

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari BOUMEDIENE



Faculté d'Electronique et d'Informatique

MEMOIRE

Présentée pour l'obtention du diplôme de

MAGISTERE EN ELECTRONIQUE

Spécialité : Contrôle de processus et Robotique

Par : BERNI Amel

Thème:

**CONTRIBUTION A LA COMMANDE HYBRIDE
FORCE/POSITION D'UN BRAS MANIPULATEUR
REDONDANT**

Soutenu le **16 juin 2004** Devant le jury composé de :

M^r R.TOUMI
M^{me} N. Saadia
M^r F. BOUJEMAA
M^r N. HAMERLAIN
M^r H. REZINE
M^r A. HADJ- LARBI

Professeur, U.ST.H.B
Maître de Conférence
Professeur, E.N.P
D^r. Es-Sciences, C.D.T.A
Docteur, E.M.P
Chargé de Cours, U.S.T.H.B

Président
Directeur de Thèse
Examineur
Examineur
Examineur
Examineur

Je dédie ce mémoire

*A mes grands parents, A mes parents,
A ma sœur et mon frère,
A mes tantes et oncles,
A mon époux,
A mes amis.*

REMERCIEMENTS

Je tiens en premier à remercier, le Bon Dieu le Tout Puissant de m'avoir donné le courage et la patience afin de mener à bien ce modeste travail.

Qu'il me soit permis de remercier et d'exprimer ma profonde gratitude à M^{me} N.Saadia pour m'avoir confié ce travail et pour la confiance son limites qu'elle m'a accordé .

Je remercié Monsieur le Professeur TOUMI pour m'avoir fait l'honneur de présider ce mémoire. Je remercié également Monsieur le Professeur BOUJEMAA; Monsieur le Professeur HAMERLAIN, Monsieur le Colonel REZINE et Monsieur HADJ-LARBI qui ont bien voulu accepter d'examiner mon travail et participé au jury.

Je remercié aussi Mme I.BOUDOUANE et Mr A.MEZIANI pour leurs aides.

J'exprime ma profonde reconnaissance à Mme A.ABABOU et Mr N.ABABOU ; aux enseignants de l'Ecole Polytechnique plus particulièrement Mr NEZLI, Mr BOUKHETALA et Mr TADJINE du laboratoire Robotique.

Mes remerciements s'adresse aussi à mes amis IBRAHIM, SAMIR , MERIEME, FARIDA et FADELA pour tous leurs aides.

Mes remerciements vont également à tous ce qui ont contribué à ma formation de prés ou de loin en particulier ceux de la faculté d'électronique et Informatique.

Merci à vous tous

Table des matières

<i>Introduction Générale</i>	1
<i>Chapitre I: POSITIONNEMENT DU PROBLEME</i>	
I.1 Introduction	5
I.2 Définition d'une tâche compliant	5
I.3 Compliance active	6
I.4 Complexité du contrôle en effort: Cas de la saisie et du transport d'un objet	7
I.5 Conclusion	10
<i>Chapitre II: MODELISATION D'UN BRAS MANIPULATEUR REDONDANT</i>	
II.1 Introduction	11
II.2 Robot manipulateur redondant	11
II.3 Modélisation géométrique	13
II.4 Modélisation Cinématique du Robot	14
II.5 Modélisation dynamique	15
II.6 Simulations du modèle dynamique du bras manipulateur redondant en boucle ouverte	18
<i>A. Modèle dynamique à vide (sans charge)</i>	18
<i>B. Modèle dynamique avec charge</i>	20
II.7 Générateur de trajectoire	21
II.8 Conclusion	23
<i>Chapitre III: INVERSION DU MODELE GEOMETRIQUE</i>	
III.1 Introduction	25
III.2 Positionnement du problème	25
A. Méthode utilisant une identification du modèle	26
B. Méthode de la boucle fermée	26
B.1 Etude de stabilité	27
B.2 Les différents choix de la matrice M	28
B.2.1 Méthodes algorithmiques directes	28
B.2.2 Méthodes basées sur une approximation linéaire	30
III.3 Commentaires et comparaison	33
III.4 Application de la méthode de levenberg-Marquardt dans le cas d'un bras manipulateurs redondant	33
A. Cas de la trajectoire de type arc cercle	34
B. Cas de la trajectoire de type droite	36
C. Cas de la trajectoire cercle	37
III.5 Application d'une approche connexionniste	38

III.5.1. Réseau de neurones à mémoire	39
III.5.2. Principe de la méthode proposée	39
III.5.3. Architecture du réseau utilisé	41
III.5.4. Construction du réseau de neurones	41
• Construction de la base d'exemples	42
• Apprentissage	42
• Test de généralisation	43
III.5.5. Simulation	43
III.6 Conclusion	45

Chapitre IV: COMMANDE HYBRIDE FORCE-POSITION

IV.1. Introduction	46
IV.2. Principe de la commande hybride force-position	46
IV.3. Structure de commande hybride force-position	48
<i>IV.3.1. Commandes hybrides force-position</i>	48
a) Structure de commande de Craig et Raibert	49
b) Structure de commande de Reboulet et Robert	50
c) Structure de commande de khatib	51
<i>IV.3.2. Structure de commande en effort externe</i>	52
<i>IV.3.3 Etude comparative</i>	53
IV.4. Synthèse de la structure de commande en effort externe	54
• Le Contrôleur de Position	55
• Le Contrôleur d'effort	56
• Schéma global de la commande en effort externe	57
IV.5. Conclusion	57

Chapitre V: SYNTHÈSE DE COMMANDES HYBRIDE FORCE / POSITION

Introduction	59
--------------	----

Première partie: Commande hybride Force - Position	
utilisant l'approche de La commande dynamique	61
V.I.1 Principe de la commande dynamique	61
V.I.2 Simulation de la commande dynamique	63
• Simulation de la commande dynamique dans un espace articulaire	65
➤ Mouvement dans un espace libre sans charge	65
➤ Mouvement dans un espace libre avec charge	67
➤ Défaillance de l'un des régulateurs (ou actionneur)	
des bras du robot manipulateur redondant	70
Conclusion 1	73
• Simulation de la commande dynamique dans l'espace cartésien	73
➤ Mouvement dans un espace cartésien libre sans charge	73
➤ Mouvement dans un espace cartésien libre puis contraint	76
a. Suivi d'un contour rectiligne en maintenant un effort constant	76

b- Suivi d'un contour curviligne en maintenant un effort constant.	81
c- Saisie et transport d'un objet	83
Conclusion 2	87
V.I.3 Conclusion	88

Deuxième partie: Commande hybride Force - Position utilisant les concepts de la logique floue

V.II.1 Commande par Logique Floue	89
1.1. Structure d'un régulateur logique flou	89
1.2. Contrôleur flou	90
1.3. Types de régulateurs flous	91
V.II.2 Stratégie de commande	91
V.II.3 Synthèse de la Commande par logique floue	93
a- Entrées et Sorties du système à commander	93
b- Entrées et Sorties du régulateur par la logique floue	93
c- Les fonctions d'appartenance	93
d- Défuzzification	94
V.II.4 Simulation de la commande par la logique floue	95
➤ Commande par la logique flou dans un espace articulaire	95
• Mouvement dans un espace libre sans charge	95
• Mouvement dans un espace libre avec charge	97
• Cas d'une défaillance de l'un des régulateurs (ou actionneur) des bras du robot manipulateur redondant	100
➤ Commande par la logique floue dans un espace cartésien	102
• Mouvement dans un espace cartésien libre sans charge	102
• Mouvement dans un espace cartésien libre puis contraint	105
a. Suivi d'un contour rectiligne en maintenant un effort constant	105
b. Suivi d'un contour curviligne en maintenant un effort constant	109
c. Saisie et transport d'un objet	111
V.II.5 Conclusion	114

Troisième partie: Commande hybride force-position utilisant les concepts des réseaux Neuro-Flou.

