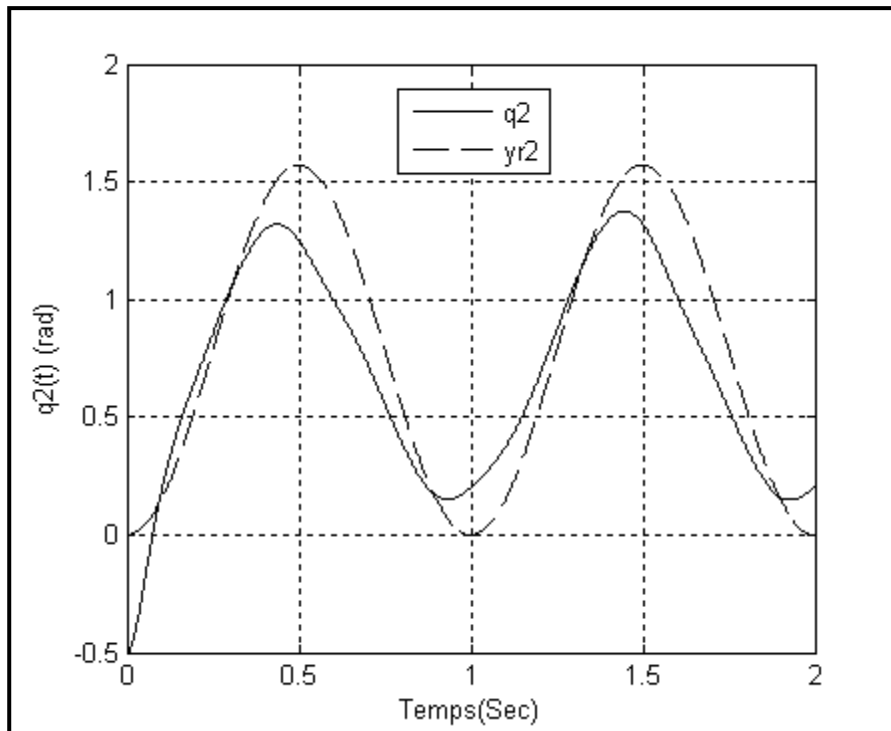
**Figure III.27 : La position angulaire  $q_1$  pour  $l_2 = 0.5$**



**Figure III.28 : La position angulaire  $q_2$  pour  $l_2 = 0.5$**



**Figure III.29 : La position angulaire  $q_1$  pour  $m_1 = 10$**



**Figure III.30 : La position angulaire  $q_2$  pour  $m_1 = 10$**

### III.2.3. Découplage par un superviseur flou.

#### a) les contrôleurs monovariables

En ignorant les interactions et selon (3.7) le système se divise en deux sous-systèmes tels que:

$$\begin{aligned} \Sigma_{s1} : \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1 + h_{11}\tau_1 \\ y_1 = x_1 \end{cases} \\ \Sigma_{s2} : \begin{cases} \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2 + h_{22}\tau_2 \\ y_2 = x_3 \end{cases} \end{aligned} \quad (3.33)$$

#### Etape 1 :

Selon les états définis par (3.28), on définit deux fonctions de Lyapunov pour les deux sous-systèmes ci-dessus

$$\begin{cases} V_1(z_{11}) = 0.5z_{11}^2 \\ V_2(z_{21}) = 0.5z_{21}^2 \end{cases}$$

Les conditions de négativité s'écrivent alors :

$$\begin{cases} \dot{V}_1 \leq 0 \Rightarrow z_{11}\dot{z}_{11} \leq 0 \\ \dot{V}_2 \leq 0 \Rightarrow z_{21}\dot{z}_{21} \leq 0 \end{cases}$$

Alors :

$$\begin{cases} \dot{z}_{11} = -c_1z_{11} \\ \dot{z}_{21} = -c_2z_{21} \end{cases}$$

Selon (3.28) et (3.33) on peut extraire les commandes virtuelles suivantes:

$$\begin{cases} x_2 = -c_1z_{11} + \dot{y}_{r1} = \alpha_{11}(z_{11}) \\ x_4 = -c_2z_{21} + \dot{y}_{r2} = \alpha_{21}(z_{21}) \end{cases}$$

#### Etape 2 :

On introduit un changement d'état tel que :

$$\begin{cases} z_{12} = x_2 - \alpha_{11}(z_{11}) \\ z_{22} = x_4 - \alpha_{21}(z_{21}) \end{cases} \quad (3.34)$$

Alors, le système (3.33) se met sous la forme :

$$\begin{aligned} \Sigma_{s1} : \begin{cases} \dot{z}_{11} = -c_1 z_{11} + z_{12} \\ \dot{z}_{12} = f_1 + h_{11} \tau_1 - \dot{\alpha}_{11} \end{cases} \\ \Sigma_{s2} : \begin{cases} \dot{z}_{21} = -c_2 z_{21} + z_{22} \\ \dot{z}_{22} = f_2 + h_{22} \tau_2 - \dot{\alpha}_{21} \end{cases} \end{aligned} \quad (3.35)$$

Pour ce nouveau système, on définit les deux fonctions de Lyapunov suivantes

$$\begin{cases} V_3 = V_1(z_{11}) + 0.5z_{12}^2 \\ V_4 = V_2(z_{21}) + 0.5z_{22}^2 \end{cases}$$

Les conditions de négativité s'écrivent alors :

$$\begin{aligned} \dot{V}_3 \leq 0 &\Rightarrow \frac{dV_1}{dz_{11}} \dot{z}_{11} + \frac{dV_1}{dz_{12}} \dot{z}_{12} \leq 0 \\ &\Rightarrow z_{11} \dot{z}_{11} + z_{12} \dot{z}_{12} \leq 0 \\ &\Rightarrow -c_1 z_{11}^2 + z_{12}(z_{11} + \dot{z}_{12}) \leq 0 \end{aligned}$$

Et

$$\begin{aligned} \dot{V}_4 \leq 0 &\Rightarrow \frac{dV_2}{dz_{21}} \dot{z}_{21} + \frac{dV_2}{dz_{22}} \dot{z}_{22} \leq 0 \\ &\Rightarrow z_{21} \dot{z}_{21} + z_{22} \dot{z}_{22} \leq 0 \\ &\Rightarrow -c_2 z_{21}^2 + z_{22}(z_{21} + \dot{z}_{22}) \leq 0 \end{aligned}$$

Soit pour réaliser ces deux conditions:

$$\begin{cases} \dot{z}_{12} + z_{11} = -c_3 z_{12} \\ \dot{z}_{22} + z_{21} = -c_4 z_{22} \end{cases}$$

Selon (3.35) on a :

$$\begin{cases} h_{11} \tau_1 = -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} - f_1 \\ h_{21} \tau_1 = -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} - f_2 \end{cases}$$

Ce qui donne :

$$\begin{cases} \tau_1 = \frac{-c_{12} z_{12} - z_{11} + \dot{\alpha}_1 - f_1}{h_{11}} \\ \tau_2 = \frac{-c_{22} z_{22} - z_{21} + \dot{\alpha}_2 - f_2}{h_{22}} \end{cases} \quad (3.36)$$

### ***b) Le superviseur flou***

Le superviseur flou assurant le découplage est gouverné par l'ensemble des règles de décisions définies par (3.37) et (3.38) ci-dessous et pour un  $n_k = 3$

$$\begin{cases} R_{11} : \text{if } \tau_1 \text{ is } A_{11} \text{ then } u_{c1} = p_{11}\tau_1 \\ R_{21} : \text{if } \tau_1 \text{ is } A_{21} \text{ then } u_{c1} = p_{21}\tau_1 \\ R_{31} : \text{if } \tau_1 \text{ is } A_{31} \text{ then } u_{c1} = p_{31}\tau_1 \end{cases} \quad (3.37)$$

$$\begin{cases} R_{12} : \text{if } \tau_2 \text{ is } A_{12} \text{ then } u_{c2} = p_{12}\tau_2 \\ R_{22} : \text{if } \tau_2 \text{ is } A_{22} \text{ then } u_{c2} = p_{22}\tau_2 \\ R_{32} : \text{if } \tau_2 \text{ is } A_{32} \text{ then } u_{c2} = p_{32}\tau_2 \end{cases} \quad (3.38)$$

Les intervalles de recherche sont tels que :

$$c_i \in [1 \quad 50] \quad p_{ik} \in [-3 \quad +3], k=1,2 \quad u_{d_i} \in [0.01 \quad 500]$$

### III.2.3.1. Résultats de simulation

L'optimisation par les MOGA permet d'avoir :

$$C = [22.17 \quad 43.83 \quad 48.99 \quad 46.93],$$

$$P_1 = [2.17 \quad 2.88 \quad 2.86], P_2 = [-2.16 \quad -2.97 \quad -2.81] \text{ et } U_d = [484.36 \quad 499.51]$$

Les positions angulaires sont données par les figures (III.31) et (III.32), une petite distorsion dans le régime transitoire et une erreur de poursuite non nulle dans le régime permanent comparativement à la méthode directe. Les couples appliqués ont illustrés par les figures (III.33) et (III.34) et les vitesses angulaires sont données par les figures (III.35) et (III.36)

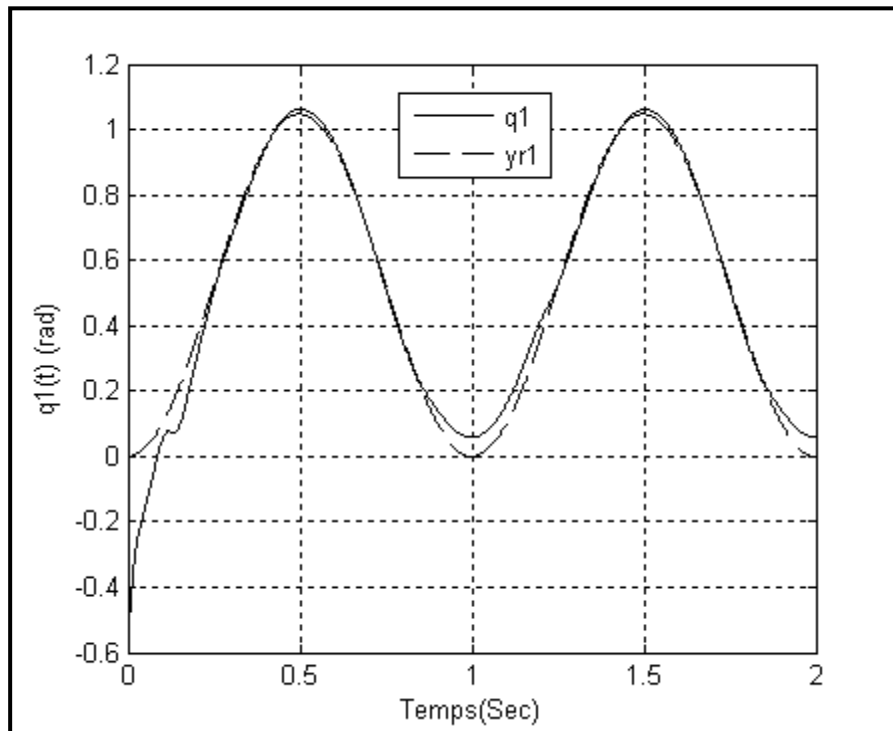


Figure III.31 : La position angulaire  $q_1$



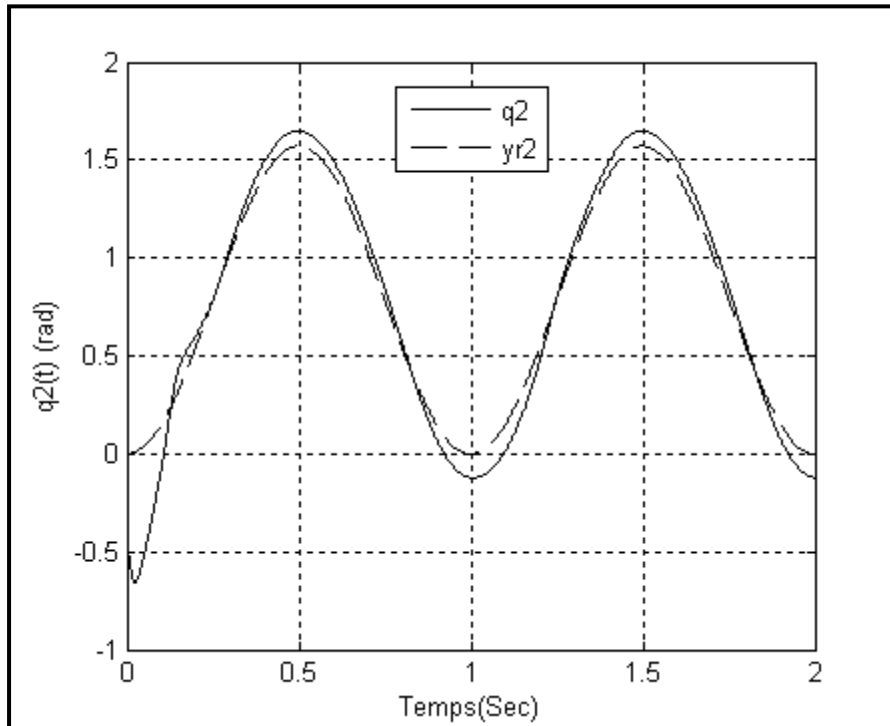


Figure III.32 : La position angulaire  $q_2$

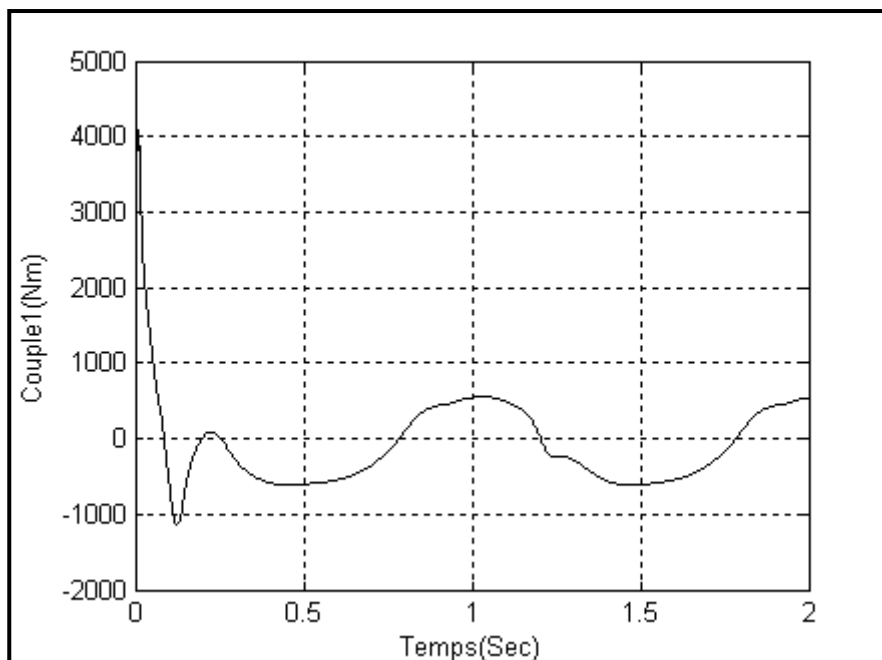
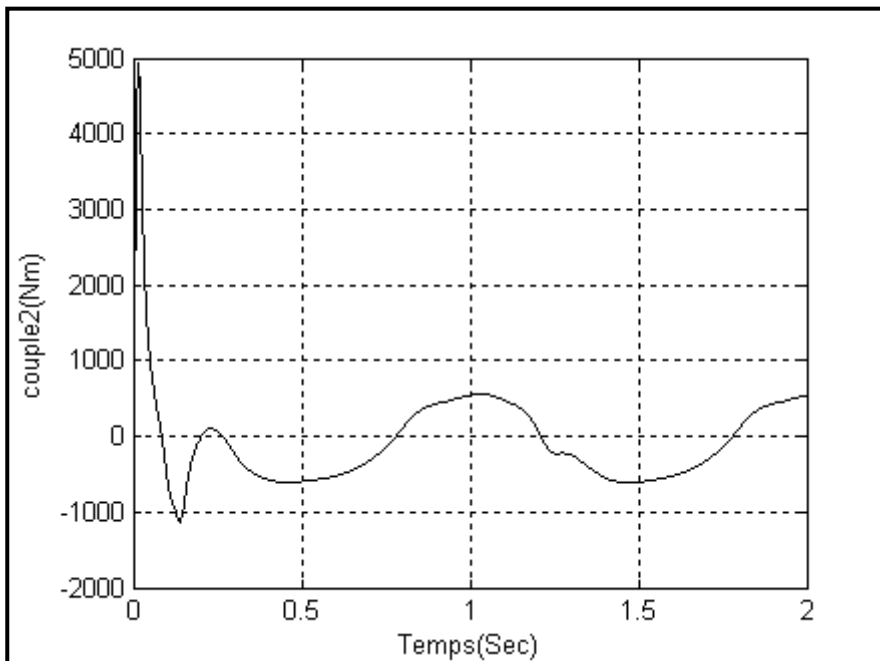
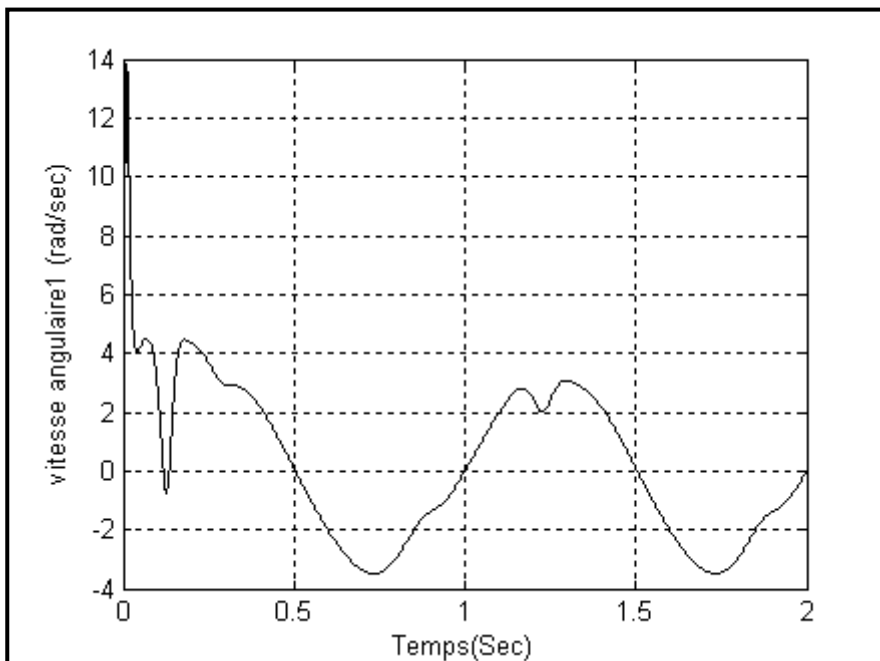