V.III.1 Représentation neuronale des règles floues	115
V.III.2 Structure de commande	116
V.III.3 Synthèse des régulateurs neuro-flous	117
V.III.4 Algorithme de commande	117
V.III.5 Algorithme d'apprentissage	118
V.III.6 Simulation de la commande par réseaux neuro-flous	119
➤ Commande par la logique flou dans un espace articulaire	119
• Mouvement dans un espace libre sans charge	119
• Mouvement dans un espace libre avec charge	121
• Cas d'une défaillance de l'un des régulateurs (ou actionneur) des bras du robot manipulateur redondant	124
➤ Commande par la logique floue dans un espace cartésien	126
• Mouvement dans un espace cartésien libre sans charge	126
• Mouvement dans un espace cartésien libre puis contraint	128
a. Suivi d'un contour rectiligne en maintenant un effort constant	128
b. Suivi d'un contour curviligne en maintenant un effort constant.	132
c. Saisie et transport d'un objet	134

V.III.7 Conclusion	136
Quatrième partie: Etude comparative des structures de commandes proposées	137
<i>Conclusion</i>	141
Chapitre VI: COOPERATION DE DEUX ROBOTS MANIPULATEURS REDONDANTS	
VI.1 Introduction	142
VI.2 Description de la tâche	143
VI.3 Architecture de commandes coopérante	143
VI.4 Architecture de commande adoptée	145
VI.5 Commande neuro-floue de deux bras manipulateurs redondants	148
VI.6 Conclusion	153
Conclusion général	155
Annexe 1 Transformation de Denavit-Hartenberg	158
Annexe 2 Equations de propagation et de rétropropagation d'un réseau de neurones à mémoires multicouche	160
Annexe 3 Conférence Internationale	164
Références bibliographiques	165

Chapitre I

POSITIONNEMENT DU PROBLEME

POSITIONNEMENT DU PROBLEME

Introduction

L'automatisation d'opérations complexes impliquant des contacts entre le robot et son environnement nécessite l'exécution de tâches où s'alternent des mouvements libres puis contraints. Le transport de pièce ou l'assemblage de pièces volumineuses nécessitant la coopération de plusieurs robots, constituent des exemples typiques de ce type de tâches.

La réalisation d'une tâche requiert un positionnement relatif très précis des effecteurs des bras manipulateurs, or cette précision est obtenue au détriment du prix des robots. De plus, l'indexation des robots par rapport à leurs environnements n'est jamais parfaite, car elle nécessite le développement d'éléments onéreux de péri-robotique (capteurs extéroceptifs). Une solution possible consiste à mesurer les efforts qui s'exercent entre les organes terminaux des robots et les objets manipulés, puis à exploiter ces mesures pour modifier en conséquence la trajectoire de référence. Cette procédure définit ce que l'on appelle **la compliance**.

Une solution classique pour prendre en compte les forces de contact consiste à interposer entre l'organe terminal et l'effecteur un dispositif présentant une certaine élasticité, le dispositif le plus connu est RCC (Remote Compliance Center) développé par Nevins. Cette technique porte le nom de **compliance passive**. Ce dispositif est simple et peu encombrant, mais ses applications restent limitées à cause de sa forte dépendance vis à vis de la tâche à accomplir et sa spécificité en fonction de la pièce à transporter, ce qui constitue un frein à sa généralisation.

Une deuxième solution basée sur la notion de déplacement gardé, consiste à équiper les robots de capteurs qui mesurent les efforts de contacts puis après analyse de ces efforts à définir les modifications des trajectoires de références. On obtient ainsi un mouvement compliant en contrôlant par exemple les efforts de contacts. Ce type de commande qu'il est possible d'implanter sur la majorité des robots actuels, à condition qu'ils soient équipés des capteurs appropriés, se nomme **compliance active**. Cette dernière technique apparaît comme la mieux appropriée pour adapter les robots à différentes classes de tâches.

Définition d'une tâche compliant

Afin de bien clarifier les notions de commande en déplacement et de commande en effort, examinons les deux classes de tâches suivantes:

- Soit un robot qui évolue librement dans les limites de son volume de travail selon des trajectoires sans collision. Dans le cas où l'organe terminal n'est pas en interaction avec un objet quelconque de son environnement, il n'exerce pas d'effort. La commande appropriée est alors une commande en position.

- Soit un robot lié de façon rigide à une pièce encastrée. Dans ce cas, aucun mouvement n'est autorisé, quelle que soit la direction, au risque d'endommager le mécanisme et son environnement. Il peut par contre, dans les limites de ses possibilités, exercer un effort dans n'importe quelle direction. La seule commande possible est alors une commande en effort.

Ainsi, les notions de commande en position et de commande en effort s'excluent mutuellement, car on ne peut contrôler simultanément une force et un déplacement selon une même direction. La solution consiste alors à effectuer une décomposition orthogonale de l'espace de la tâche en deux sous-espaces qui sont respectivement le sous espace des degrés de liberté commandés en position et celui des degrés de liberté commandés en force. Une matrice de sélection permet de choisir quels sont les degrés de liberté commandés en force et ceux commandés en position. On définit alors la compliance comme la somme de ces deux sous-espaces définis par la nature de la tâche à réaliser et par les modèles du robot et de l'environnement [Mas 81].

En résumé, une tâche compliant est définie dans un repère dit repère de compliance qui compte généralement, trois degrés de liberté en position et trois degrés de liberté en force. Pour chaque degrés de liberté, on spécifie soit une position (ou une rotation), soit une force (ou un couple), compte tenu de la tâche spécifiée. Ce repère peut être associé à l'organe terminal, à l'environnement ou à l'objet à manipuler.

Compliance active

Un capteur tridimensionnel d'effort placé au niveau du poignet des robots fournit une information précise sur les forces de contact. Cette information est plus significative que celle qui provient d'autres capteurs, tels que les capteurs de position ou de vision car la variation d'effort en cas de contact est élevée, et cette information est indispensable à la réalisation de tâches telles que la saisie et le transport d'objet. Il est donc nécessaire d'introduire directement dans la commande les informations d'effort.

Les méthodes basées sur une **compliance active** qui ont été développées depuis une dizaine d'années, apportent une grande sécurité pour l'effecteur et son environnement. Dans ces méthodes, la compliance étant introduite au niveau du système de commande, on peut programmer la position du centre de compliance ainsi que les directions suivant lesquelles on veut adapter les robots aux efforts de contact. Les différentes commandes réalisant ce type de compliance peuvent se résumer ainsi [Kha 99]:

- **Commande par raideur active**

Cette commande considère les robots comme un ressort de raideur programmable. L'utilisateur élabore la commande en attribuant un gain élevé aux directions commandées en position, et un faible gain aux directions commandées en effort. La commande mise en œuvre permet de programmer un ressort opérationnel soit dans l'espace articulaire, soit dans l'espace opérationnel.

- **Commande par retour explicite d'effort**

Les robots sont commandés soit en position soit en vitesse, les informations en efforts étant utilisées pour corriger les signaux d'entrées. Deux solutions ont été proposées, selon que la correction s'effectue en position ou en vitesse, soit:

- a- La commande par matrice de raideur où les robots se comportent comme des ressorts.
- b- La commande par matrice d'amortissement où les robots se comportent comme des amortisseurs.

- **Commande en impédance**

La commande en impédance consiste à développer une relation entre les efforts d'interactions et les positions des effecteurs, soit en quelques sorte à contrôler les impédances de sorties des robots. Comme le suggère Hog [Hog.85], le contrôle en impédance peut être basé sur un contrôle de position ou de couple. La première solution requiert des capteurs de positions tandis que la seconde utilise des capteurs d'efforts [Hog 87].

- **Commande hybride force/position**

L'objectif des méthodes décrites précédemment est plus basé sur la réalisation d'un comportement que sur celle d'une consigne. Aucune commande ne contrôle directement les efforts exercés. Nous nous attacherons, dans la suite de cette étude à imposer des efforts désirés sur les systèmes (dans notre cas les deux robots avec l'objet), indépendamment du mouvement. Les commandes hybrides force/position permettent de satisfaire ces contraintes.

Complexité du contrôle en effort: *Cas de la saisie et du transport d'un objet*

La commande en effort d'un robot manipulateur présente des caractéristiques différentes de la commande en position. La dynamique des phénomènes de contact entre le robot et l'environnement est bien plus élevée que celle associée à la commande en position. On peut de ce fait, rencontrer de réels problèmes de stabilité. Le système peut aussi être brutalement en boucle ouverte car le contact avec l'environnement dépend de la position du manipulateur. En effet, à l'inverse d'une commande en position l'information d'effort n'existe pas forcément quelle que soit la configuration du robot. Si pour une raison quelconque, comme par exemple les coefficients du régulateur en effort sont mal réglés, le robot effectue un rebond et perd le contact avec l'environnement, la commande en effort agit alors en boucle ouverte. Ceci peut conduire à un comportement destructif pour le robot et le capteur d'effort selon l'algorithme de commande utilisé. A cela s'ajoute une méconnaissance réelle des caractéristiques mécaniques de l'environnement correspondant à une erreur de modélisation qui peut modifier de manière significative les performances de la loi de commande en effort.

Dans ce mémoire, nous nous focalisons sur la problématique du contrôle en effort des tâches de transport d'objets par deux bras manipulateurs. Ces tâches constituent une classe de tâches compliantes. Elle est ainsi très utilisée en robotique pour évaluer des approches de commande de robots manipulateurs coopérant compte tenu des contraintes qu'elle implique. La mise en œuvre d'une telle opération, nécessite l'exécution des étapes suivantes:

Etape 1: Déplacement de l'organe terminal de chaque robot vers chaque extrémité de l'objet.

Etape 2: Positionnement de l'organe terminal de chaque bras manipulateur sur chaque extrémité de l'objet

Etape 3: Saisie simultanée de l'objet

Etape 4: Transport de l'objet vers une position finale.

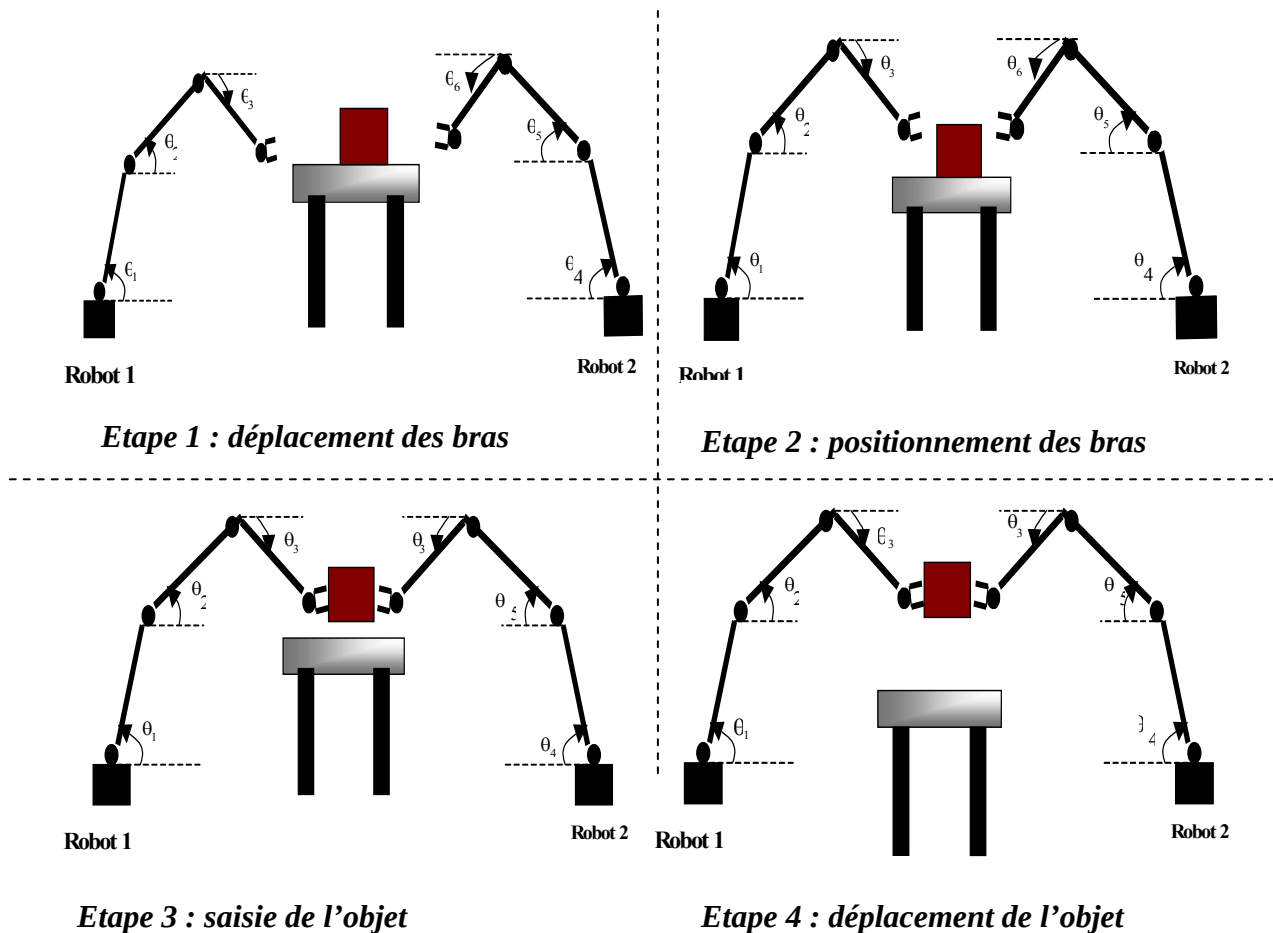


Figure 1. Réalisation d'une tâche de Saisie et de transport d'un objet.

L'analyse de ces étapes fait apparaître les points délicats suivants:

- La transition entre les mouvements libres et les mouvements contraints est source de perturbations impulsionnelles qui peuvent introduire des rebonds ou des glissements lors de la mise en contact, ce qui induit des vibrations dans la structure mécanique.
- La variation brusque de la géométrie du contact, dont l'occurrence temporelle n'est pas facile à prévoir avec précision, entraîne une modification significative du comportement de l'interaction.

L'analyse des étapes précédemment citées fait apparaître que dans le cas d'une tâche de saisie et déplacement d'objet ces difficultés augmentent. En effet, Pour que la tâche soit

réalisable il est indispensable que les deux robots entrent en contact avec l'objet simultanément et que le contact ne soit pas brusque. De plus durant le transport de l'objet il est nécessaire que les mouvements des deux robots soient coordonnés afin d'éviter la déformation de l'objet (voire sa destruction) ou encore la détérioration des organes terminaux des robots. De plus, ces difficultés augmentent dès que le système doit saisir des objets volumineux, la manipulation d'objets lourds (ou possédant un balourd important), le transport et la déformation « en l'air » d'objets souples ou flexibles ou la réalisation d'assemblage en l'air. Afin de remédier à ces problèmes et d'augmenter le nombre d'applications industrielles, il est nécessaire de réaliser une coopération entre plusieurs robots manipulateurs. Cette coopération permet ainsi la réalisation de tâches qu'un seul robot ne pourrait effectuer seul.

Dans ce cas, il est impératif de choisir une approche qui a pour principe général d'introduire une compliance active dans le système de commande. De nombreuses applications nécessitent de commander la position de l'effecteur et les efforts d'interactions avec l'environnement. En effet, généralement une commande en position seule, n'est pas suffisante pour les tâches complexes à accomplir. Il est donc nécessaire de commander aussi les efforts de contact sur les objets transportés. Dans ce cas, le contrôle d'effort dépend explicitement de la tâche, au cours du mouvement les objets flexibles doivent être maintenus en extension ou en pression constante. Les objets rigides peuvent être tenus uniquement par les effecteurs des robots, en exerçant des efforts sur sa surface, mais ces efforts ne doivent pas être trop élevés pour ne pas l'endommager.

En plus du problème précédemment cité, lors de l'accomplissement d'une tâche, il est nécessaire de connaître les trajectoires avant tout déplacement ou en temps réel. La trajectoire désirée est généralement décrite dans le repère des coordonnées cartésiennes. Elle nécessite donc, d'être convertie dans le repère articulaire afin d'analyser et de contrôler le comportement du système.