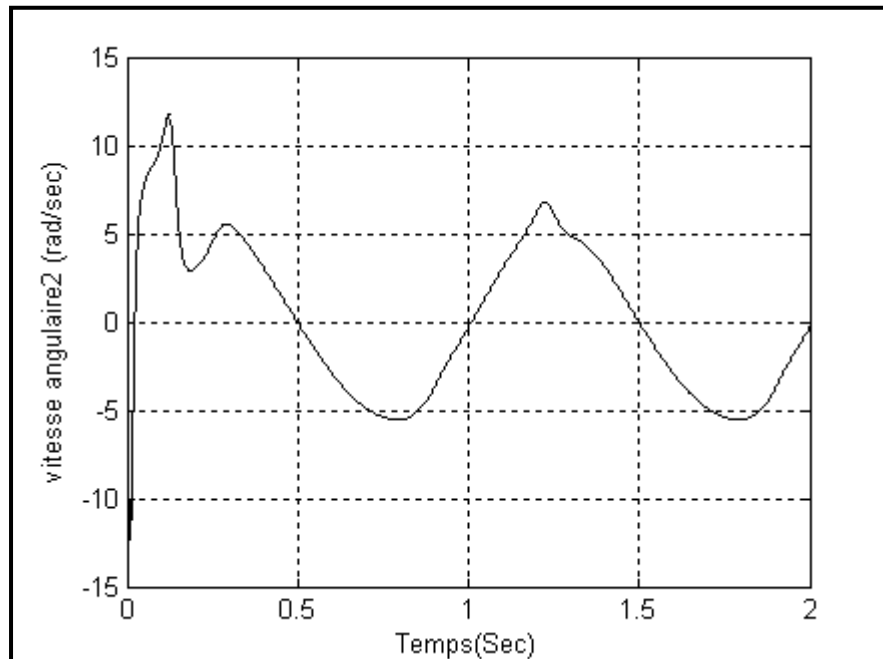
Figure III.33 : Le couple  $\tau_1$



**Figure III.34 : Le couple  $\tau_2$**



**Figure III.35 : La vitesse angulaire  $\dot{q}_1$**

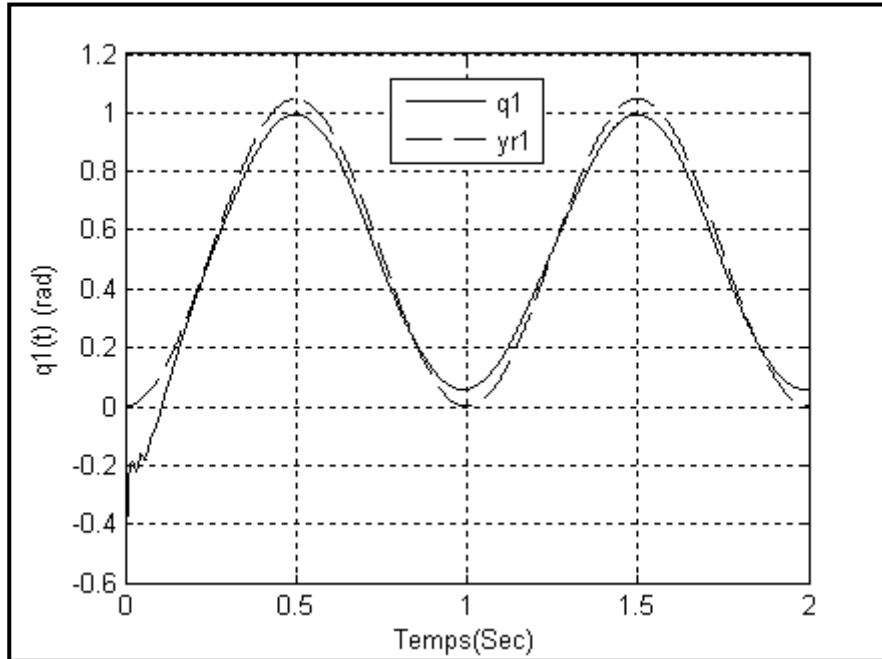


**Figure III.36 : La vitesse angulaire  $\dot{q}_2$**

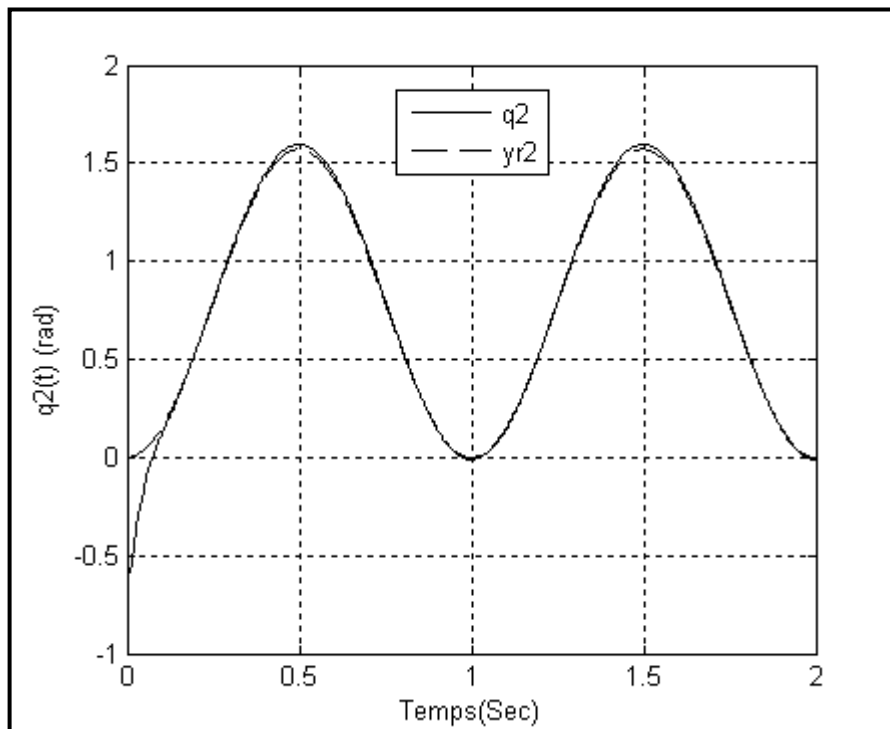
### III.2.3.2. Test de robustesse

Pour monter la robustesse du découplage par le superviseur flou, on introduit les mêmes perturbations que pour le contrôleur multivariable. Les figures (III.37) et (III.38) montrent les positions angulaires pour une longueur  $l_2 = 0.5$ , on constate que le superviseur permet de générer les commandes compensatrices adéquates pour atténuer l'effet des perturbations. Cependant, et selon les figures (III.39) et (III.40), les couples appliqués dans ce cas sont très élevés ce qui interpréter par la présence de la longueur  $l_2$  dans les fonctions non linéaires  $f_1$  et  $f_2$ ; et donc le superviseur génère de fortes commandes compensatrices pour atténuer les effets indésirables des perturbations.

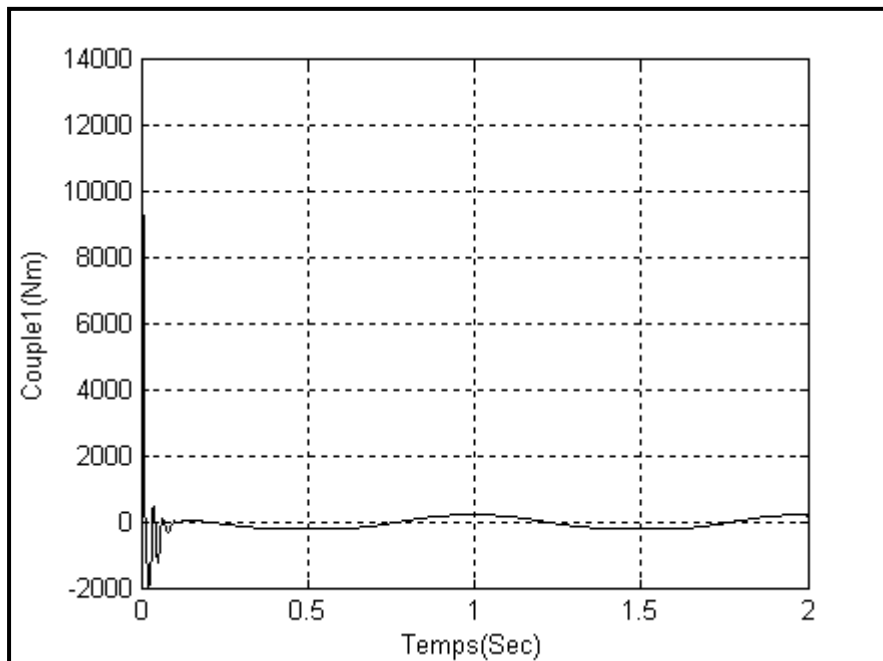
La présence des paramètres dans les fonctions non linéaires  $f_1$  et  $f_2$  minimise l'effet de la compensation pour certains changements, voire même la non fonctionnalité du superviseur, ce qui est le cas pour un changement de la masse  $m_1 = 10$ , les figures (III.41) et (III.42) illustrent un tel inconvénient



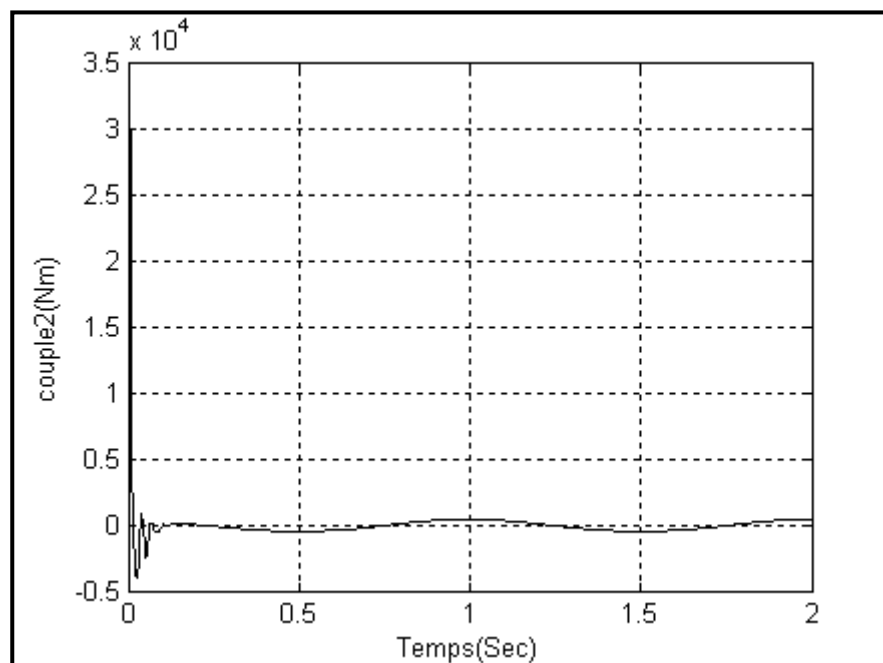
**Figure III.37 : La position angulaire  $q_1$  pour  $l_2 = 0.5$**



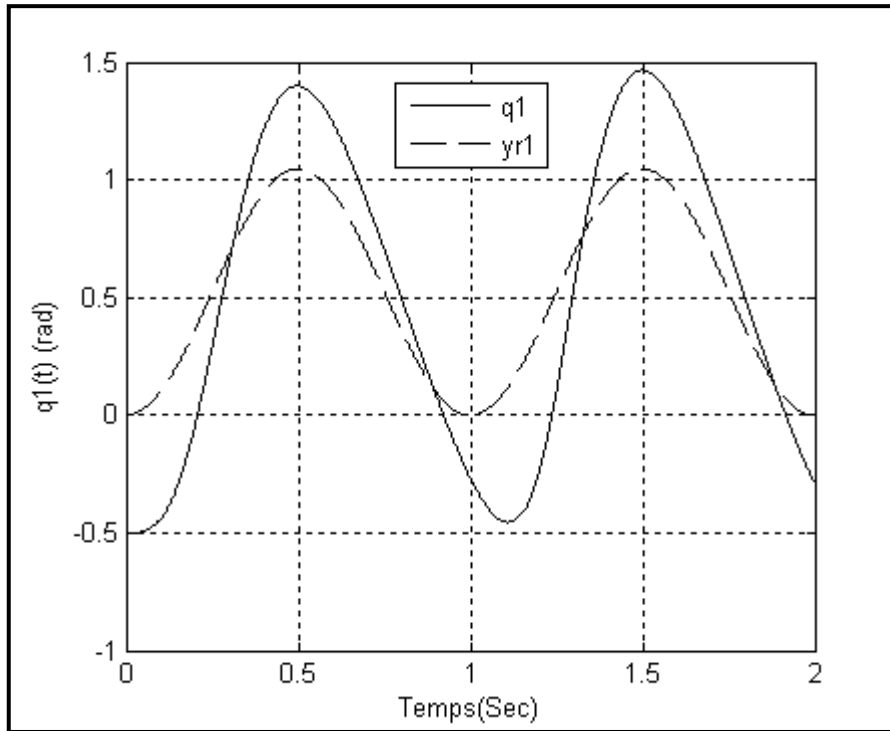
**Figure III.38 : La position angulaire  $q_2$  pour  $l_2 = 0.5$**



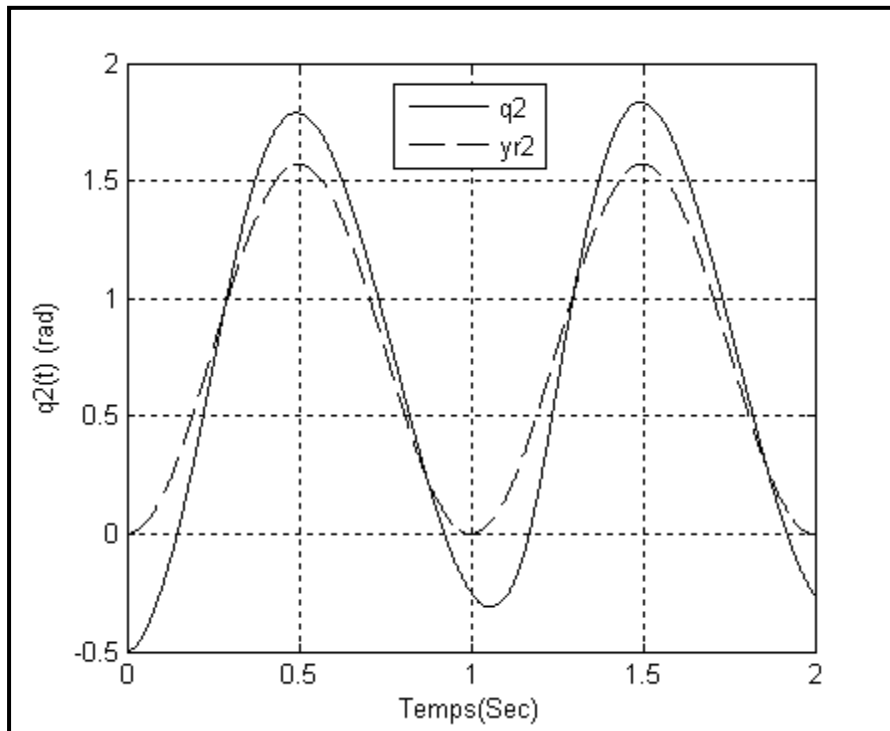
**Figure III.39 : Le couple  $\tau_1$  pour  $l_2 = 0.5$**



**Figure III.40 : Le couple  $\tau_2$  pour  $l_2 = 0.5$**



**Figure III.41 : La position angulaire  $q_1$  pour  $m_1 = 10$**



**Figure III.42 : La position angulaire  $q_2$  pour  $m_1 = 10$**

## **IV. Sommaire**

Dans ce chapitre on a présenté une approche linguistique permettant le découplage dans un système MIMO non linéaire. Après avoir décrit le fondement mathématique de cette approche, des applications ont été présentées pour étudier son efficacité.