Le passage d'un espace à un autre est tout à fait primordial, il nécessite de manière générale la connaissance des modèles directs et inverses du robot. Ces modèles sont souvent très difficiles à identifier en raison de leur caractère fortement non-linéaire et ils mettent en jeu d'importants volumes de calculs. Si nous obtenons relativement facilement les modèles directs, les solutions inverses n'ont pas a priori, d'expression analytique. Retrouver le modèle inverse est encore plus difficile si le robot est redondant. En effet, à une situation donnée de l'organe terminal d'un robot redondant correspond une infinité de configurations possibles qui permettent à l'effecteur de réaliser la tâche opérationnelle désirée de plus il est nécessaire d'en choisir qu'une. Le modèle géométrique inverse peut généralement être approché par des solutions numériques. On peut distinguer deux méthodes permettant l'inversion du modèle géométrique:

- **La première méthode réalise une identification du modèle (notée par IP):** Son principe est une identification hors ligne des paramètres du modèle inverse d'une fonction $F(q)$. Dans ce cas, l'erreur cartésienne e permet de modifier les paramètres du modèle inverse pour obtenir sa convergence vers F^{-1} [Ben 02].
- **La seconde méthode est dite boucle fermée BF.** C'est une boucle dynamique d'asservissement de la variable opérationnelle x dont la consigne est la variable désirée x_d . Son principe consiste à faire une approximation linéaire du modèle

inverse. Selon les cas présentés on cite la méthode basée sur le modèle différentiel inverse et la méthode de Levenberg-Marquardt. L'approximation linéaire du premier ordre ne conduit pas toujours à la solution désirée. Des divergences ou des oscillations peuvent se produire, d'où la nécessité de passer dans ce cas au modèle quadratique du second ordre pour assurer la convergence.

Dans cette présente étude, nous nous intéresserons dans un premier temps à la résolution du modèle inverse. En effet, nous proposons deux méthodes (Levenberg-Marquardt et par identification en utilisant les réseaux de neurones à mémoires) pour calculer le modèle géométrique inverse d'un robot manipulateur redondant à trois liaisons rotoïdes. Le modèle géométrique ainsi calculé sera utilisé dans les structures de commandes proposées dans la suite de notre travail.

Le deuxième problème à résoudre consiste en la commande du bras manipulateur redondant. En effet, dans une seconde étape nous traitons le problème de la commande hybride force position du bras manipulateur redondant en utilisant différentes approches.

Conclusion

Dans ce premier chapitre, nous avons positionné le problème qui consiste à commander deux bras manipulateurs redondants réalisant une tâche de saisie et de transport d'un objet. Cette tâche est choisie pour sa complexité de réalisation. En effet, nous avons mis en évidence que sa réalisation nécessite une bonne coordination entre les robots ainsi qu'une commande hybride force / position de chaque bras.

De plus une méthodologie de calcul du modèle géométrique inverse est nécessaire compte tenu de la redondance du système et de la spécification de la tâche dans le repère opérationnel. En effet, la transformation de l'espace opérationnel vers l'espace articulaire des trajectoires du robot manipulateur dans ce cas est indispensable et est un problème d'actualité.

Suite à l'analyse de la tâche proposée, nous diviserons notre travail en deux grandes parties: la première traite le problème de l'inversion du modèle géométrique, la seconde traite le problème de la commande du bras manipulateur considéré.

La synthèse en simulation de lois de commande nécessite une représentation mathématique du système considéré, le prochain chapitre portera donc sur la représentation mathématique des modèles d'un bras manipulateur redondant à trois liaisons rotoïdes.

MODELISATION D'UN BRAS MANIPULATEUR REDONDANT

II.1 Introduction

Pour effectuer l'analyse et la synthèse d'un système dynamique, il est nécessaire de connaître les relations entre ses grandeurs d'entrées et de sorties. L'ensemble de ces relations constitue le modèle mathématique du système. L'efficacité de ce modèle repose sur une analogie entre le comportement des objets physiques et celui des équations mathématiques.

Dans le cadre de l'automatique, modéliser un système consiste à établir un ensemble de relations mathématiques qui permettent de décrire, avec une précision suffisante, les interactions internes et les interactions entre ce système et son environnement extérieur. Lorsque les relations sus-citées sont issues des équations de la physique, le modèle obtenu est dit modèle de connaissance. Si ces relations découlent des observations disponibles sur un système, on aboutit à un modèle de représentation, obtenu par identification.

Dans le domaine de la robotique l'élaboration du modèle nécessite une étude approfondie et détaillée sur la structure du robot. L'utilisation de la transformation de Denavit-Hartenberg (noté: DH) facilite la description géométrique du manipulateur. Ceci nous permet d'aboutir aux modèles cinématique et géométrique direct et inverse du robot. La même transformation offre une souplesse dans le calcul du modèle dynamique direct, en utilisant le formalisme d'Euler-Lagrange (noté: EL), et du modèle dynamique inverse, en utilisant le principe D'Alembert [Reb 84].

Pour notre étude nous avons utilisé un Robot Redondant à trois bras dont ses trois articulations sont rotoïdes comme le montre la figure 1.

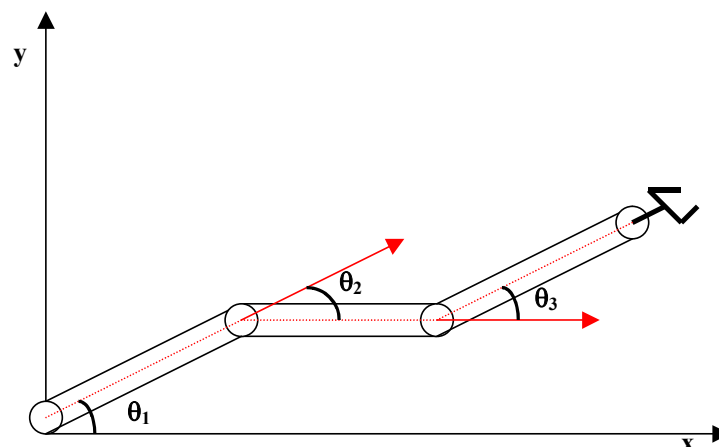


Figure 1. Présentation du bras manipulateur à trois liaisons rotoïdes.

II.2 Robot manipulateur redondant

On appelle robot redondant [Kha 99], un robot possédant plus d'actionneurs que le nombre de degrés de liberté, autrement dit un robot est cinématiquement redondant lorsque le nombre n d'articulations indépendantes est supérieur aux m degrés de libertés que possède son organe terminal ($n > m$). $R = n - m$ constitue le degré de redondance du robot [Ser 89c].

La structure cinématique des robots redondants autorise le déplacement de ses articulations sans bouger l'organe terminal. En conséquence, les robots cinématiquement redondants possèdent la propriété fondamentale suivante : à une situation donnée de l'organe terminal correspond une infinité de configurations possibles qui permettent à l'effecteur de décrire une même trajectoire opérationnelle désirée. La figure 2 montre par exemple trois configurations possibles parmi l'ensemble des solutions pour un robot à 3 degrés de libertés dans le plan [Ama 97].

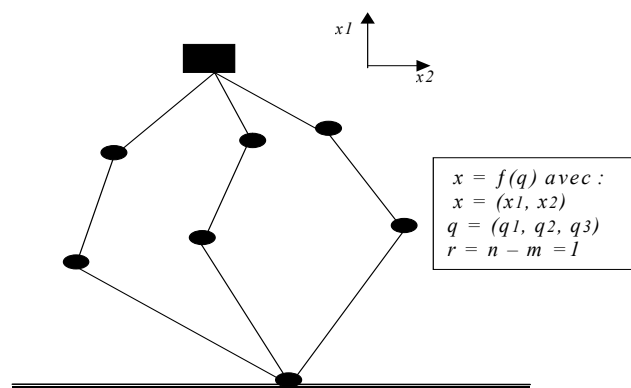


Figure 2. Différentes configurations pour un robot redondant.