Pour la station d'épuration des eaux usées, le découplage a été parfaitement réalisé avec une bonne rejection de l'effet des interactions, cependant la variation d'un paramètre du système a entraîné un dysfonctionnement. Pour le robot, la variation de la longueur n'a pas influencé sur le bon fonctionnement du compensateur, car ce paramètre intervient directement sur l'interaction, mais cette action de rejet a nécessité de forts couples pour la réjection. Tandis que la variation de la masse détériore les performances du système.



**John Henry Holland**  
2 février 1929-

*Computer programs that "evolve" in ways that resemble natural selection can solve complex problems even their creators do not fully understand*

*by John H. Holland*



**David Edward Goldberg**  
26 septembre 1953-

## CHAPITRE IV

# L'APPROCHE DE DECENTRALISATION PAR BACKSTEPPING

### I. Introduction

La méthode proposée dans ce chapitre consiste à considérer les interactions comme des perturbations influençant sur la dynamique et le comportement désiré de chaque sous – système. L'objectif sera donc de minimiser leurs effets. L'algorithme est basé sur l'utilisation de la commande stabilisante backstepping en présence d'incertitude décrite dans le chapitre précédent, elle opère selon l'algorithme suivant :

1. On définit l'ensemble des sous-systèmes constituant le système global ; chaque sous – système  $\sum_{si}$  est défini par le triplet  $(x_i, u_i, y_i)$ ,  $x_i$  représente son vecteur d'état,  $u_i$  représente son signal de commande et  $y_i$  sa sortie appropriée;



2. Pour le sous – système  $\sum_{si}$  les triplets  $(x_j, u_j, y_j), j \neq i$ , définissent des fonctions d’interaction non linéaires;
3. Pour le sous– système  $\sum_{si}$  les interactions sont rangées dans un vecteur  $\Delta_i$ , ce vecteur est considéré comme vecteur de perturbation;
4. Les paramètres d’un sous– système  $\sum_{si}$  sont placés dans le vecteur  $\Delta_i$ ; ceci évite l’influence de toute variation des paramètres de la structure sur les comportements désirés;
5. Pour chaque sous- système, un contrôleur backstepping monovarié est considéré selon le lemme (2.5).

Cette approche consiste donc en une décentralisation du système global, le fondement mathématique est décrit dans le prochain paragraphe.

## II. Fondement mathématique de l’approche

Soit le système non linéaire multivariable décrit par l’équation (3.1) avec la supposition (3.1). On considère les hypothèses suivantes :

### **Supposition 4.1**

1. On considère que le vecteur  $f_{v_i}$  défini par (3.3) est décomposable en deux composantes telles que :

$$f_{v_i}(x) = \psi_{v_i1}(x) + \psi_{v_i2}(x), \quad 1 \leq i \leq m \quad (4.1)$$

$\psi_{v_i1}$  dépend seulement de  $x_i$ .

2. On suppose que pour chaque sous- système  $\sum_{si}$  l’élément  $g_{v_i}$  de la matrice  $g_{v_i}$ , décrite par (3.4), est décomposable en deux composantes telles que :

$$g_{v_i}(x) = \varphi_{v_i1}(x) + \varphi_{v_i2}(x), \quad 1 \leq i \leq m \quad (4.2)$$

$\varphi_{v_i1}$  dépend seulement de  $x_i$ .

Sous ces deux suppositions, la dynamique (3.5) se reformule comme suit:

$$\sum_{si} \cdot \begin{cases} \dot{x} = \psi_{v_i}(x) + \varphi_{v_i}(x)u_i + (\tilde{\psi}_{v_i}(x) + \tilde{\varphi}_{v_i}(x) + \Delta_{ci}(x, u)) \\ y_i = h_i(x_{v_i}) \end{cases} \quad (4.3)$$

$$1 \leq i \leq m$$

Avec :

$$\begin{cases} \psi_{v_i}(x) = [\psi_{11} & \psi_{21} & \dots & \psi_{v_i,1}] \\ \varphi_{v_i}(x) = [0 & 0 & \dots & \varphi_{v_i,1}] \\ \tilde{\psi}_{v_i}(x) = [\psi_{12} & \psi_{22} & \dots & \psi_{v_i,2}] \\ \tilde{\varphi}_{v_i}(x, u) = [0 & \dots & 0 & \varphi_{v_i,2} u_i] \end{cases} \quad (4.4)$$

$$1 \leq i \leq m$$

On pose:

$$\tilde{\psi}_{v_i}(x) + \tilde{\varphi}_{v_i}(x, u) + \Delta_{ci}(x, u) = F_i(x)\Delta_i(x, u), \quad 1 \leq i \leq m \quad (4.5)$$

Où  $F_i(x)$  dépend uniquement de  $x_i$  et  $\Delta_i \in \mathfrak{R}^{v_i}$  contient toutes les interactions et les paramètres de la structure. En substituant (4.6) dans (4.4) on obtient un ensemble de sous-systèmes monovariabiles dont les dynamiques sont décrites par:

$$\sum_{si} \cdot \begin{cases} \dot{x}_i = \psi_{v_i}(x) + \varphi_{v_i}(x) u_i + F_i(x)\Delta_i(x, u) \\ y_i = h_i(x) \end{cases}, \quad 1 \leq i \leq m \quad (4.6)$$

Sous cette représentation, l'ensemble présente des perturbations ce qui conduit vers l'application du lemme (2.5). Les paramètres du contrôleur sont déterminés par les algorithmes génétiques.

### III. Applications

#### III.1. Double pendules inversés.

##### III.1.1. Dynamique du système

La figure (IV.1), représente un double pendule inversé connecté par un ressort de rappel, deux couples  $u_1$  et  $u_2$  sont appliqués aux entrées à travers des servo-moteurs afin de garantir les positions perpendiculaires.

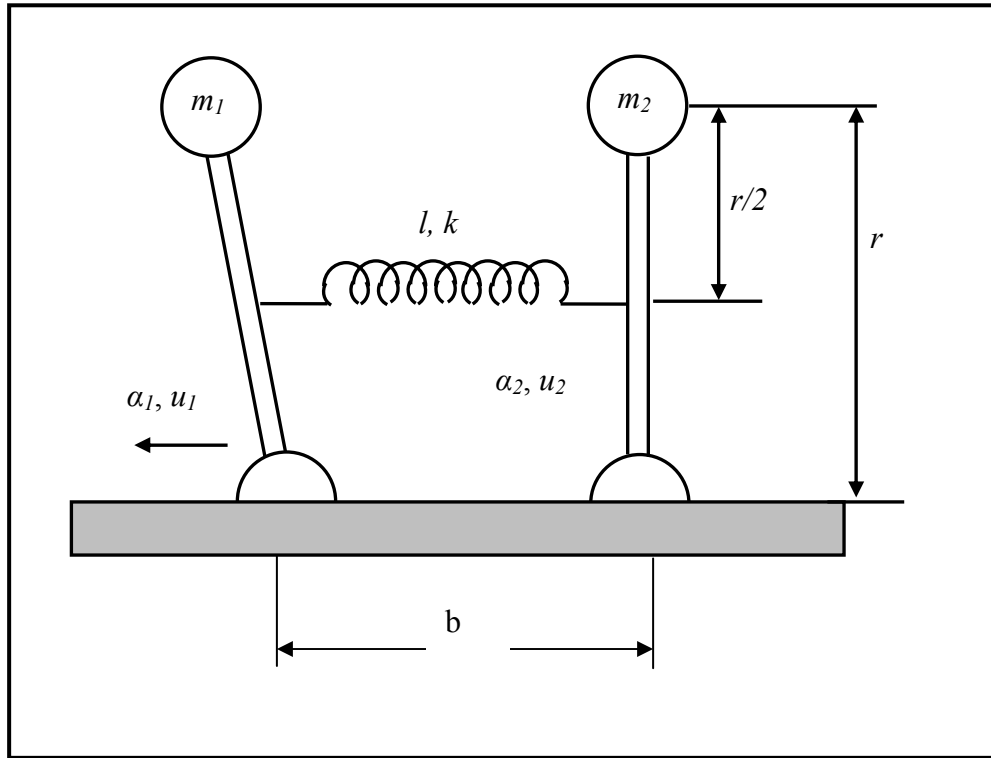


Figure IV.1 Double pendule inversé

La dynamique est donnée [12],[63]

$$\begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = \theta_1 \sin(x_{11}) + \theta_2 \sin(x_{21}) + \theta_3 + \frac{u_1}{j_1} \\ \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = \theta_4 \sin(x_{21}) + \theta_5 \sin(x_{11}) + \theta_6 + \frac{u_2}{j_2} \end{cases} \quad (4.7)$$

avec :

$$\begin{cases} \theta_1 = \frac{m_1 g r}{J_1} - \frac{k r^2}{4 J_1}, \theta_2 = \frac{k r^2}{4 J_1}, \theta_3 = \frac{k r}{2 J_1} (l - b) \\ \theta_4 = \frac{m_2 g r}{J_2} - \frac{k r^2}{4 J_2}, \theta_5 = \frac{k r^2}{4 J_2}, \theta_6 = -\frac{k r}{2 J_2} (l - b) \end{cases}$$

Les variables d'état sont les positions angulaires et les vitesses angulaires :

$$[x_{11} \ x_{12}]^T = [\alpha_1 \ \dot{\alpha}_1]^T \text{ et } [x_{21} \ x_{22}]^T = [\alpha_2 \ \dot{\alpha}_2]^T .$$

$$m_1 = 2\text{kg}, m_2 = 2.5\text{kg}, k = 100\text{N/m}, l = 0.5\text{m}, r = 0.5\text{m}, b = 0.4\text{m}, j_1 = 0.5 \text{ et } j_2 = 0.625 .$$

L'objectif est de définir deux couples pour maintenir l'état d'équilibre défini par  $\alpha_1 = 0$  et  $\alpha_2 = 0$ . Les conditions initiales sont  $\alpha_1 = 0.5$   $\alpha_2 = -0.5$

La figure (IV.2) montre le comportement du système en boucle ouverte. On remarque que le système est instable.

### III.1.2. L'approche directe : Contrôleur Backstepping multivariable.

Les lois de commandes seront calculées en deux étapes. En suivant le même raisonnement que dans les exemples précédent on peut définir les deux lois de commandes suivantes

$$\begin{cases} u_1 = -c_2 z_{12} - z_{11} - \theta_1 \sin(z_{11}) - \theta_2 \sin(z_{21}) - \theta_3 + \dot{\alpha}_{11} \\ u_2 = -c_4 z_{22} - z_{21} - \theta_4 \sin(z_{21}) - \theta_5 \sin(z_{11}) - \theta_6 + \dot{\alpha}_{21} \end{cases} \quad (4.8)$$

Pour ce problèmes l'intervalle de recherches sont définis par  $c_i \in [1 \ 10]$  et un vecteur  $\Gamma = [1 \ 1 \ 1]$ .

#### III.1.2.1. Résultats de simulation

Les paramètres obtenus par les AG sont  $C = [10 \ 10 \ 10 \ 10]$

Les positions angulaires sont données par la figure (IV.3) et des couples par (IV.4). La figure (IV.5) montre les vitesses angulaires.

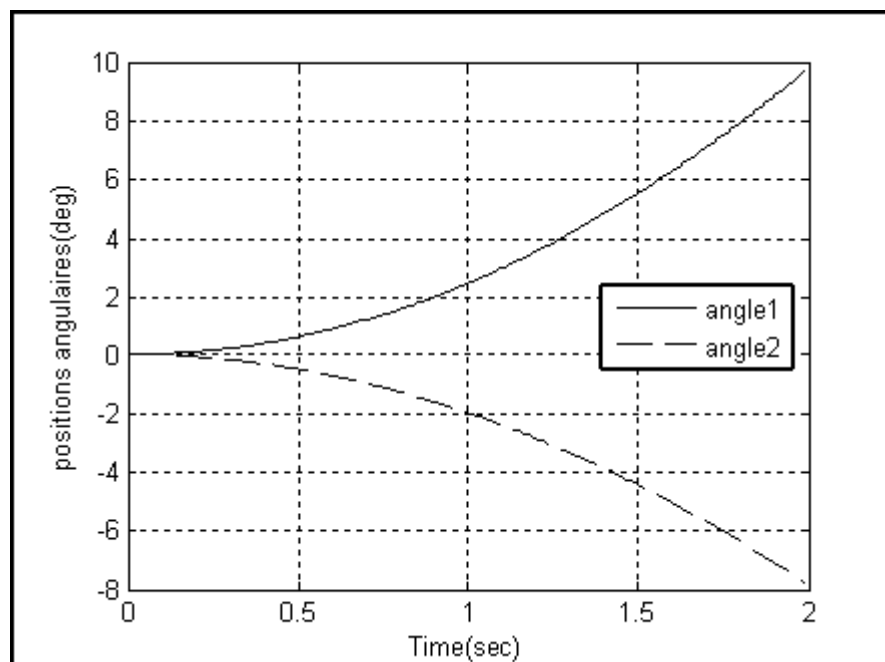
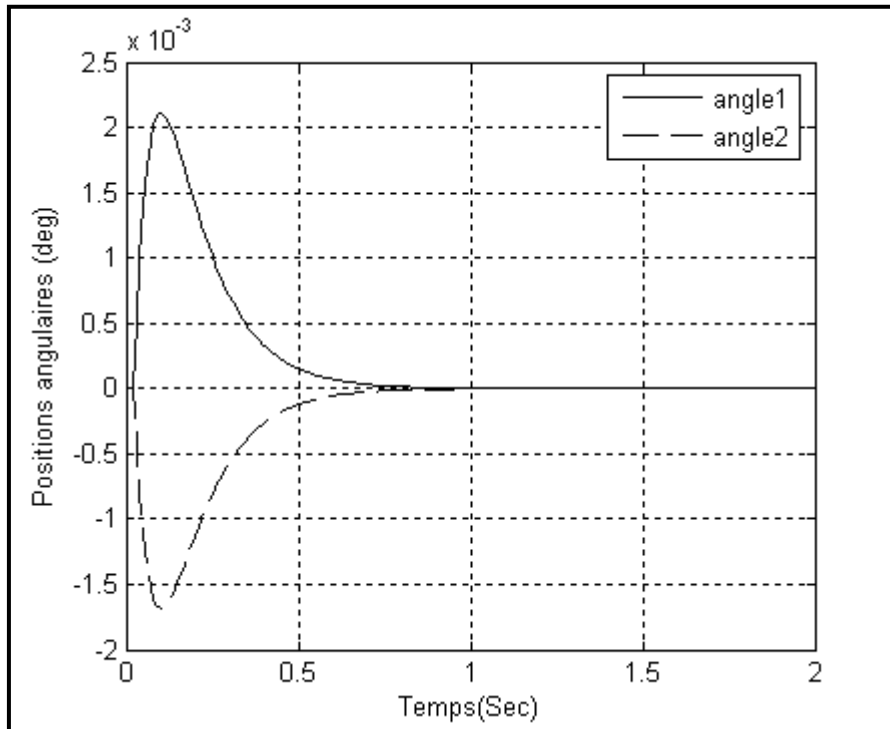
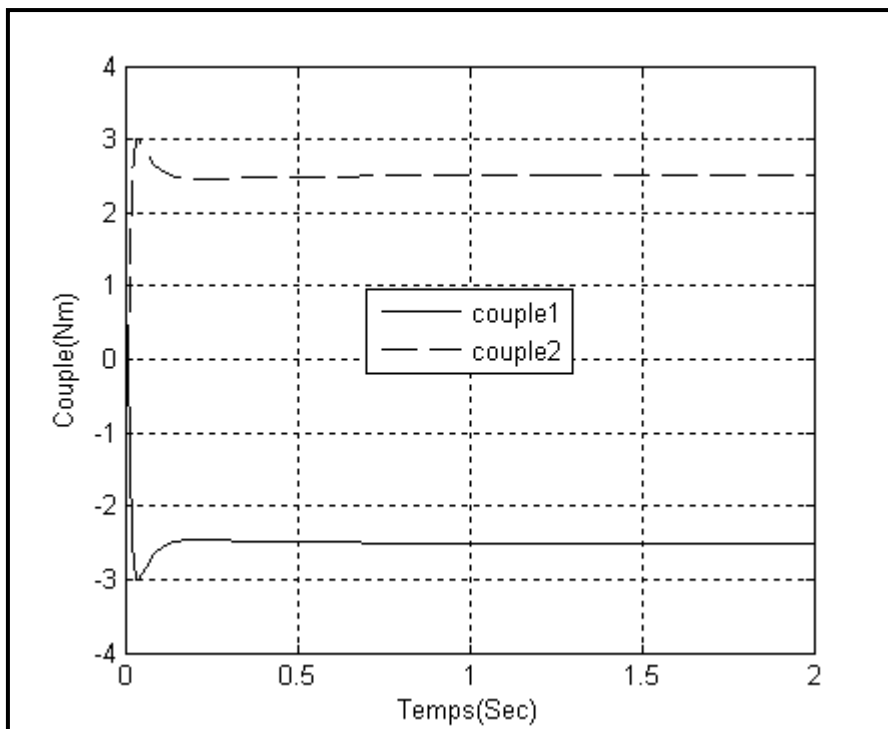


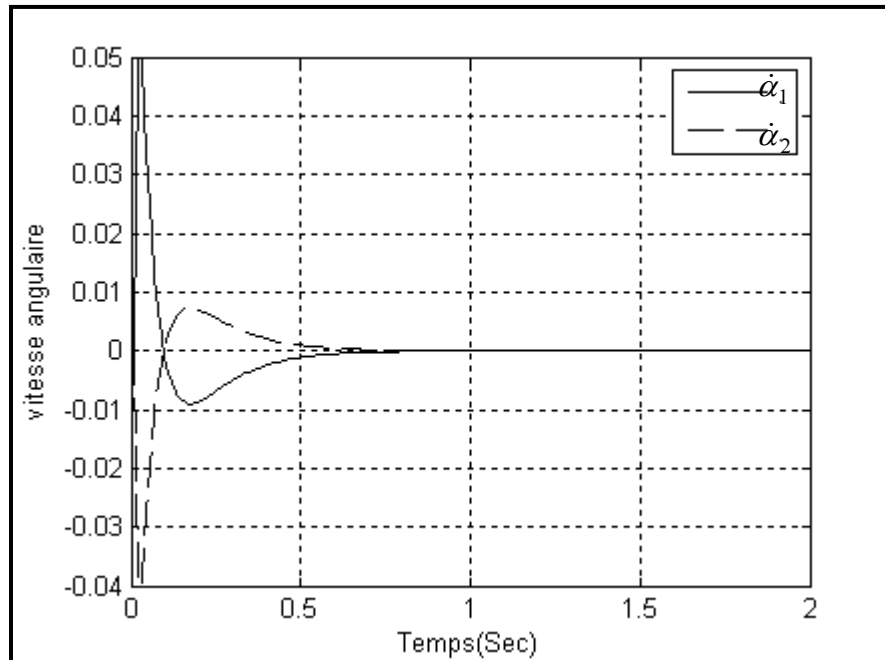
Figure IV.2 : Positions angulaires en fonctionnement



**Figure IV.3 : Positions angulaires**



**Figure IV.4 : Couples appliqués**

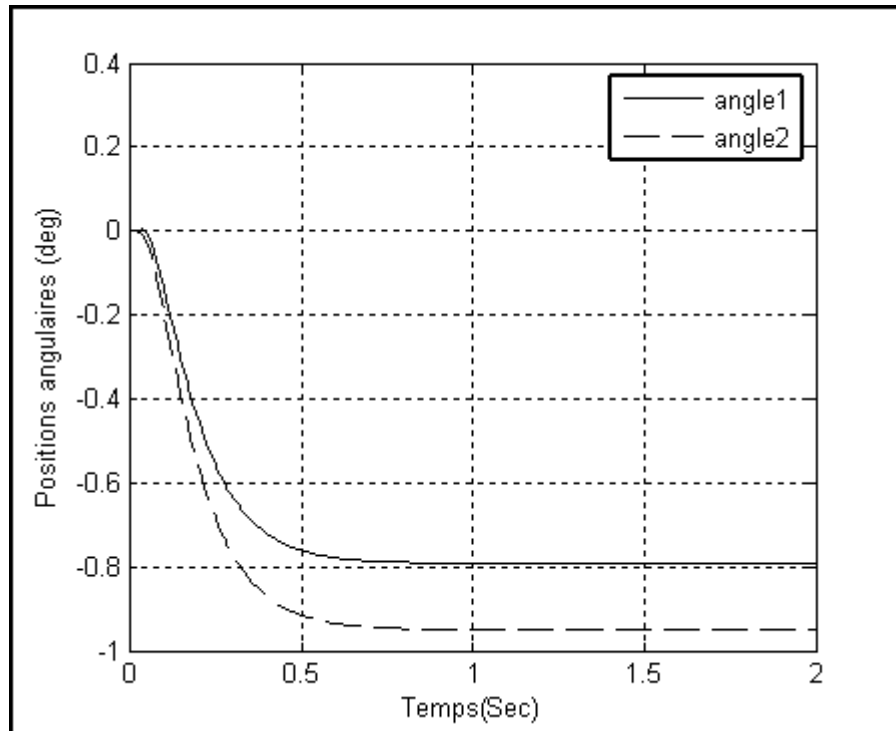


**Figure IV.5 : Vitesses angulaires**

### III.1.2.2. Test de robustesse

Pour tester la robustesse on introduit le paramètre  $\delta = kr(l - b)$  et on procède par une augmentation de ce paramètre.

La figure (IV.6) montre l'évolution des positions angulaires pour  $10\delta$ , on remarque alors qu'il y a une sensibilité vis-à-vis cette variation.



**Figure IV.6 : Positions angulaires pour  $\delta = 10kr(l-b)$**

### III.1.3. Commande par décentralisation.

Dans ce qui suit le système est réécrit selon la forme (3.1) telle que :

$$f(x) = \begin{bmatrix} x_{1,2} \\ \theta_1 \sin(x_{11}) + \theta_2 \sin(x_{21}) + \theta_3 \\ x_{2,2} \\ \theta_4 \sin(x_{21}) + \theta_5 \sin(x_{11}) + \theta_6 \end{bmatrix}, g(x) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$h(x) = \begin{bmatrix} x_{11} \\ x_{21} \end{bmatrix}$$

A travers cette écriture on peut déduire que :

$$\left\{ \begin{array}{l}
 f_{v_1} = \begin{bmatrix} x_{12} \\ \theta_1 \sin(x_{11}) + \theta_2 \sin(x_{21}) + \theta_3 \end{bmatrix} \\
 = \begin{bmatrix} x_{12} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \theta_1 \sin(x_{11}) + \theta_2 \sin(x_{21}) + \theta_3 \end{bmatrix} \\
 = \psi_{v_11}(x) + \psi_{v_12}(x) \\
 f_{v_2} = \begin{bmatrix} x_{22} \\ \theta_4 \sin(x_{21}) + \theta_5 \sin(x_{11}) + \theta_6 \end{bmatrix} \\
 = \begin{bmatrix} x_{22} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \theta_4 \sin(x_{21}) + \theta_5 \sin(x_{11}) + \theta_6 \end{bmatrix} \\
 = \psi_{v_21}(x) + \psi_{v_22}(x) \\
 g_{v_1} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \\
 g_{v_2} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}
 \end{array} \right.$$

Selon (4.5) on a :

$$\left\{ \begin{array}{l}
 F_1(x) = [\sin(x_{11}) \quad 1 \quad 1 \quad 1] \\
 \Delta_1(x, u, t) = \begin{bmatrix} \theta_1 & \theta_2 \sin(x_{21}) & \theta_3 & (\frac{1}{j_1} - 1)u_1 \end{bmatrix}
 \end{array} \right.$$

Et

$$\left\{ \begin{array}{l}
 F_2(x) = [\sin(x_{21}) \quad 1 \quad 1 \quad 1] \\
 \Delta_2(x, u, t) = \begin{bmatrix} \theta_4 & \theta_5 \sin(x_{11}) & \theta_6 & (\frac{1}{j_2} - 1)u_2 \end{bmatrix}
 \end{array} \right.$$

Ce qui permet de passer à la forme décrite par (4.6) avec m=2, alors :

$$\begin{array}{l}
 \sum_{s1} \cdot \begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = u_1 + F_1(x)\Delta_1(x, u, t) \end{cases} \\
 \sum_{s2} \cdot \begin{cases} \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = u_2 + F_2(x)\Delta_2(x, u, t) \end{cases}
 \end{array} \tag{4.9}$$

Les lois de commandes seront calculées en absence des perturbations en deux étapes, puis et par l'application du lemme (2.5) on s'intéresse à l'atténuation des effets perturbateurs.

On considère les sous-systèmes sans perturbations



$$\sum_{s1} \begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = u_1 \end{cases} \quad (4.10)$$

$$\sum_{s2} \begin{cases} \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = u_2 \end{cases}$$

On suit le même raisonnement que dans les paragraphes précédents on montre que les lois de commandes stabilisant le système (4.10) sont :

$$\begin{cases} u_1 = -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} \\ u_2 = -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} \end{cases} \quad (4.11)$$

Avec

$$\begin{cases} z_{11} = x_{11} \\ z_{21} = x_{21} \\ z_{12} = x_{12} - \alpha_{11} \\ z_{22} = x_{22} - \alpha_{21} \\ \alpha_{11} = -c_1 x_{11} \\ \alpha_{21} = -c_2 x_{21} \end{cases}$$

Selon le lemme (2.5) les lois de commande pour le système globale (4.9) sont données par :

$$\begin{cases} u_1 = -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} - k_1 z_{12} (\sin^2(x_{11}) + 3) \\ u_2 = -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} - k_2 z_{22} (\sin^2(x_{21}) + 3) \end{cases} \quad (4.12)$$

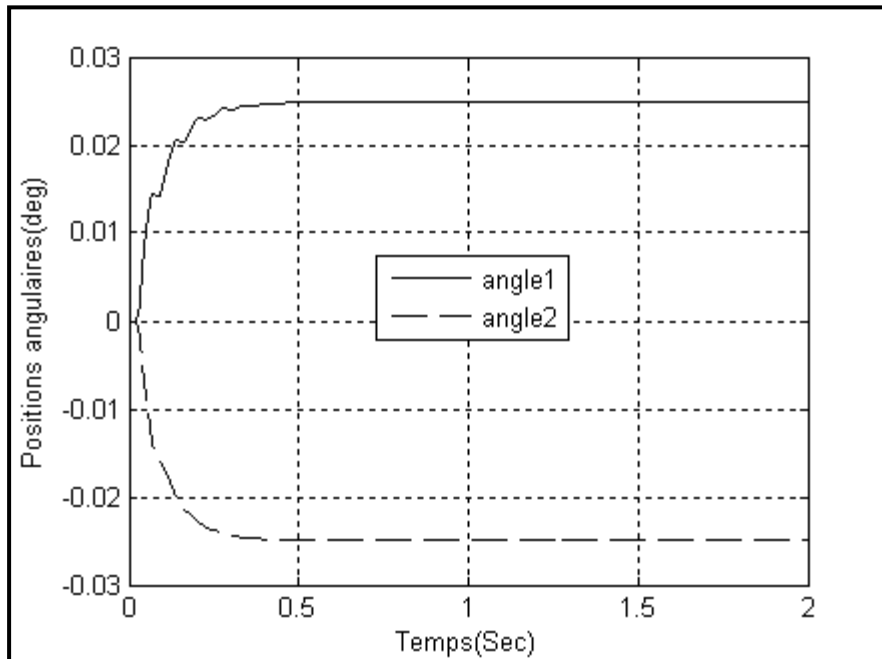
il s'agit des deux objectifs :

- Trouver les paramètres  $c_i$  assurant une bonne convergence
- Trouver les paramètres  $k_i$  réalisant l'atténuation des effets perturbateurs.

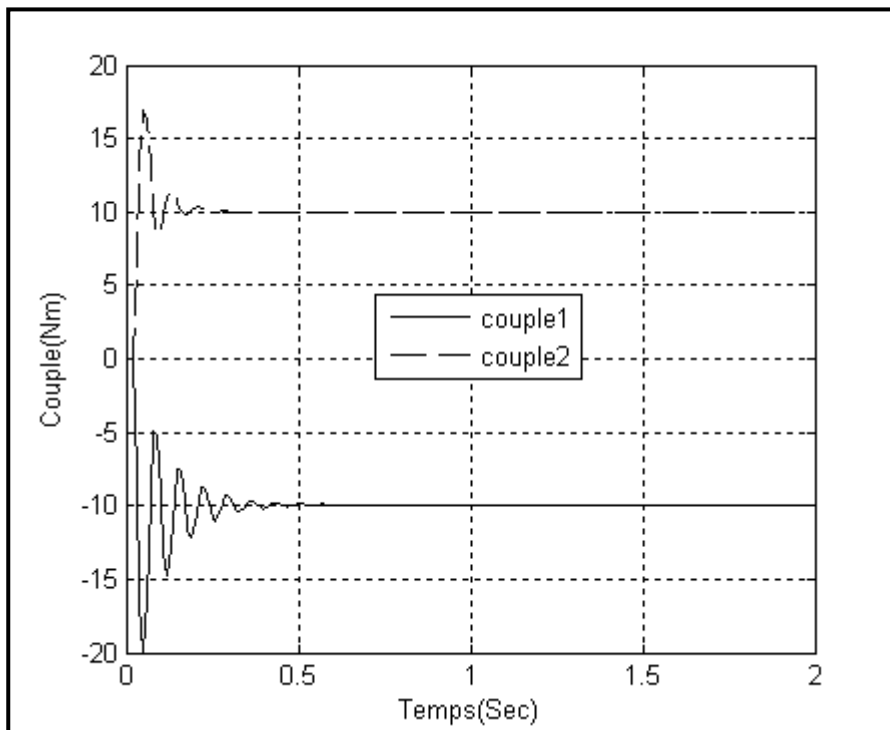
L'optimisation est par les MOGA concerne les paramètres de conception  $c_i$  et  $k_i$ . Les intervalles de recherche sont  $c_i \in [1 \ 10]$  et  $k_i \in [1 \ 10]$ .

### III.1.3.1. Résultats de simulation

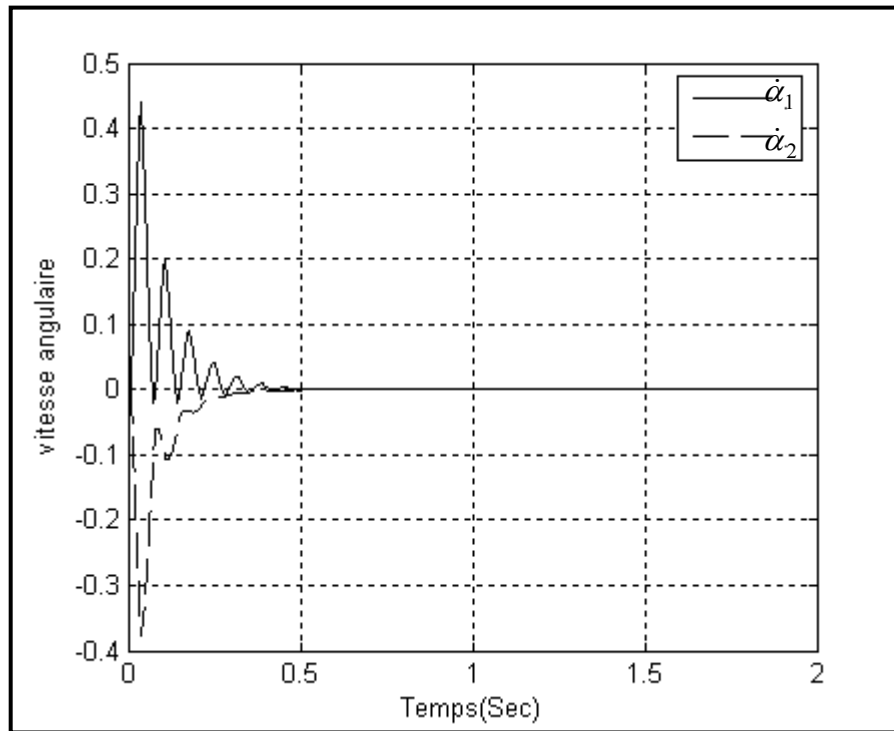
Les meilleurs paramètres obtenus par les MOGA sont  $c = [10 \ 9.99 \ 9.86 \ 9.7]$  et  $k = [10 \ 10]$  les positions angulaires sont données par la figure (IV.7). On constate que la décentralisation a été effectuée à travers les couples illustrés par la figure (IV.8). La figure (IV.9) montre les vitesses angulaires.



**Figure IV.7 : Positions angulaires**



**Figure IV.8: Couples appliqués**



**Figure IV.9 : Vitesses angulaires**

### III.1.3.2. Test de robustesse

La robustesse est testée de la même façon que précédemment, il s'agit d'augmenter le paramètre  $\delta = kr(l - b)$  et de voir son effet sur les positions angulaires.

La figure (IV.10) montre que les contrôleurs ont pu atténuer l'effet de cette augmentation ce qui est interprété par une décentralisation du système global.