Ce type de robot permet d'exécuter une tâche avec plus de facilité, dans certains cas contraignant, qu'avec les robots non-redondants. En effet, le mouvement du robot peut être assujéti à des contraintes. Ces derniers sont souvent liés à son environnement qui est, en général, encombré d'objets fixes ou mobiles. Durant l'exécution de la trajectoire, c'est l'ensemble de ses articulations qui doit éviter le contact avec les obstacles (Figure 3). Cela est possible dans le cas d'un robot redondant car il peut atteindre chaque zone sous différents aspects de configuration.

Les contraintes peuvent également être liées à la morphologie du robot. Il s'agit par exemple des butées articulaires inhérentes à sa structure mécanique. Le robot peut être amené aussi à satisfaire un ou plusieurs critères objectifs nécessaires à l'exécution de la tâche, comme: l'évitement de certaines configurations singulières, la minimisation de l'énergie cinétique, l'accroissement de sa manipulabilité etc.

Pour toutes ces raisons, il est plus judicieux d'utiliser des robots redondants pour exécuter deux types de tâches : primaire et secondaire.

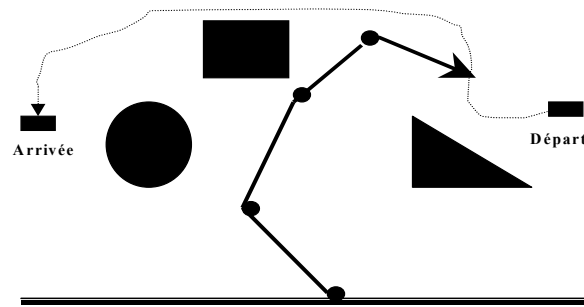


Figure 3. Un robot redondant peut éviter les obstacles de son environnement

La tâche primaire, dite aussi prioritaire, est en général le suivi de trajectoire. La tâche secondaire, dite de contrainte ou de priorité inférieure, consiste à réaliser d'autres critères objectifs et/ou tenir compte de contraintes concernant l'environnement, la morphologie du robot ou la tâche à exécuter elle-même.

II.3 Modélisation géométrique :

A. Modélisation géométrique directe

Les transformations précédentes peuvent être utilisées pour établir les relations entre les coordonnées cartésiennes et les coordonnées généralisées. En effet, le modèle géométrique consiste à résoudre l'équation de transformation afin de trouver la position de l'organe terminal en terme d'angles et de déplacements entre les articulations c'est-à-dire [kha 99] [Coi 81]:

$$x = f(q) \dots \dots \dots (II.1).$$

Où « f » est une application de l'ensemble des configurations articulaires atteignables vers l'espace cartésien.

$$f(q) \in \mathbb{R}^m \quad f(q) = [f_1(q), f_2(q), \dots, f_m(q)]^T \dots \dots \dots (II.2).$$

Les angles et déplacements sont appelés : coordonnées articulaires alors que les positions de l'organe terminal (outil de travail) dans l'espace sont décrites comme étant des coordonnées opérationnelles ou cartésiennes [Ren 84].

Dans le cas du bras manipulateur redondant à trois axes rotoïdes considéré dans la présente étude, le modèle géométrique est donné par l'équation suivante :

$$X = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) + L_3 \cos(\theta_1 + \theta_2 + \theta_3) \\ L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2) + L_3 \sin(\theta_1 + \theta_2 + \theta_3) \end{bmatrix} \dots \dots \dots (II.3).$$

Avec $L_1=50\text{cm}$, $L_2=40\text{cm}$ et $L_3=30\text{cm}$.

B. Modélisation géométrique inverse

On sait donc en général écrire le modèle géométrique direct, notamment par les matrices. Mais pour déterminer les variables articulaires qui placent l'organe terminal du robot dans une situation souhaitée, il faut résoudre en « q » l'équation (II.1) pour un vecteur x donné. On parle alors de modèle géométrique inverse qui sera noté [kha 99] [Ama 97]:

$$q = f^{-1}(x) \dots\dots\dots (II.4).$$

Dans le cas où « n=m », il existerait un nombre fini de solutions « q » vérifiant l'équation (II.1) pour « x » donné. Le cas où « n≠m » sera traité au chapitre III.

La difficulté de la résolution inverse réside dans le fait que la fonction est non-linéaire, les théorèmes d'existence et d'unicité font alors défaut. Il n'existe pas de méthode analytique générale pour résoudre le modèle géométrique inverse sauf pour certains cas particuliers de robots présentant des spécificités géométriques (axes parallèles concourants ou perpendiculaires).

Ces méthodes ne s'appliquent donc qu'à des robots manipulateurs simples comme un robot à six degrés de liberté ayant trois articulations rotoïdes d'axes concourants ou possédant trois articulations prismatiques. En dehors de ces cas on utilise des méthodes numériques itératives qui présentent un caractère général mais nécessitent un grand nombre d'opérations et posent de délicats problèmes de convergence. Le chapitre III exposera en détail ces méthodes.

II.4 Modélisation Cinématique du Robot

Le modèle cinématique, appelé encore modèle différentiel permet de calculer la différentielle dX des coordonnées opérationnelles en fonction de la différentielle dθ des coordonnées généralisées. En d'autres termes, il exprime la vitesse de translation ou de rotation de l'organe terminal en fonction des vitesses articulaires [Coi 81] [Vib 87].

Nous avons :

$$\frac{dX}{dt} = \left(\frac{df}{d\theta} \right) \times \left(\frac{d\theta}{dt} \right) \Rightarrow \dot{X} = J \dot{\theta} \dots\dots\dots (II.5).$$

Où J est la matrice Jacobienne du robot manipulateur.

Tout manipulateur peut être considéré comme une chaîne de liaisons connectées par des articulations. Chaque liaison est caractérisée par son repère.

En utilisant les transformations homogènes, on peut décrire la position et l'orientation de chaque repère par rapport à un autre. Denavit et Hartenberg ont établi une transformation qui permet le passage d'une liaison à la prochaine (voir annexe.1).

Ainsi, en utilisant l'équation II.5 on déduit le modèle cinématique:

$$\dot{X} = J \dot{q} = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \end{bmatrix} \dot{q} \dots\dots\dots(II.6).$$

Avec :

$$S_{ij} = \sin(q_i + q_j), C_{ij} = \cos(q_i + q_j), S_{ijk} = \sin(q_i + q_j + q_k) \text{ et } C_{ijk} = \cos(q_i + q_j + q_k)$$

$$J_{11} = -L_1 S_1 - L_2 S_{12} - L_3 S_{123}, \quad J_{12} = -L_1 S_{12} - L_3 S_{123}, \quad J_{13} = -L_3 S_{123}$$

$$J_{21} = L_1 C_1 + L_2 C_{12} + L_3 C_{123}, \quad J_{22} = L_1 C_{12} + L_3 C_{123}, \quad J_{23} = L_3 C_{123}$$

Le domaine de validité du modèle cinématique est borné aux faibles vitesses mais non nulle. Lorsque ces vitesses sont importantes, le modèle cinématique ne permet plus de négliger les effets dynamiques issus de ces grandes vitesses. On fait appel dans ce cas au modèle dynamique.

II.5 Modélisation dynamique :

Après avoir défini la modélisation cinématique du robot et la spécification de ses tâches, on s'intéresse dans cette partie, à l'élaboration du modèle dynamique des robots, dans le but de la commande. La complexité structurelle des systèmes mécaniques articulés nous oblige à choisir une approche systématique pour résoudre ce problème [Vib 87] [Coi 81].

On s'intéresse maintenant aux couples produits par les mouvements du manipulateur. Il s'agit d'établir les équations différentielles qui relient ces couples « $\Gamma_i(t)$ » aux positions, vitesses et accélérations articulaires respectivement $q_i(t)$, $\dot{q}_i(t)$ et $\ddot{q}_i(t)$. L'ensemble de ces équations constitue le modèle dynamique du manipulateur [Khal 99] [Pru 88].