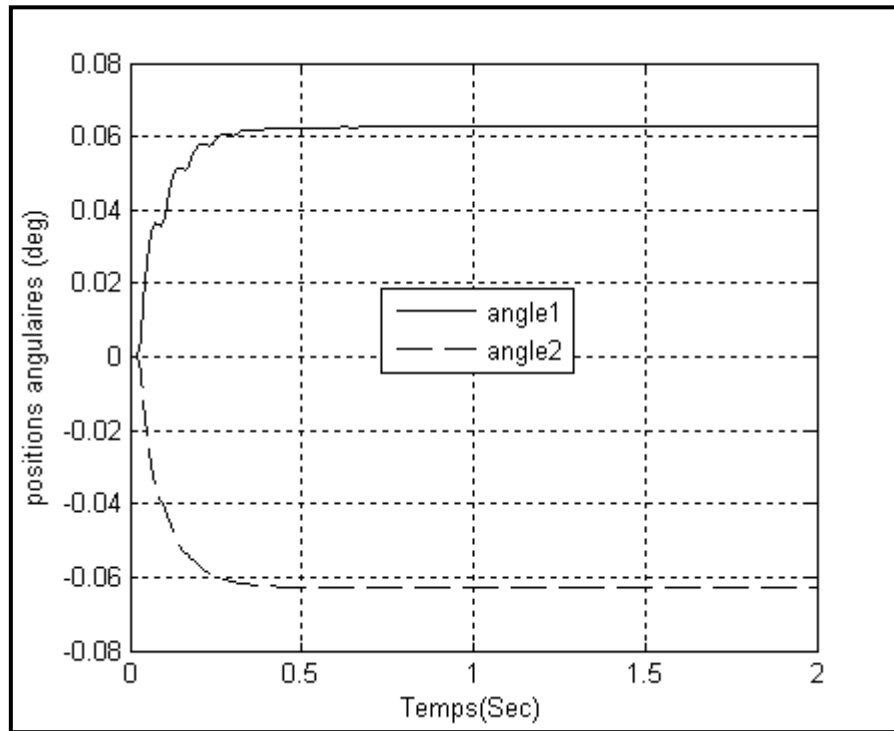


Figure IV.10 : Positions angulaires pour  $\delta = 10kr(l-b)$

## III.2. Bras de robots à deux segments

### III.2.1. Commande par décentralisation

Dans ce paragraphe, on fait une étude comparative entre l'approche de décentralisation et l'approche directe présente dans le paragraphe (§III.2.2) du chapitre précédent.

Selon (3.1) on peut déduire que :

$$f(x) = \begin{bmatrix} x_{12} \\ f_1(x) \\ x_{22} \\ f_2(x) \end{bmatrix}, g(x) = \begin{bmatrix} 0 & 0 \\ h_{11} & h_{12} \\ 0 & 0 \\ h_{21} & h_{22} \end{bmatrix}, u = [\tau_1 \quad \tau_2]^T \text{ et } h(x) = [x_{11} \quad x_{21}]^T$$

Avec

$$\begin{cases} h_{11} = \frac{1}{l_1^2} \frac{1}{m_1 + m_2 \sin^2(d)} \\ h_{12} = h_{21} = \frac{-1}{l_1 l_2} \frac{\cos(d)}{m_1 + m_2 \sin^2(d)} \\ h_{22} = \frac{m}{m_2 l_2^2} \frac{1}{\cos(d) m_1 + m_2 \sin^2(d)} \end{cases}$$

$$\left\{ \begin{array}{l} f_1(x) = -g \frac{m}{l_1} \sin(x_{11}) \frac{1}{m_1 + m_2 \sin^2(d)} + \left[ -2m_2 x_{12}^2 \sin\left(\frac{d}{2}\right) \right. \\ \left. + \frac{m_2 l_2}{l_1} x_{22}^2 \sin(d) + \frac{m_2}{l_1} g \cos(d) \sin(x_{21}) \right] \frac{1}{m_1 + m_2 \sin^2(d)} \\ f_2(x) = -g \frac{m}{l_1} \sin(x_{21}) \frac{1}{m_1 + m_2 \sin^2(d)} + \left[ -2m_2 x_{22}^2 \sin\left(\frac{d}{2}\right) \right. \\ \left. + \frac{m l_1}{l_2} x_{12}^2 \sin(d) + \frac{m}{l_2} g \cos(d) \sin(x_{11}) \right] \frac{1}{m_1 + m_2 \sin^2(d)} \end{array} \right.$$

Alors

$$\left\{ \begin{array}{l} f_{v_1} = \begin{bmatrix} x_{12} \\ f_1 \end{bmatrix} = \begin{bmatrix} x_{12} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ f_1 \end{bmatrix} \\ \quad = \psi_{v_1 1}(x) + \psi_{v_1 2}(x) \\ f_{v_2} = \begin{bmatrix} x_{22} \\ f_2 \end{bmatrix} = \begin{bmatrix} x_{22} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ f_2 \end{bmatrix} \\ \quad = \psi_{v_2 1}(x) + \psi_{v_2 2}(x) \\ g_{v_1} = \begin{bmatrix} 0 & 0 \\ h_{11} & h_{12} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ h_{11} - 1 & h_{12} \end{bmatrix} \\ \quad = \varphi_{v_1 1}(x) + \varphi_{v_1 2}(x) \\ g_{v_2} = \begin{bmatrix} 0 & 0 \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ h_{21} & h_{22} - 1 \end{bmatrix} \\ \quad = \varphi_{v_2 1}(x) + \varphi_{v_2 2}(x) \end{array} \right.$$

Selon (4.5)

$$\left\{ \begin{array}{l} F_1(x) = [\sin(x_{11}) \quad 1 \quad 1 \quad 1] \\ \Delta_1 = \begin{bmatrix} -g \frac{m}{l_1} \frac{1}{m_1 + m_2 \sin^2(d)} \\ (-2m_2 x_{12}^2 \sin\left(\frac{d}{2}\right) + \frac{m_2 l_2}{l_1} x_{22}^2 \sin(d) + \frac{m_2}{l_1} g \cos(d) \sin(x_{21})) \frac{1}{m_1 + m_2 \sin^2(d)} \\ (h_{11} - 1)\tau_1 \\ h_{12}\tau_2 \end{bmatrix} \end{array} \right.$$

Et

$$\left\{ \begin{array}{l} F_2(x) = [\sin(x_{21}) \quad 1 \quad 1 \quad 1] \\ \Delta_2 = \left[ \begin{array}{l} -g \frac{m}{l_1} \frac{1}{m_1 + m_2 \sin^2(d)} \\ (-2m_2 x_{22}^2 \sin(\frac{d}{2}) + \frac{ml_1}{l_2} x_{12}^2 \sin(d) + \frac{m}{l_2} g \cos(d) \sin(x_{11})) \frac{1}{m_1 + m_2 \sin^2(d)} \\ h_{21} \tau_1 \\ (h_{22} - 1) \tau_2 \end{array} \right] \end{array} \right.$$

Avec  $d = x_{11} - x_{21}$ ,  $m = m_1 + m_2$

Ce qui mène, selon (4.5) aux deux sous-systèmes monovariabales

$$\begin{aligned} \sum_{s^1} \dot{x} & \begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = \tau_1 + F_1(x) \Delta_1(x, u, t) \end{cases} \\ \sum_{s^2} \dot{x} & \begin{cases} \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = \tau_2 + F_2(x) \Delta_2(x, u, t) \end{cases} \end{aligned} \quad (4.13)$$

Il est évident que le système (4.13) est analogue au système (4.9) ce qui mène aux mêmes lois de commande (4.12) pour stabiliser ce système. Les commandes décentralisatrices sont selon le lemme (2.5) :

$$\begin{cases} \tau_1 = -c_3 z_{12} - z_{11} + \dot{\alpha}_{11} - k_1 z_{12} (\sin^2(x_{11}) + 3) \\ \tau_2 = -c_4 z_{22} - z_{21} + \dot{\alpha}_{21} - k_2 z_{22} (\sin^2(x_{21}) + 3) \end{cases} \quad (4.14)$$

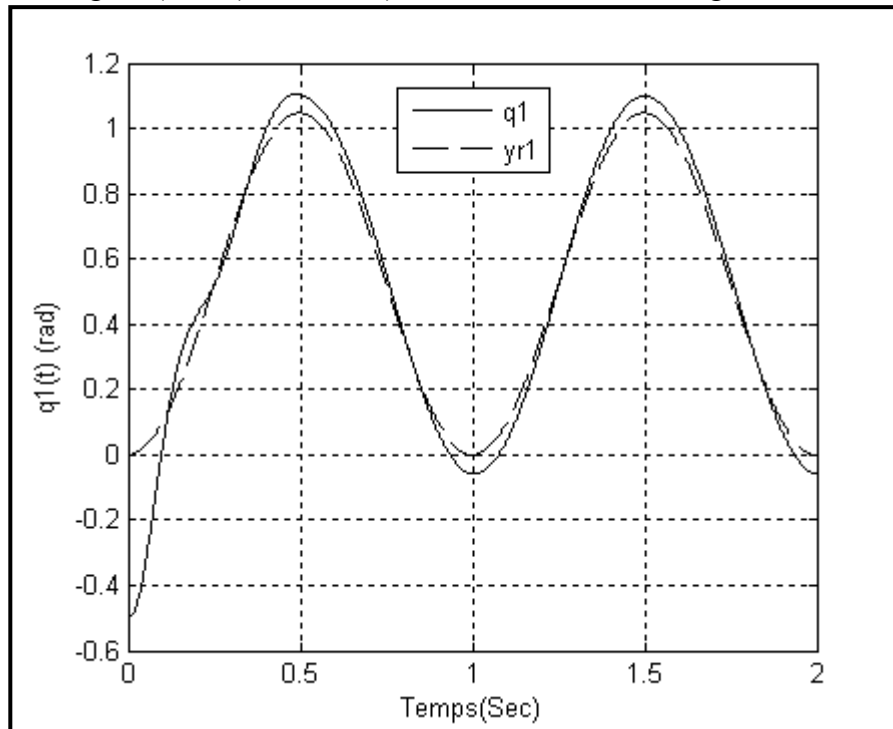
Avec

$$\begin{cases} z_{11} = x_{11} - y_{r1} \\ z_{21} = x_{21} - y_{r2} \\ z_{12} = x_{12} - \alpha_{11} \\ z_{22} = x_{22} - \alpha_{21} \\ \alpha_{11} = -c_1 x_{11} + \dot{y}_{r1} \\ \alpha_{21} = -c_2 x_{21} + \dot{y}_{r2} \end{cases}$$

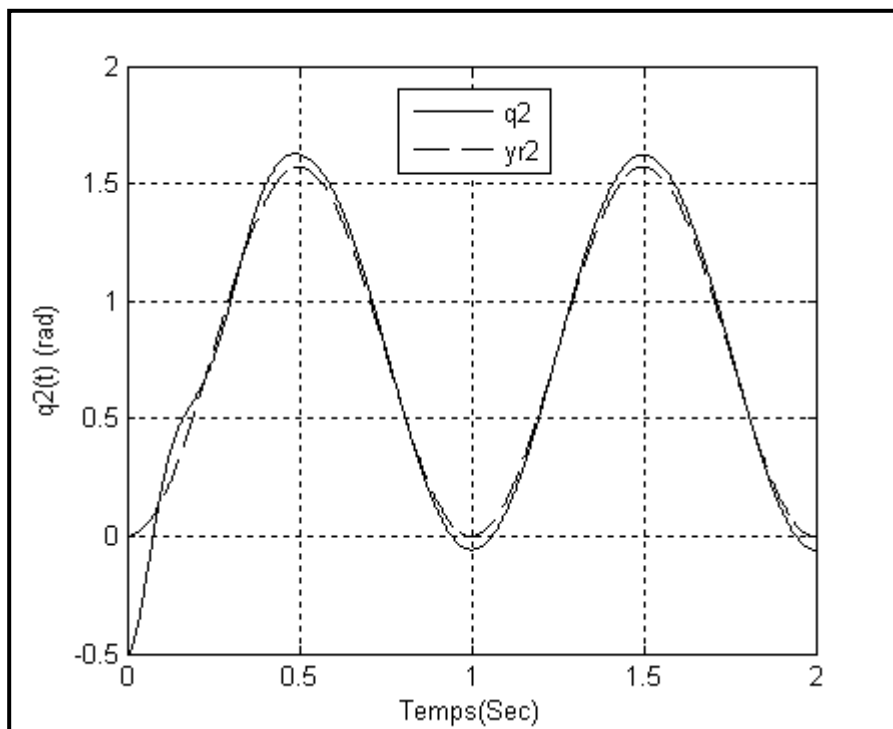
La recherche des vecteurs C et K se fait par les MOGA et avec les intervalles de recherches tels que  $c_i, k_i \in [1 \quad 100]$ .

### III.2.1.1. Résultats de simulation

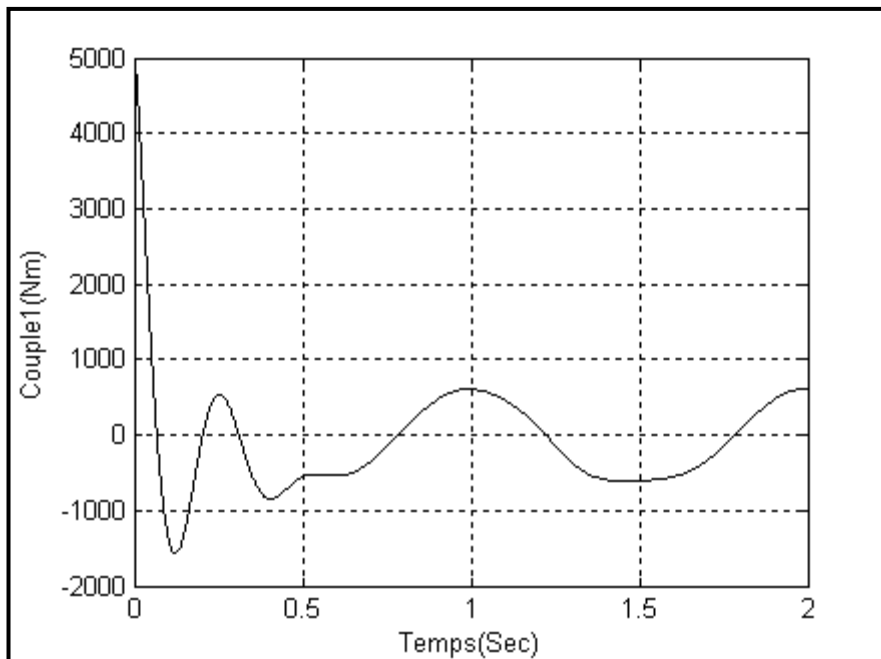
Les paramètres obtenus sont :  $[23.74 \quad 78.13 \quad 97.48 \quad 73.48]$  et  $k = [99.42 \quad 16.29]$ . Les positions angulaires sont données par les figures (IV.11) et (IV.12). On constate que la décentralisation a été effectuée à travers les couples illustrés par les figures (IV.13) et (IV.14). Les figure (IV.15) et 5IV.16) montrent les vitesses angulaires.