Deux exploitations du modèle sont possibles :

- Résoudre les équations différentielles pour obtenir les mouvements $q_i(t)$ pour des $\Gamma_i(t)$ donnés dans le but d'une simulation du comportement dynamique. On parle alors de modèle dynamique direct (MDD).
- Calculer les efforts $\Gamma_i(t)$ pour les mouvements $q_i(t)$ connus, donnés à l'avance ou mesurés. Cette approche constitue le modèle dynamique inverse (MDI).

Le modèle dynamique des robots peut être obtenu à partir des lois de la mécanique Newtonienne et Lagrangienne. Les formalismes d'Euler-Lagrange et Newton-Euler, nous permettent d'aboutir aux équations du mouvement des robots [Bal 95] [Ser 86].

L'équation de Lagrange-Euler est :

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \left(\frac{\delta L}{\delta q_i} \right) + \left(\frac{\delta E_d}{\delta \dot{q}_i} \right) = \Gamma_i \quad i = 1, 3 \dots \dots \dots (II.7).$$

Où « L » désigne le Lagrangien: $L = E_c - E_p = K - P$.

Tels que :

« E_c » est l'énergie cinétique, « E_p » est l'énergie potentielle, « E_d » est l'énergie de dissipation et « Γ_i » est la force généralisée.

Pour le calcul de l'énergie cinétique, l'expression de la vitesse est:

$$\mathbf{v}_0^i = \frac{d}{dt} (\mathbf{r}_0^i) = \frac{d}{dt} (T_0^i \mathbf{r}_i^i) \dots \dots \dots (II.8).$$

Avec $\mathbf{r}_i^i$ les coordonnées du point i dans le repère R_i .

D'où :

$$\mathbf{v}_0^i = \sum_{j=1}^i \left[\mathbf{u}_{ij} \dot{q}_j \right] \mathbf{r}_i^i \dots \dots \dots (II.9).$$

Tel que :

$$\mathbf{u}_{ij} = \begin{cases} T_0^{j-1} \mathbf{Q}_j T_{j-1}^i & j \leq i \quad \text{et} \quad \mathbf{Q}_j = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ 0 & j \geq i \end{cases}$$

L'énergie cinétique de l'élément i dans la liaison i est :

$$dK_i = \frac{1}{2} tr (\mathbf{V}_i \mathbf{V}_i^T) dm = \frac{1}{2} tr \left[\sum_{p=1}^i \sum_{r=1}^i \mathbf{u}_{ip} \left[\mathbf{r}_i^i dm \mathbf{r}_i^{iT} \right] \mathbf{u}_{ir}^T \dot{q}_p \dot{q}_r \right] \dots \dots \dots (II.10).$$

Donc l'énergie cinétique de la liaison i est :

$$K_i = \int dk_i = \frac{1}{2} tr \left[\sum_{p=1}^i \sum_{r=1}^i \mathbf{u}_{ip} [\mathbf{J}_i] \mathbf{u}_{ir}^T \dot{q}_p \dot{q}_r \right] \dots \dots \dots (II.11).$$

Avec :

$$J_i = \begin{bmatrix} \int x_i^2 dm & \int x_i y_i dm & \int x_i z_i dm & \int x_i dm \\ \int x_i y_i dm & \int y_i^2 dm & \int x_i z_i dm & \int y_i dm \\ \int x_i z_i dm & \int y_i z_i dm & \int z_i^2 dm & \int z_i dm \\ \int x_i dm & \int y_i dm & \int z_i dm & \int dm \end{bmatrix}$$

Et l'énergie cinétique totale du robot est $K = \sum_{i=1}^n K_i + E_{c_a}$

Où:

E_{c_a} est l'énergie cinétique introduite par les actionneurs.

$$E_{c_a} = \frac{1}{2} \sum_{n=1} I_n \dot{q}_n^2 \dots\dots\dots (II.12).$$

I_i est le moment d'inertie de l'actionneur (toutes les articulations sont rotatives).

L'énergie potentielle est:

$$p = \sum_{i=1}^n (-m_i g^T r_0^i) = \sum_{i=1}^n -m_i g^T (T_0^i r_0^i) \dots\dots\dots (II.13)$$

Avec:

$g^T = (g_x \ g_y \ g_z \ 1)$. Dans le repère $R_0(x_0 \ y_0 \ z_0)$, on a $g^T = (0 \ 0 \ -g \ 1)$.

Et l'énergie de dissipation est:

$$E_d = \frac{1}{2} \sum_{i=1}^n b_i \dot{q}_i^2 \dots\dots\dots (II.14).$$

Où b_i est le coefficient de frottement visqueux.

En déduit alors les forces généralisées:

$$\begin{aligned} \Gamma_i = & \sum_{k=1}^n \sum_{p=1}^k \text{tr}[u_{k p} J_k u_{k i}^T] \dot{q}_p - \sum_{k=1}^n m_k g^T u_{k i} r_k^k + \sum_{k=1}^n \sum_{p=1}^k \sum_{r=1}^k \text{tr}[u_{k p} J_k u_{k r}^T] \dot{q}_p \dot{q}_r \\ & + I_i \ddot{q}_i + b_i \dot{q}_i \end{aligned} \dots\dots\dots (II.15).$$

Avec $i=1, n$

$$\text{Où: } u_{ijk} = \begin{cases} T_0^{k-1} Q_k T_{k-1}^{j-1} Q_j T_{j-1}^i & ; \quad k \leq j \leq i. \\ T_0^{j-1} Q_j T_{j-1}^{k-1} Q_k T_{k-1}^i & ; \quad j \leq k \leq i. \dots\dots\dots (II.16). \\ 0 & ; \quad j \leq i \leq k. \end{cases}$$

On remarque que si on utilise l'équation (II.13) les calculs sont longs. Il est donc préférable de calculer d'abord le Lagrangien ensuite procéder par dérivations en utilisant l'équation (II.9) [Gue 95].

$$\text{Sachant que: } u_{ij} = \begin{cases} T_0^{j-1} Q_j T_{j-1}^i & j \leq i \\ 0 & j \geq i \end{cases}$$

$$\text{On a alors: } \begin{cases} u_{11} = T_0^0 Q_1 T_0^1 \\ u_{12} = 0 \\ u_{13} = 0 \end{cases} \quad \begin{cases} u_{21} = T_0^0 Q_1 T_0^2 \\ u_{22} = T_0^1 Q_2 T_1^2 \\ u_{23} = 0 \end{cases} \quad \begin{cases} u_{31} = T_0^0 Q_1 T_0^3 \\ u_{32} = T_0^1 Q_2 T_1^3 \\ u_{33} = T_0^2 Q_3 T_2^3 \end{cases}$$

Les expressions des u_{ij} sont exposées en annexe 1.

$$\text{On a: } \begin{cases} L = \frac{1}{2} \sum_{i=1}^2 \sum_{j=1}^i \sum_{k=1}^k \text{tr}[u_{ij} J_j u_{ki}^T] \dot{q}_j \dot{q}_i + \sum_{i=1}^2 m_i g(T_0^i r_i^i) \\ \Gamma_i = \frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \left(\frac{\delta L}{\delta q_i} \right) + \left(\frac{\delta E_d}{\delta \dot{q}_i} \right) \end{cases} \dots\dots\dots (II.17).$$

$$\text{Donc: } \begin{cases} L_1 = \frac{1}{2} \sum_{j=1}^1 \text{tr}(u_{1j} J_1 u_{11}^T) \dot{q}_j \dot{q}_1 & \text{pour } i = 1 \\ L_2 = \frac{1}{2} \sum_{j=1}^2 \text{tr}(u_{2j} J_2 u_{21}^T) \dot{q}_1 \dot{q}_j + \text{tr}(u_{2j} J_2 u_{22}^T) \dot{q}_2 \dot{q}_j & \text{pour } i = 2 \\ L_3 = \frac{1}{2} \sum_{j=1}^3 \text{tr}(u_{3j} J_3 u_{31}^T) \dot{q}_1 \dot{q}_j + \text{tr}(u_{3j} J_3 u_{32}^T) \dot{q}_2 \dot{q}_j + \text{tr}(u_{3j} J_3 u_{33}^T) \dot{q}_3 \dot{q}_j & \text{pour } i = 3 \end{cases}$$

Le 2^{ème} terme du lagrangien (énergie potentielle) devient [Kal 95]:

$$J_1 = \begin{bmatrix} \frac{1}{3} m_1 l^2 & 0 & 0 & -\frac{1}{2} m_1 l \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{2} m_1 l & 0 & 0 & m_1 \end{bmatrix};$$

En posant:

$$M_{ij}(q) = \sum_{k=\max(i,j)}^n \text{trace}(u_{kj} J_k u_{ki}^T), \quad i=1,n, \quad j=1,n$$

$$G_i(q) = - \sum_{j=1}^n m_j g^T u_{ji} r_i^i$$

$$C_{ij}(q, \dot{q}) = \sum_{l=\max(i,j,k)}^n \text{trace}(u_{kl} J_k u_{li}^T), \quad i=1,n, \quad j=1,n \quad k=1,n$$

L'équation (II.7) devient alors

$$\Gamma_i = \sum_{j=1}^n M_{ij}(q) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n C_{ijk}(q) \dot{q}_j \dot{q}_k + G_i(q) \dots \dots \dots (II.18).$$

L'équation (II .18) devient:

$$\Gamma = M(q) \cdot \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) \dots \dots \dots (II.19).$$

N.B : L'équation (II.19) est l'équation différentielle du modèle dynamique du robot manipulateur redondant

La mise sous forme matricielle du modèle du robot redondant donne le système suivant:

$$\begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{11} & c_{12} & c_{13} \\ c_{11} & c_{12} & c_{13} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix}$$

Avec:

M la matrice d'Inertie, C la matrice des forces centrifuges et coriolis et G la matrice Gravitationnelle.

II.6 Simulations du modèle dynamique du bras manipulateur redondant en boucle ouverte

A. Modèle dynamique à vide (sans charge)

Il est nécessaire de simuler le modèle de connaissances établi par formalisme de Lagrange-Euler. La mise sous forme d'état des équations différentielles non-linéaire trouvée,

nous permet d'utiliser la méthode de Runge-kutta (RK) d'ordre quatre sous les conditions initiales nulles.

Le vecteur d'état $X^T = [X_1 \ X_2 \ X_3 \ X_4 \ X_5 \ X_6] = \begin{bmatrix} q_1 & q_2 & q_3 & \dot{q}_1 & \dot{q}_2 & \dot{q}_3 \end{bmatrix}$

Les réponses temporelles du robot redondant en position et en vitesse pour une entrée échelon sont consignées sur la figure 4.

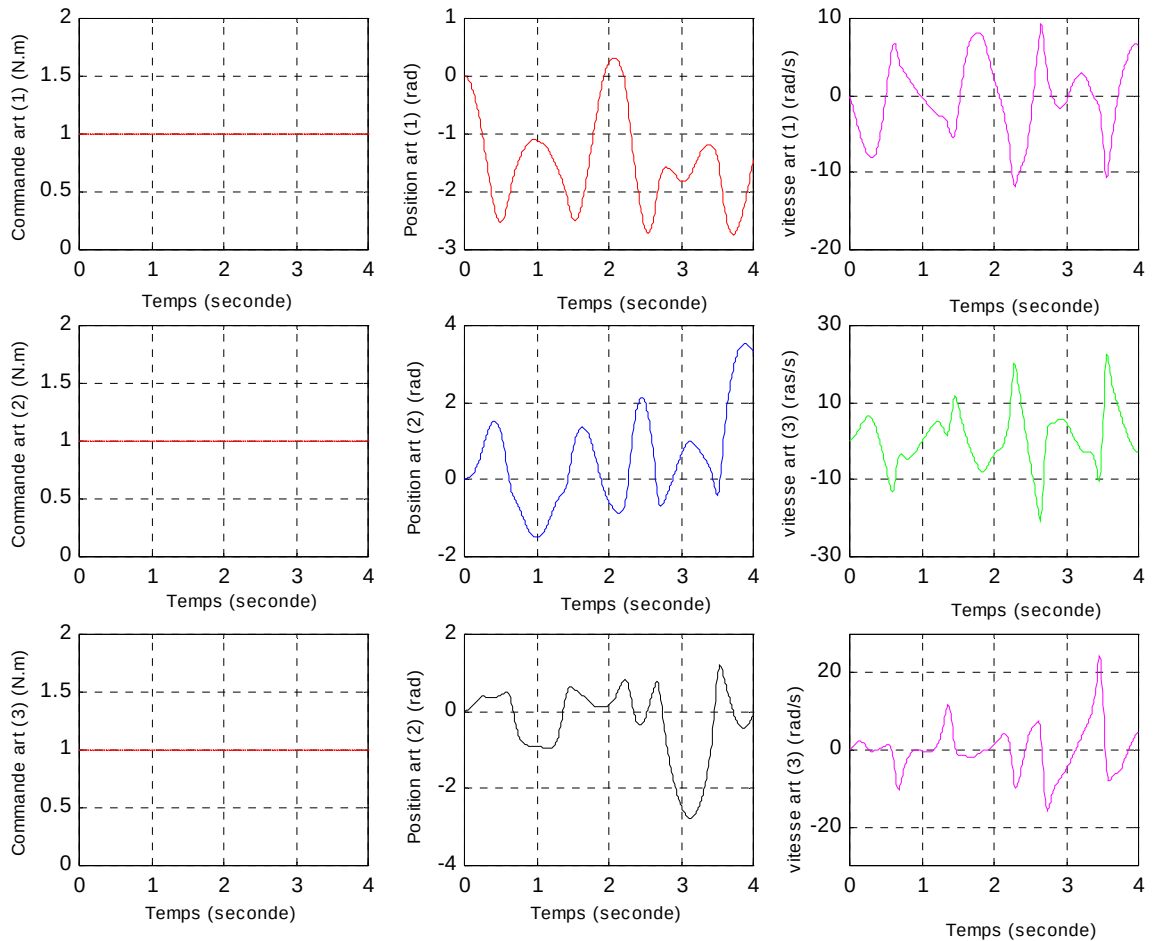


Figure 4. Réponse indicielle du robot redondant à vide en boucle ouverte
(Position et vitesse).

La figure 4 illustre la réponse indicielle en boucle ouverte du système sans charge. Les résultats de simulation obtenus montrent l'instabilité du système. Ceci s'explique par l'existence de couplage entre les différentes articulations, la nature fortement non linéaire du modèle dynamique du bras manipulateur considéré et la présence d'une action intégrale vu que l'équation dynamique est une équation différentielle du deuxième ordre. Le but essentiel de cette simulation est de vérifier la validité de notre modèle.

L'intégration de l'équation d'état du système a été effectuée avec l'algorithme de Runge-Kutta d'ordre 4, cette méthode présente l'avantage d'être stable et précise.

Les simulations de la réponse du système en boucle ouverte ont été réalisées par programmation en langage MATLAB. Aucune fonction préprogrammée de MATLAB n'est utilisée dans nos simulations ainsi nous avons programmé l'algorithme de résolution Runge-Kutta d'ordre 4 pour la résolution de l'équation différentielle du second ordre. La période d'échantillonnage utilisée dans nos simulations égale à 0.01 secondes.

B. Modèle dynamique avec charge

Dans la pratique lors de l'exécution d'une tâche spécifique, le robot peut être amené à porter des charges, et ces charges peuvent varier d'un moment à l'autre, pour analyser leurs effets, on doit les introduire dans l'équation dynamique [Ser 89b].

Considérons une masse ponctuelle « m » portée par l'organe terminal. Si on veut la déplacer avec une accélération « $\ddot{x}$ » dans un plan où la force de gravitation intervient, on doit donc exercer sur l'organe terminal une force supplémentaire « F_s » avec :

$$F_s = m \left(\ddot{x} + g' \right) \dots \dots \dots (II.20).$$

Comme physiquement on ne peut pas appliquer directement cette force à l'organe terminal, on calcule alors une force généralisée équivalente.

$$\Gamma_s = J'(q).F_s \dots \dots \dots (II.21).$$

Avec: $J'(q)$ est la transposée de la matrice Jacobienne.