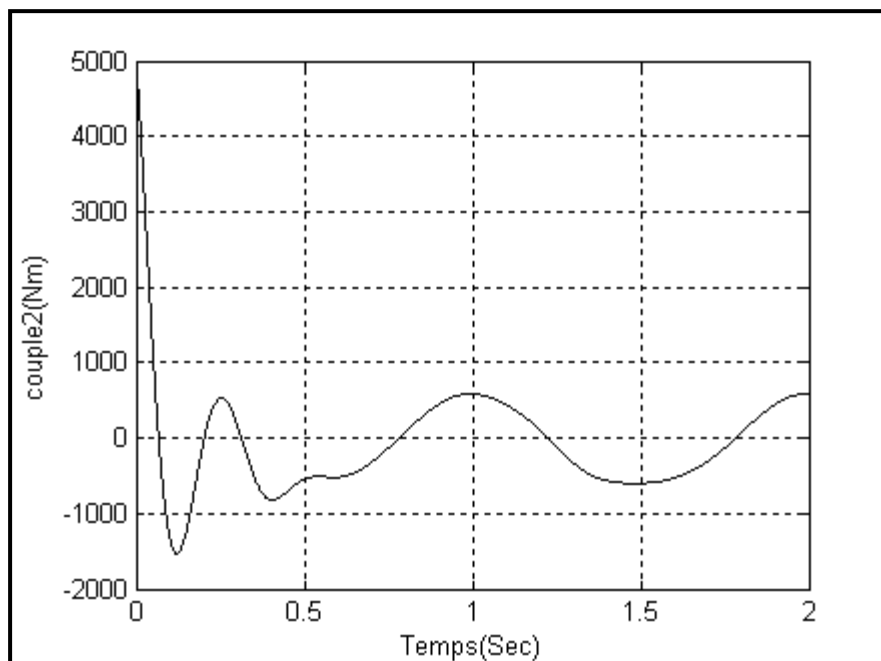
**Figure IV.11 : Position angulaire  $q_1$**



**Figure IV.12 : Position angulaire  $q_2$**

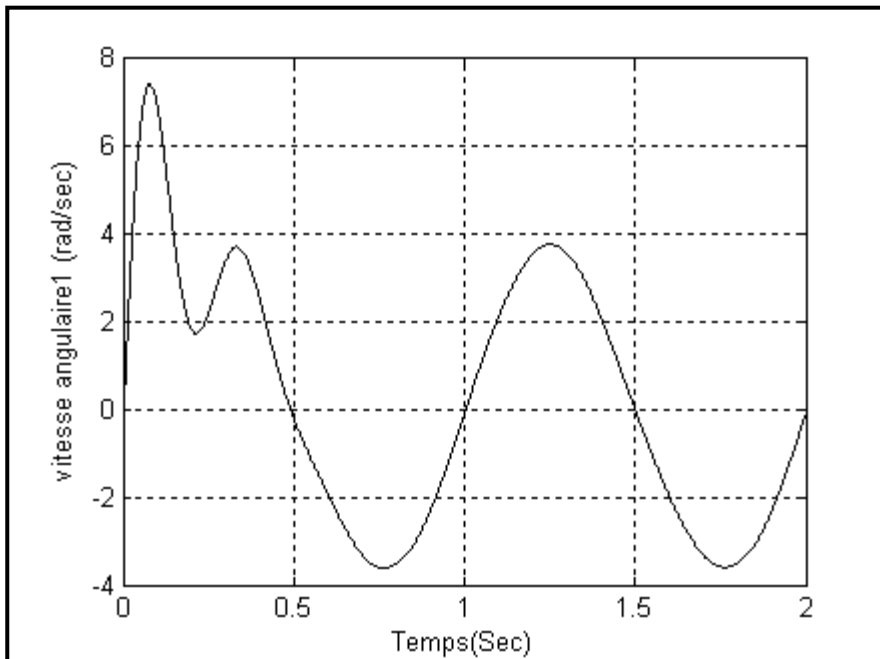


**Figure IV.13 : Couple  $\tau_1$**

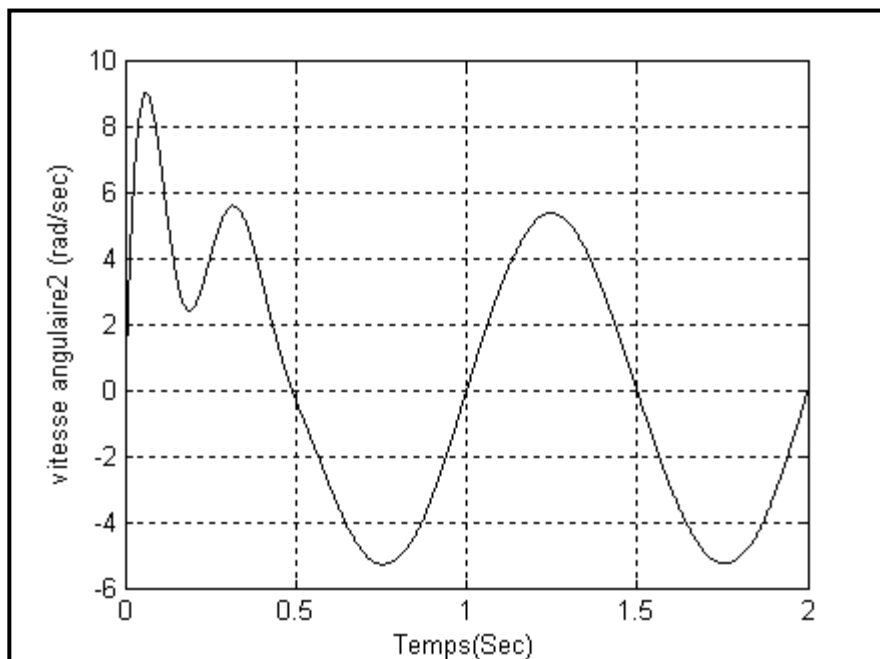


**Figure IV.14 : Couple  $\tau_2$**





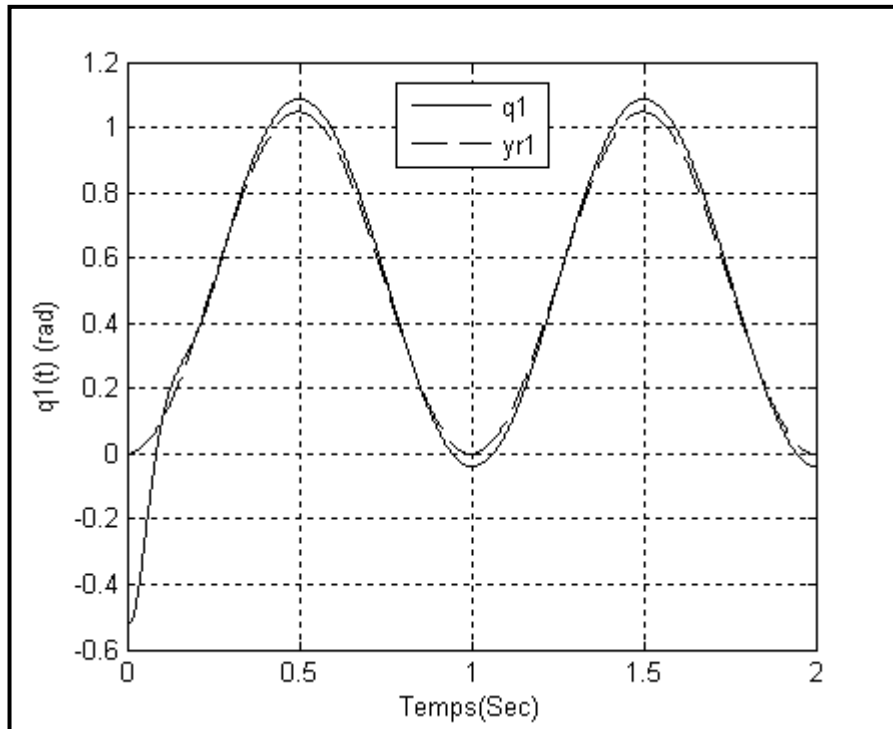
**Figure IV.15 : Vitesse angulaire**



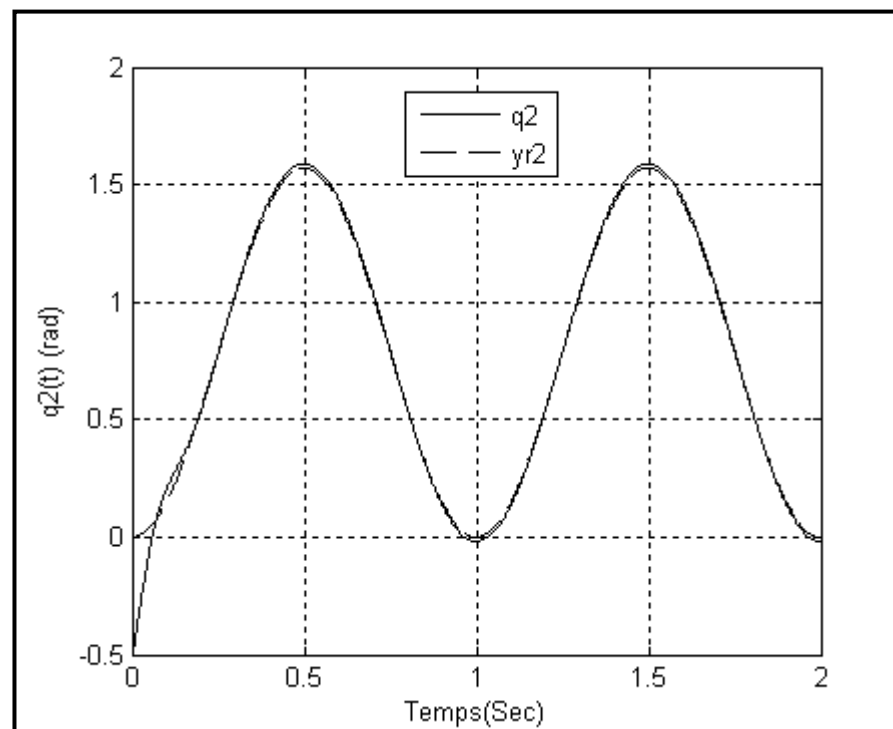
**Figure IV.16 : vitesse angulaire**

### III.2.1.2. Test de robustesse

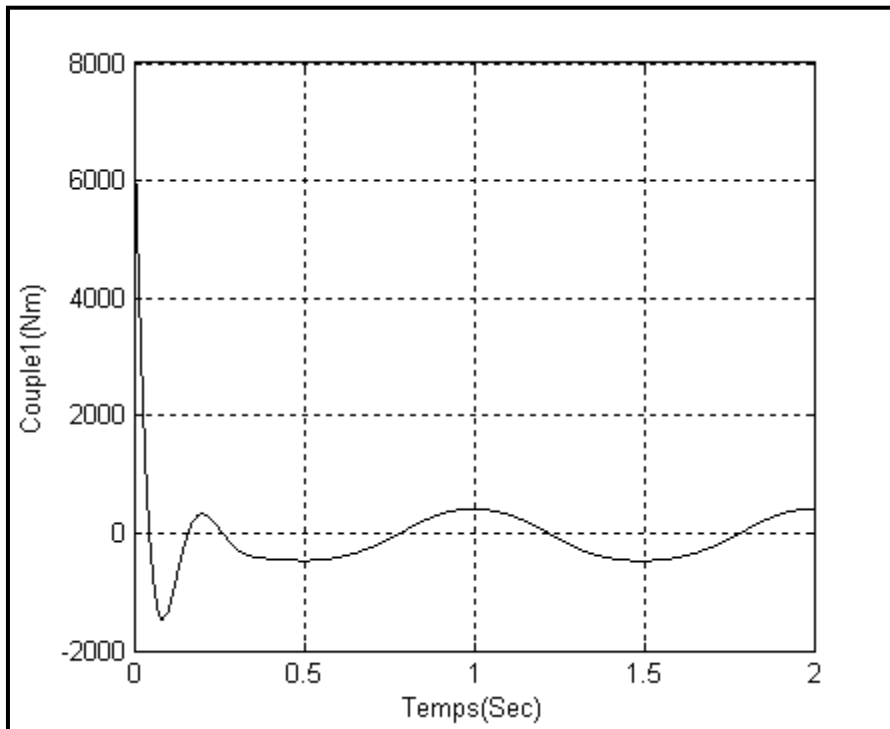
Le test s'effectue de la même manière que précédemment, il s'agit d'augmenter l'effet des interactions en diminuant la longueur du bras, selon les figures (IV.17) et (IV.18), une importante robustesse est obtenue. Dans un second lieu il s'agit de perturber une masse, selon les figures (IV.21) et (IV.22), cette variation de masse n'atteint pas le comportement désiré.



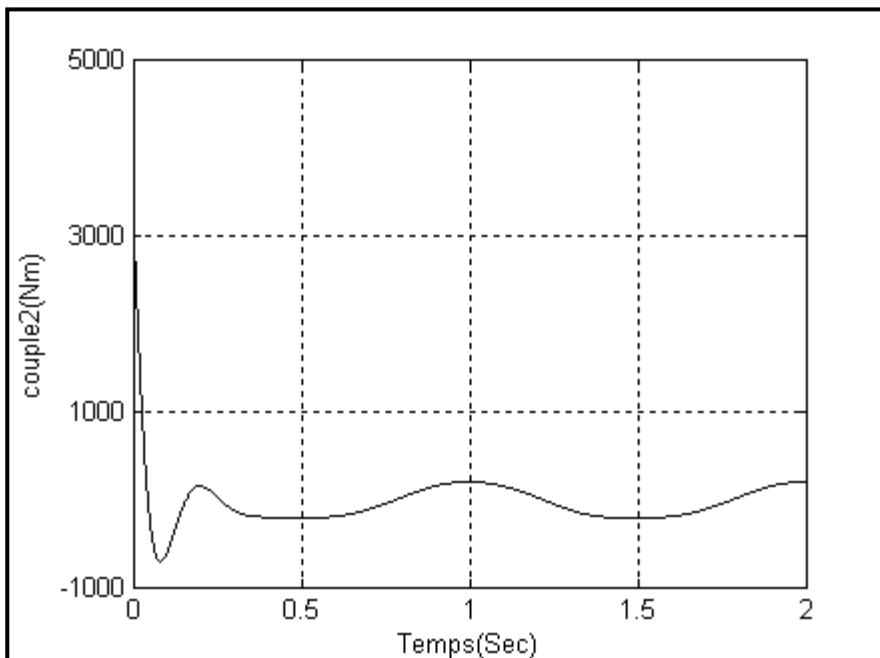
**Figure IV.17 : Position angulaire  $q_1$  pour  $l_2=0.5$**



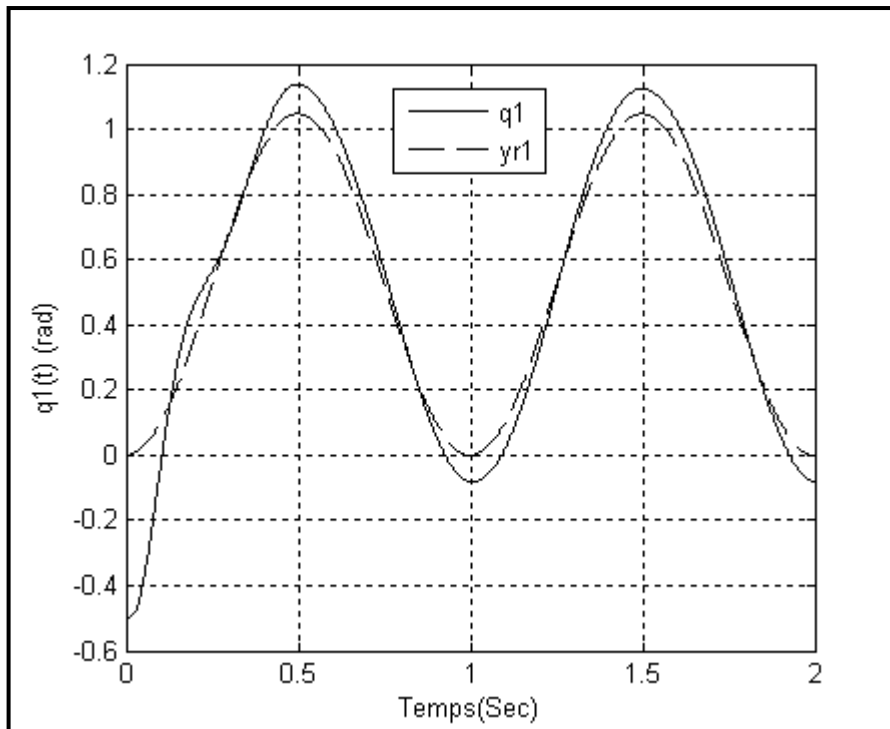
**Figure IV.18 : Position angulaire  $q_2$  pour  $l_2=0.5$**



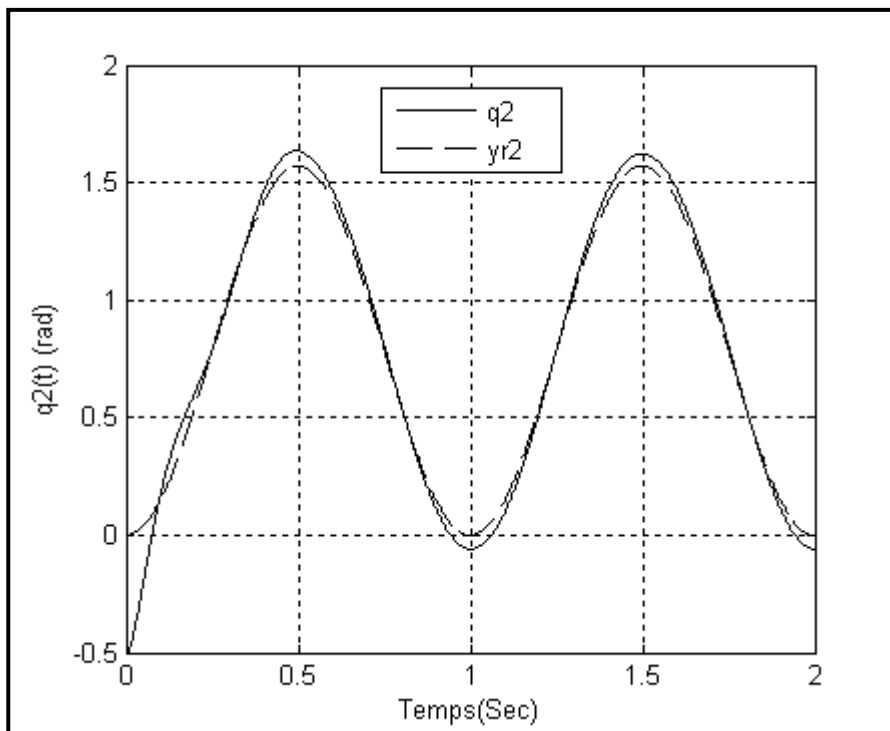
**Figure IV.19 : Couple  $\tau_1$  pour  $l_2=0.5$**



**Figure IV.20 : Couple  $\tau_2$  pour  $l_2=0.5$**



**Figure IV.21 : Position angulaire  $q_1$  pour  $m_1=10$**



**Figure IV.22 : Position angulaire  $q_2$  pour  $m_1=10$**

## **IV. Sommaire**

Dans ce chapitre on a présenté une approche de décentralisation par backstepping. Cette approche vient de corriger l'inconvénient de l'approche précédente dans le sens où les paramètres de la structure à commander influencent sur la comportement du système. Il s'agit alors d'éviter un algorithme d'adaptation et de générer des lois de commande plus simple dans leurs formules comparativement aux lois calculées pour l'approche précédente.

Vu la structure des lois de commande on peut tirer deux principaux avantages, dans un premiers lieu c'est la simplicité des formule et l'absence des paramètres du système dans ces lois. Le second avantage c'est que les lois de commande nécessitent uniquement les mesures provenant séparément de chaque capteur pour piloter le système global.

L'approche a été testée pour deux systèmes; pour le double pendule inversé, la décentralisation a été parfaitement réalisée avec une bonne rejection de l'effet des interactions et la variation des paramètres du système n'entraîne aucune perturbation dans le comportement désiré. Pour le robot, la variation de la longueur n'a pas influencé sur le bon fonctionnement du compensateur et ne nécessite pas de forts couples pour la réjection. la variation de la masse détériore uniquement le comportement du premier segment et ne touche nullement au deuxième segment.

## Conclusion générale

Dans cette thèse deux méthodes de commande des systèmes non linéaires multivariables ont été introduites. Une méthode de commande par découplage et une méthode par décentralisation.

Pour le découplage et afin d'éviter le calcul de la matrice de découplage qui présente le risque de singularité, on a proposé un découplage par raisonnement linguistique basée sur la théorie de la logique floue, le superviseur flou chargé de l'anticipation et la compensation des interactions provenant des lois commandes issus des contrôleurs monovariables. L'inconvénient de cette approche de découplage est qu'elle nécessite une loi d'adaptation en cas de présence de variation dans les paramètres, de plus le découplage risque de ne pas garantir le bon fonctionnement au cas de perturbation. La robustesse de cet algorithme est limitée comme le montrent les différents résultats de simulation.

Pour cela, une commande par décentralisation a été implantée pour corriger ce problème, cette commande traite parallèlement les deux problèmes, il s'agit d'introduire la théorie de backstepping avec incertitude dans la boucle de commande. Les interactions sont considérées comme des perturbations. Les résultats obtenus en simulation sur deux systèmes sont encourageants.

Dans les perspectives à venir, on envisage un algorithme d'adaptation pour corriger le problème de l'influence de la variation des paramètres sur le comportement désiré dans le cas de l'approche linguistique. Pour l'approche de décentralisation backstepping, il s'agit de développer des techniques et des algorithmes permettant la décomposition des non linéarité afin de généraliser la méthode proposée.

# Quelques notions sur la géométrie Différentielle

Dans ce qui suit, un vecteur de fonctions  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$  est appelé un champ de vecteur dans  $\mathfrak{R}^n$ . Un champ de vecteur est dit lisse, si la fonction  $f$  est continûment partiellement dérivable jusqu'à un ordre donné.

### **Définition C.1: Fonction Lipschitz**

Une fonction  $f(t,x)$  est dite Lipschitz si :

$$\|f(t,x) - f(t,y)\| \leq L\|x - y\|$$

Pour tout  $(t,x)$  et  $(t,y)$  situent dans un voisinage  $(t_0,x_0)$  et  $(t_0,y_0)$

### **Définition C.2 : dérivée de Lie**

Soit  $h : \mathfrak{R}^n \rightarrow \mathfrak{R}$  une fonction lisse et soit  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$  un champ de vecteur lisse, la dérivée de Lie de  $h$  respectivement à  $f$  est une fonction scalaire définie par :

$$L_f h = \nabla h \cdot f$$

### **Définition C.3 : crochet de Lie**

Soit  $f$  et  $g$  deux champs de vecteur dans  $\mathfrak{R}^n$ , le crochet de Lie est un champ de vecteur défini par :  $ad_f g = [f, g] = \nabla g \cdot f - \nabla f \cdot g$

### **Définition C.4 : Diffeomorphisme et transformations d'état**

Une fonction  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ , définie dans une région  $\Omega$  est appelée diffeomorphisme si elle est lisse et si son inverse  $f^{-1}$  existe et lisse.

Si  $\Omega$  représente tout l'espace  $\mathfrak{R}$ , la fonction  $f$  est dite difféomorphisme global.

Le lemme suivant, permet de s'assurer si la fonction  $f$  est un difféomorphisme local

**Lemme C.1**

Soit une fonction lisse  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ , définie dans une région  $\Omega$ , si la matrice jacobienne  $\nabla f$  est non singulière au voisinage d'un point  $x = x_0$  de  $\Omega$ , alors  $f$  est un difféomorphisme local dans une sous région de  $\Omega$

**Définition C.5**

Un ensemble de champ de vecteur linéairement indépendant  $[f_1 \ f_2 \ \dots \ f_m]$  dans  $\mathfrak{R}^n$  est dit complètement intégrable, si est seulement s'ils existent  $n - m$  fonctions scalaires  $h_1(x), \dots, h_{n-m}(x)$  satisfaisant l'équation différentielle partielle  $\nabla h_i f_j = 0, 1 \leq i \leq n-m, 1 \leq j \leq n$  et les gradient  $\nabla h_i$  sont linéairement indépendant.

**Définition C.6**

Un ensemble de champ de vecteur linéairement indépendant  $[f_1 \ f_2 \ \dots \ f_m]$  dans  $\mathfrak{R}^n$  est dit involutive si est seulement s'ils existent des fonctions scalaires

$$\alpha_{ijk} : \mathfrak{R}^n \rightarrow \mathfrak{R} \text{ tel que } [f_i, f_j](x) = \sum_{k=1}^m \alpha_{ijk}(x) f_k(x), \forall i, j$$

**Théorème C.1 (Frobenius)**

Soit un ensemble de champ de vecteur linéairement indépendant  $[f_1 \ f_2 \ \dots \ f_m]$ , l'ensemble est complètement intégrable si et seulement s'il est involutive

**Définition C.7 : degré relatif**

On dit que le système :

$$\dot{x} = f(x) + g(x)u$$

a un degré relatif  $r$  dans une région  $\Omega$ , si  $\forall x \in \Omega$  on a :

$$\begin{cases} L_g L_f^i h(x) = 0, & 0 \leq i < r-1 \\ L_g L_f^{r-1} h(x) \neq 0 \end{cases}$$



## Contrôleur à logique floue

Les contrôleurs à logique floue opèrent sur des variables linguistiques au lieu des variables numériques, ils fournissent un algorithme de conversion d'une stratégie de contrôle linguistique basée sur l'expertise humaine en une stratégie de contrôle automatique, il est décrit par un ensemble de règles de contrôle flou du type :

$$R_i : \text{if } x \text{ is } A_i \text{ and } y \text{ is } B_{i1} \text{ then } z \text{ is } C_i$$

La structure classique d'un FLC est montrée par figure (B.1).

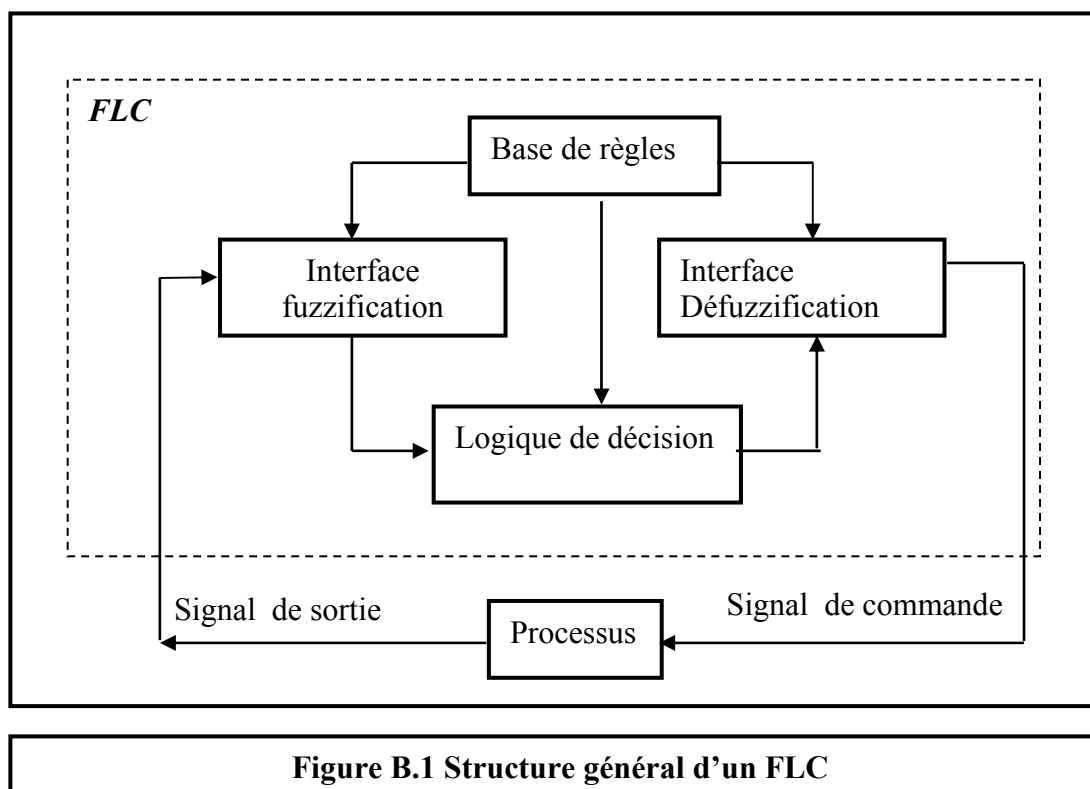


Figure B.1 Structure général d'un FLC

Le FLC fonctionne selon trois étapes :

## **V. Interface fuzzification**

L'interface fuzzification consiste à :

- ✓ Mesurer les valeurs numériques des variables d'entrées.
- ✓ Les projetées dans l'univers de discours en utilisant un facteur d'échelle.
- ✓ Transformer ces valeurs numériques en valeurs linguistiques convenable en utilisant des fonctions d'appartenances qui servent à subdiviser l'espace d'entrée, univers du discours en sous ensembles flous.

## **VI. Base de connaissances**

Elle contient les informations du domaine d'application et le but de contrôle, elle est définie par les deux principales bases suivantes :

### **VI.1. Base de données :**

Elle fournit des informations nécessaires utilisées pour

L'exploration des règles de contrôle flou et la manipulation des données dans un FLC.

### **VI.2. Base de règles:**

Elle caractérise le but et la politique du contrôle flou via un ensemble de règles de contrôle flou.

## **VII. Logique de décision**

C'est le noyau du FLC; elle est capable de simuler les décisions humaines basées sur le concept flou et d'inférer des actions de contrôle flou par l'intervention de l'implication floue et des règles d'inférence dans la logique floue.

## **VIII. Défuzzification**

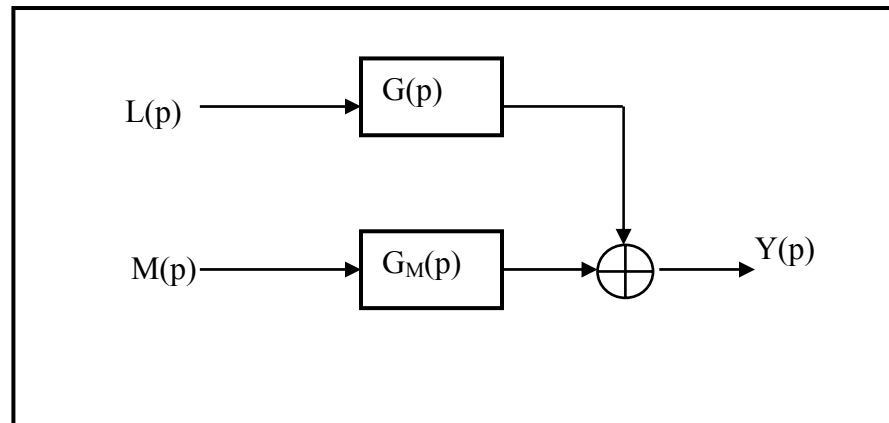
Elle consiste à:

- ✓ Transformer les valeurs de la sortie comprissent dans l'univers du discours en valeurs réelles comprissent dans le domaine de variation.
- ✓ Extraire de la sortie de vérité la valeur numérique de la sortie.

# La commande FeedForward

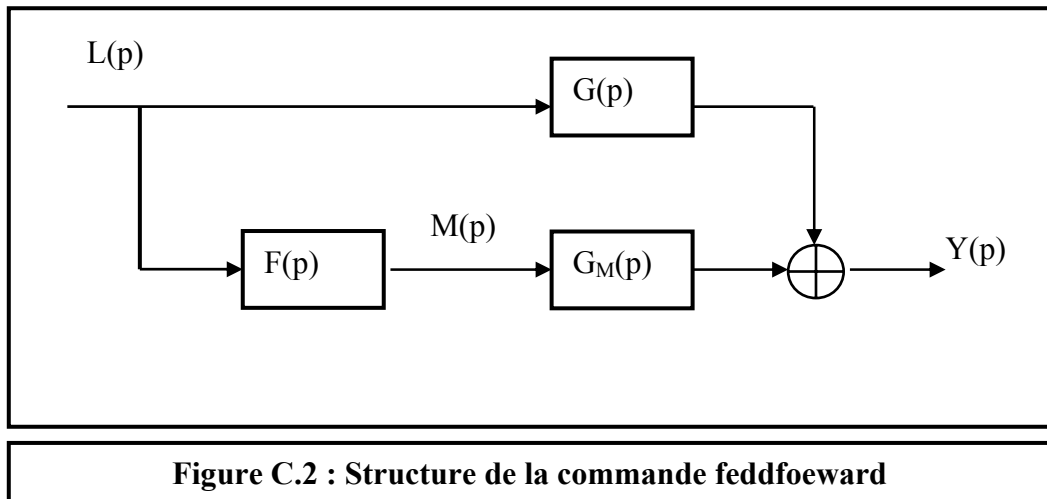
## I. LECONTROL FEEDFORWARD

Soit le système avec bruit schématisé par la figure (C.1), la commande *feedforward* tente de ramener la sortie du processus vers un point de référence en dépit de toute perturbation externe, elle est basée sur une détection anticipée de la perturbation qui agit sur la sortie du système et la production immédiate d'un signal d'ajustement de la commande, La conception d'un tel système consiste à trouver un contrôleur  $F(p)$  assurant une sortie constante en régime permanent et en présence de toute perturbation.



**Figure C.1** Système avec perturbation sur la sortie

La figure (C.2) montre la structure d'une telle commande



Le contrôleur feedforward a pour objectif l'annulation de l'effet de la perturbation  $L(p)$  sur la sortie  $Y(p)$  à l'état permanent, sa fonction de transfert est donnée par :

$$\frac{M(p)}{L(p)} = -\frac{G_L(p)}{G_M(p)} = F(p)$$

Cette formule représente la fonction de transfert du détecteur  $F(p)$  dont le calcul est essentiellement basé sur la connaissance a priori du bruit, cette connaissance représente l'inconvénient majeur de la commande *feedforward*.

## II. FEEDFORWARD-FEEDBACK CONTROL

Dans le milieu industriel on trouve souvent une combinaison entre le *feedforward control* et le *feedback control*, cette combinaison a eu lieu pour atteindre deux objectifs :

- La commande feedforward prend en charge la génération d'un signal d'ajustement de la commande pour les perturbations fréquentes agissant sur la sortie.
- La commande feedback est utilisée pour la correction des erreurs de la sortie causées par la perturbation non détectées par la commande feedforward.

## Les Algorithmes génétiques multi- objectifs

Les algorithmes génétiques multi- objectifs présentent une autre variante des algorithmes génétiques simples dans la mesure où il opère sur des problèmes d'optimisation multi-critères en ajoutant la notion *d'arrangement* et de *dummy fitness* afin de réaliser les opérateurs génétiques.

L'organigramme des MOGA est donné par le schéma (D.1).

Selon cet organigramme on peut définir trois nouvelles étapes additives dans l'organigramme des algorithmes simples, il s'agit :

- ✓ D'une identification des individus non dominant,
- ✓ Du *dummy fitness*
- ✓ De *sharing method*.

### Définition A.1

Soit un problème d'optimisation défini par :

$\min(f_i(x)), i = 1 : n$  Avec des contraintes

$$\begin{cases} g_j(x) = 0, j = 1 : M \\ h_k(x) \leq 0, k = 1 : K \end{cases}$$

Tels que :

$f_i$  sont les fonctions coût et  $x$  est un vecteur de  $p$  composantes représentant les variables de décision.

Un vecteur  $x^1$  est appelé inférieur partiellement qu'un autre vecteur  $x^2$  si :

$$\begin{cases} \forall i f_i(x^1) \leq f_i(x^2) \\ \exists i f_i(x^1) < f_i(x^2) \end{cases}$$

On dit alors que la solution  $x^1$  domine la solution  $x^2$

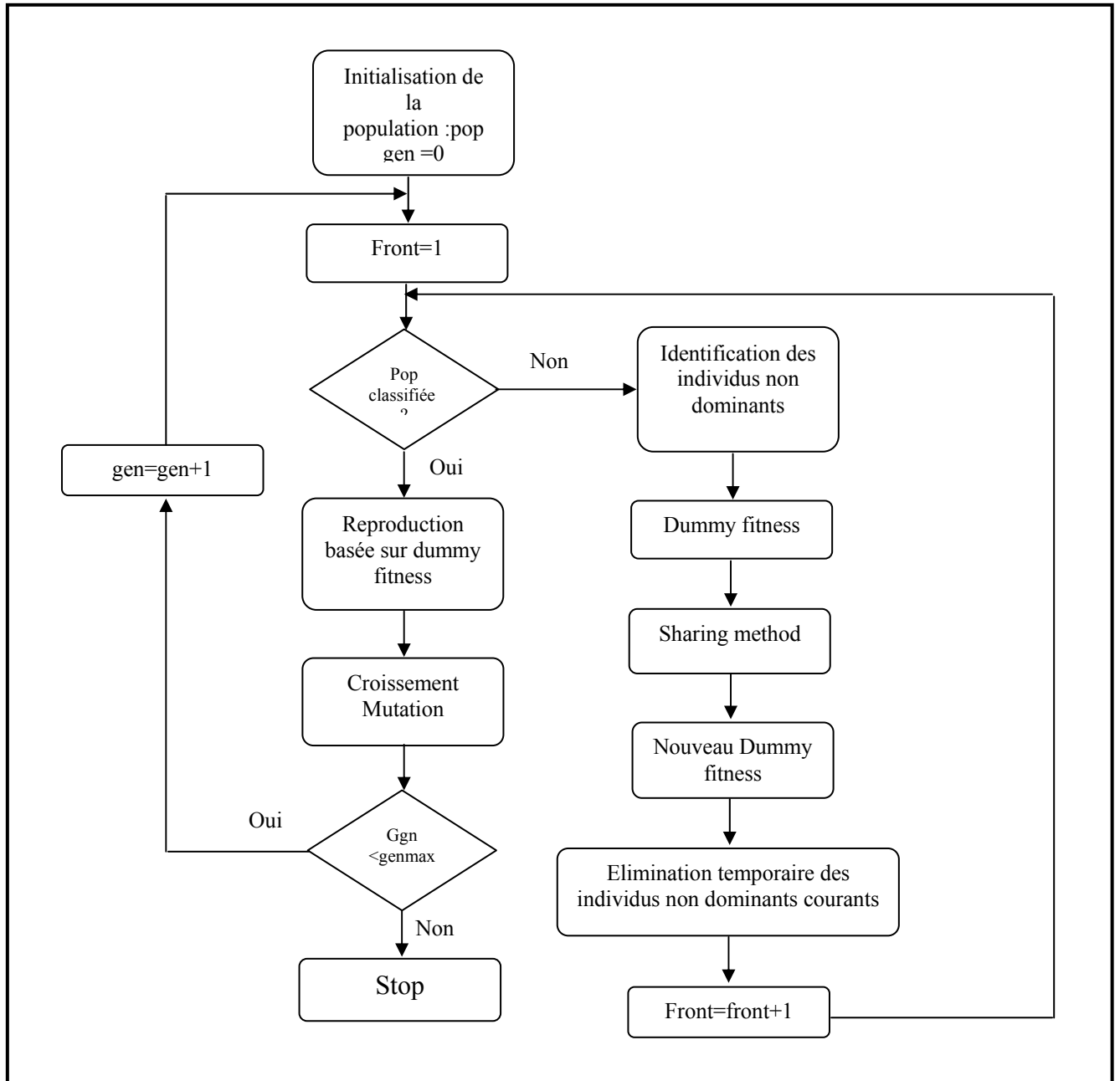


Figure D.1 Organigramme des MOGA

On se basant su cette définition, l’algorithme d’optimisation est décrit comme suit :

1. Identification des individus non dominants, ces individus prennent le front 1 et leur probabilité d’être reproduire est très élevée.
2. On assigne le même *dummy fitness*  $f_i$  pour les individus non dominants  $i$  du front 1.

$$f_i = r$$

$r$  représente le rang du front

3. Procéder par *sharing method* dans le but de maintenir la diversité, il s’agit alors de diviser le *dummy fitness* par une quantité proportionnelle au nombre des individus  $y$  appartenant, soi alors :

$$f_i = \frac{f_i}{m_i}$$

Avec :

$$m_i = \sum_{j=1}^M Sh(d(i, j))$$

$$sh(d(i, j)) = \begin{cases} sh(0) = 1 \\ 1 - \frac{d(i, j)}{\sigma_{share}} \text{ si } d(i, j) < \sigma_{share} \\ 0 \text{ si } d(i, j) \geq \sigma_{share} \end{cases}$$

Où  $\sigma_{share}$  représente la distance phénotypique maximale permise entre deux individus. Sa valeur est choisie par l’utilisateur, cependant Goldberg a proposé une mécanisme adaptatif pour sa détermination.

$d(i, j)$  représente la distance entre deux individus  $i$  et  $j$ , elle est définie par :

$$d(i, j) = \sqrt{\sum_{k=1}^{np} \left( \frac{x_k^i - x_k^j}{\max(k) - \min(k)} \right)^2}$$

$np$  représente le nombre de paramètres définissant le chromosome  $i$  et  $j$  ;  $x_k^i$  est la valeur réelle du paramètre  $k$  pour le chromosome  $i$  ;  $\max(k) = \max_{i=1..M} x_k^i$  et

$\min(k) = \min_{i=1..M} x_k^i$  où  $M$  représente le nombre d’individu dans un front

4. Eliminer temporairement ces individus non dominants de la population et procéder par une deuxième classification en associant au départ à tout les individus le même *dummy fitness* défini par  $\min \frac{f_i}{m_i}$  trouvé dans le front passé
5. Le processus est refait jusqu'à ce que toute la population soit classifiée.
6. Entamé l'optimisation par les algorithmes génétique standard



# Bibliographie

- [1] R. Marino et P. Tomei, "*Nonlinear control design*", Prentice-Hall, Europe, 1995.
- [2] J.J.E. Slotine et W. Li, "*Applied Nonlinear Control*", Prentice-Hall, Englewoods Cliffs, New-Jersey, 1991.
- [3] J.M. Maciejowski, "*Multivariable Feedback Design*", Addison-Wesley Publishing Company ", Great Britain, 1994
- [4] E. C. Chen and H.K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks", IEEE trans. Automatic. Control Vol. 40, N°5, pp 791-801, 1995
- [5] S. S. Sastry and A. Isidori, "Adaptive control of linearizable systems", Trans. Automatic. Control Vol. 34, pp 1123-1131, 1989
- [6] K. Nam and A. Arapostathis, "A model reference adaptive control scheme for pure-feedback nonlinear systems", IEEE Trans. Automatic. Control Vol. 33, pp 791-803-811, 1988
- [7] D. N. Godbole and S.S. Sastry, "Approximate Decoupling and asymptotic tracking for MIMO systems", IEEE Trans. Automatic. Control, Vol. 40, N°3, pp441-450, 1995
- [8] C. C. Liu and F. C. Chen, "Adaptive control of non-linear continuous time systems using neural networks- general relative degree and MIMO cases," International Journal of control, Vol. 58, pp. 317-335, 1993.
- [9] F. C. Chen and H. K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," IEEE Trans. Automat. Control, vol. 40, no. 5, pp. 791-801, 1995.

- [10] S. S. Ge, C. C. Hang and T. Zhang, "Stable Adaptive Control for Nonlinear Multivariable Systems with a Triangular Control Structure", Proceedings of the 38<sup>th</sup> Conference on Decision & Control, Phoenix, Arizona USA – pp 5176-5581, December 1999
- [11] S. Jain and F. Khorrami, "Decentralized adaptive output feedback design for large-scale nonlinear systems," IEEE Trans. Automat. Contr., vol. 42, pp. 729–735, May 1997.
- [12] J.T. Spooner and K. M. Passino, "Decentralized adaptive Control of Nonlinear System Using Radial basic Neural Networks", IEEE transactions on automatic control, vol. 44, no. 11, pp 2050-2057, Nov 1999
- [13] Hassan K. Khalil, "*Nonlinear systems*", Macmillan publishing company, 1992,
- [14] J. M. Coron, L. Praly, A. R. Teel, "*Feedback stabilization of non linear system: sufficient conditions and Lyapunov and input output technique*", Springer Verlag 1995
- [15] Ludovic FAUBOURG, « *Constructions de fonctions de Lyapunov contrôlées et stabilisation non linéaire* », thèse de Doctorat en sciences, Dec 2001
- [16] A. Ioannou et J. Sun, "*Robust adaptive control*", Prentice-Hall, Upper saddle river, New-Jersey, 1996
- [17] L. A. Zadeh. "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes", IEEE Transactions on Systems, Man and Cybernetics., Vol. SMC-3, n°1, pp 28-44, 1973.
- [18] E. H. Mamdani, "Application of fuzzy algorithms for control of a simple dynamic process", *IEE Proc. Control theory and applications*, 121, pp.1585-1588, 1974.
- [19] C.C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller -Part I and II," *IEEE Trans. Syst. Man, Cybern.*, SMC-20, pp. 404 - 418, 1990.
- [20] A. Homaifar and Ed. McCormick, "Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms", *IEEE Trans. Fuzzy Systems*, Vol. 3 N° 2, pp. 129-139, 1995.
- [21] C.L. Karr and E.J. Gentry, "Fuzzy control of pH using genetic algorithms", *IEEE Trans. on Fuzzy Systems*, vol. 1, N° 1, pp. 46-53, 1993.
- [22] J. Kim and B.P. Zeigler, "Designing Fuzzy logic controllers using a multiresolutional

- search paradigm", *IEEE Trans. on Fuzzy Systems*, Vol. 4, N° 3, pp.213-226, 1996.
- [23] M. A. Lee and H. Tagaki, "Integrating design stages of fuzzy systems using a genetic algorithm", Proc. 2<sup>nd</sup> IEEE Int. Conf. Fuzzy Syst. (FUZZY/IEEE'93), vol 2. New York, pp 612-617, 1993.
- [24] D.A. Linkens and H.O. Nyongesa , "Genetic Algorithms for Fuzzy control- Part I and II", *IEE Proc. Control Theory and applications*, Vol. 142 , N° 3, pp. 161-176, 1995.
- [25] T. Nishiyama, T. Tagaki, R. Yager and S. Nakanishi,"Automatic generation of fuzzy inference rules by genetic algorithm", Proc 8<sup>th</sup> Fuzzy Syst. Symp, pp.237-240., 1992.
- [26] P. Thrift , "Fuzzy logic synthesis with genetic algorithms", Proc. of the fourth Int. Conf. on Genetic Algorithms, pp. 509-513, 1991.
- [27] F. Rosenblat, "The perceptron: a probabilistic model for information storage and organization in the brain", *Psychological review*, 65, pp 386-408, 1958
- [28] L. X. Wang, «design of adaptive fuzzy controllers for nonlinear systems by input-output linearization», The Industrial Fuzzy Control and Intelligent Systems Conference and the NASA Joint Technology pp 89 – 93, USA, 18-21 Dec.1994
- [29] L.X. Wang, and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning ", *IEEE Transaction on Neural Networks*, Vol. 3, pp 807-814, 1992.
- [30] Q. Zhang and A. Benveniste, "Wavelet networks", *IEEE Transaction on Neural Networks*, Vol.3, pp-889-898 ,1992.
- [31] Y.M. Cheng et B.S. Chen, "Adaptive Wavelet Network Control Design for Nonlinear Systems", *Proc. Natl. Sci. Coun. ROC(A)*, Vol 22, N°6, PP.783-799, 1998
- [32] J. Doyle, K. Glover, P.P. Khargonekar, B.A. Francis, "State space solution to standard  $H_2$  and  $H_\infty$  control problems", *IEEE Transaction on Automatic and Control*, Vol. 34, No. 8, pp.831-847, 1989.
- [33] A.A. Stoorvogel, "*The  $H_\infty$  Control Problem: A State Space approach*", Prentice Hall, New York, 1992.
- [34] C. K. Chui, "*An Introduction to Wavelets*", Academic Press, New York, New York, USA 1992.

- [35] K. S. Narendra, A. M. Annaswamy. "*Stable Adaptive Systems*", Prentice-Hall, International Editions, 1989.
- [36] T. J. Koo, «Stable model reference adaptive fuzzy control of a class of nonlinear systems». IEEE Trans. on fuzzy systems, Vol. 9 N°4, pp 624-636, August 2001
- [37] J.R. Layne and K.M. Passino, "Fuzzy model reference learning control for cargo ship steering", IEEE Contr. Sys. Mag., vol. 13 N°12, pp 23-34 1993
- [38] Y. WANG et, T. CHAI, «Output-Feedback Control of Uncertain Nonlinear Systems Using Adaptive Fuzzy Observer», Proc of American Control Conference, Vol.4, pp 2613 - 2618 June 8-10, Portland, OR, USA. 2005
- [39] M. Kistic,, I. Kanellakopoulos and .P, Kokotovic "*Nonlinear and adaptive control design*". John Wiley, new York 1995.
- [40] P.V. Kokotovic, "The joy of feedback: Nonlinear and control", IEEE Control systems, pp 7- 17, June 1992.
- [41] E.D. Sontag, " A universal construction of Arstein's theorem on nonlinear stabilization", system and control letters vol. 13 , pp 117-123, 1989
- [42] Z. Artstein, " Stabilization with relaxed control", Nonlinear analysis TMA, Vol. 7 N°11, pp 1163-1173, Nov.1983
- [43] V. Jurjevic," and J.P. Quinn, "Controllability and stability" Journal of differential equations, Vol. 28, pp 381-389, 1978
- [44] R. Freeman and L. Praly, "Integrator backstepping for bounded controls and control rates", IEEE Trans. On Automatic and Control, Vol 43, N° 2, Feb 1998
- [45] K. Ezal, Z.Pan, "Locally Optimal and robust backstepping design", IEEE Trans. On Automatic and Control, Vol. 45, N° 2, pp 260-271, Feb 2000
- [46] W.L. Luyben, "*Process Modelling, Simulation, and Control for Chemical Engineers*", McGraw Hill Book Company, les Etat Unis, 1973
- [47] A. Benaskeur et A. Desbiens, "Modified Error Decentralized Control with Observer Backstepping", International Journal of Control, Vol. 75, No. 12, pp. 923-931, 2002.
- [48] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems ans its Application to Modeling and Control", IEEE Trans. Syst. Man, and Cybern, vol. SMC-15, N° 1,

- Jan/Feb. pp 117-130,1985
- [49] D. E. Goldberg, "*Genetic algorithms in Search, Optimization and Machine learning*", Addison-Wesley, 1989.
- [50] Z. Michalewicz. "*Genetic Algorithms + data structures = evolution programs*", Springer Verlag, 1992.
- [51] J.D. Schafer, "*Some experiments in machine learning using vector evaluated genetic algorithms*", Unpublished doctoral dissertation, Vanderbilt University, Nashville (TN), 1984
- [52] D.E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization", Proc. Of second intern. Conf. on genetic algorithm, pp 41-49 Morgan Kaufmann, 1987
- [53] Horn and N. Nafpliotis, "Multi- objective optimization using the niched pareto genetic algorithm", In First IEEE conference on evolutionary computation, IEEE world congress on computational intelligence, volume 1, 1994.
- [54] N. Srinivas and K. Deb, "Multi- objective optimization using non-dominated sorting in genetic algorithm", Evolutionary Compute, Vol. 2, N°3: pp 221-248, 1995.
- [55] N. Marco, J.A. Désidéri and S. Lanteri, "*Multi-Objective Optimization in CFD by Genetic Algorithms*". Rapport de recherche, INRIA, 1999
- [56] D. Dochain, "*Automatique des bioprocédé*", Edition Hermes sciences Europe Ltd, 2001
- [57] J.E. Bailey and D.F. Ollis, "*Biochemical Engineering Fundamentals*", McGraw Hill, New york, 1986
- [58] F. Nejjari, C. BenYoussef, A. Benhammou and B. Dahhou, "Procedures for state and parameter estimation of a biological wastewater treatment", CESA'96 IMACS, Vol.1, pp. 238-243, Lille France, July 9-12, 1996.
- [59] G. Olsson, "State of the art in sewage treatment plant control". AIChE Symposium Series, n°159, Vol.72, pp.52-76, 1976
- [60] Y.C. Chang, "A nonlinear mixed  $H_2/H_\infty$  control for robust tracking design of robotic systems". Int. Journal on Control, Vol. 67, N°6, pp837-857, 1997
- [61] B.S. Chen, T. S. Lee and J.H. Feng, "A nonlinear  $H_\infty$  control design in robotic systems

under parameter perturbation and external disturbance”. Int. Journal on Control, Vol. 59, N°2, pp439-461, 1994

[62] B.S. Chen, Y.C. Chang and T. S. Lee, “Adaptive control in robotic systems with  $H_\infty$  tracking performance”. Automatica, Vol. 33 N°2, pp227-234, 1997

[63] J.T. Spooner and K.M. Passino, “Adaptive control of a class of decentralized nonlinear systems”. IEEE trans. Automatic and control, Vol. 41 pp280-284, Feb 1996