Pour exprimer l'accélération cartésienne « $\ddot{x}$ » en fonction des variables généralisées [tidjani-belazerag], on calcule la dérivée deuxième de l'équation du modèle géométrique direct $x = F(q)$, on obtient :

$$\ddot{x} = \dot{J}(q, \dot{q}).\dot{q} + J(q).\ddot{q} \dots \dots \dots (II.22).$$

$$\text{Où: } \dot{J} = \frac{dJ}{dt}$$

En substituant des équations (II.20), (II.21) et en tenant compte de l'équation (II.22); il résulte:

$$\Gamma_s = m . J'(q) . \left(\dot{J}(q, \dot{q}).\dot{q} + J(q).\ddot{q} + g' \right) \dots \dots \dots (II.23).$$

On introduit le couple supplémentaire dans le modèle de l'équation (II.19) on obtient:

$$\Gamma = M(q).\ddot{q} + C(q, \dot{q}) + G(q) + \Gamma_s \Rightarrow \Gamma = M'(q).\ddot{q} + C'(q, \dot{q}) + G'(q) \quad \dots\dots(II.24).$$

$$\text{avec: } M'(q) = M(q) + m \cdot J'(q) \cdot J(q).$$

$$C'(q, \dot{q}) = C(q, \dot{q}) + m \cdot J'(q) \cdot \dot{J}(q, \dot{q}).$$

$$G'(q) = G(q) + m \cdot J'(q) \cdot g'$$

Les résultats de simulation en boucle ouverte du modèle dynamique portant une charge de 1.5 Kg sont illustrés dans la figure 5.

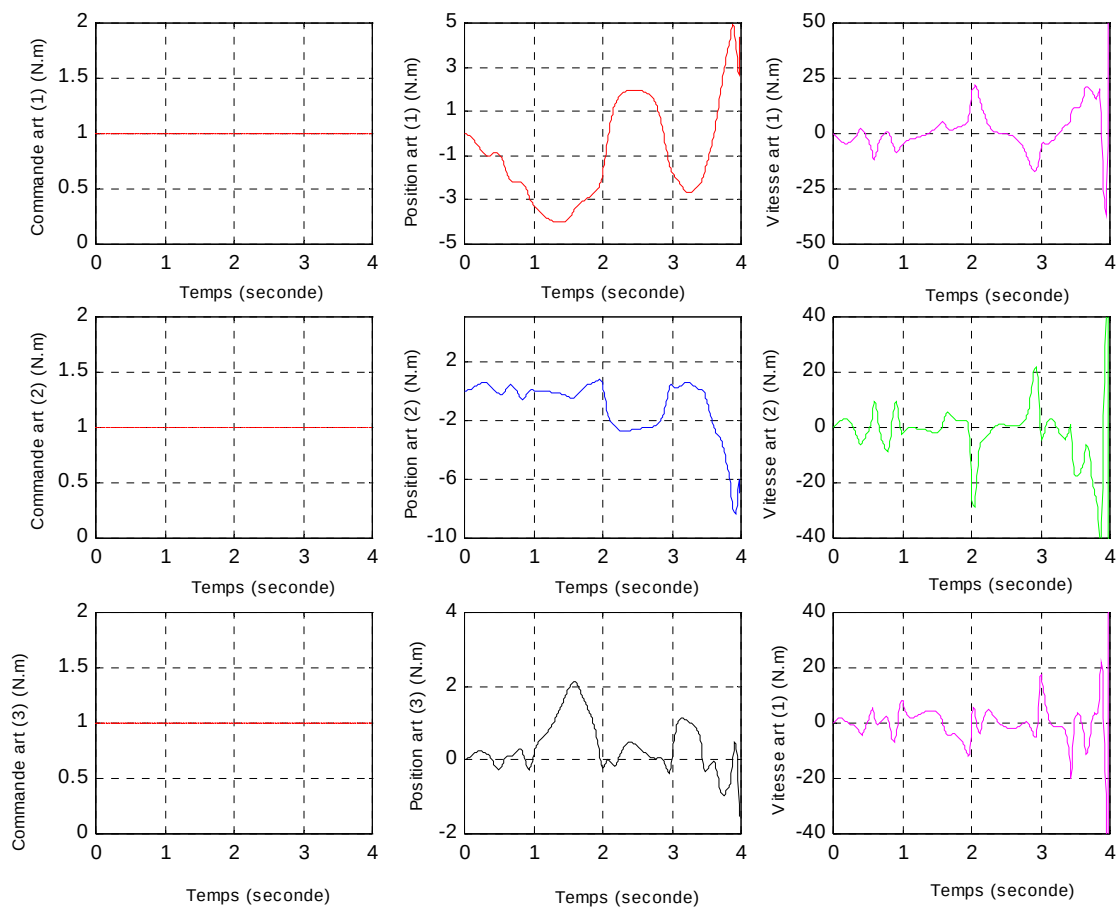


Figure 5. Réponse indicielle du robot redondant avec charge.

La figure 5 illustre la réponse indicielle en boucle ouverte du système avec charge. Les courbes représentées montrent l'évolution temporelle des positions et vitesses articulaires.

Les résultats obtenus montrent initialement de faibles vitesses sur les trois articulations. Puis ces vitesses augmentent au cours du temps. Ceci s'explique par le fait que le couple doit

compenser l'effet de la charge supplémentaire et l'inertie des bras du robot manipulateur lors de démarrage.

II.7 Générateur de trajectoire

La dynamique du robot exige d'imposer des trajectoires réalisables, la continuité des positions, vitesses et accélérations offre au robot la possibilité de poursuivre la trajectoire avec des commandes réalisables physiquement. La décomposition de la tâche en plusieurs points intermédiaires nécessite donc une continuité du premier et du second ordre.

Le fait d'imposer une position finale constante par exemple; nécessite l'utilisation d'un polynôme du troisième degré permettant ainsi une continuité en position et en vitesse. Cette tâche se traduit par le passage d'un état d'équilibre à un autre état d'équilibre. Ces deux états définiront les conditions aux limites du polynôme d'interpolation [Gue 95]:

$$\begin{aligned}
 & p(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \\
 & \begin{cases} p(0) = \theta_0 & \dot{p}(0) = \dot{\theta}_0 \\ p(t_f) = \theta_f & \dot{p}(t_f) = \dot{\theta}_f \end{cases} \dots\dots\dots (II.25).
 \end{aligned}$$

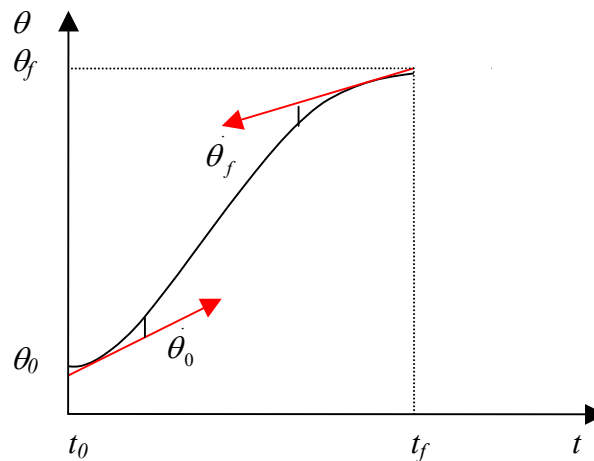


Figure 6. Trajectoire cycloïde

Généralement $\dot{\theta}_0 = \dot{\theta}_f = 0$

a partir des conditions aux limites (équation II.25), on calcule les différents coefficients du polynôme $p(t)$, on aura alors [Gue 95] [Guo 00]:

$$\begin{cases} a_0 = \theta_0 \\ a_1 = \dot{\theta}_0 \\ a_2 = \frac{3}{t_f^3}(\theta_f - \theta_0) - \frac{2}{t_f}\dot{\theta}_f \\ a_2 = \frac{2}{t_f^3}(\theta_f - \theta_0) + \frac{1}{t_f^2}(\dot{\theta}_f + \dot{\theta}_0) \end{cases} \dots\dots\dots(II.26).$$

On peut aussi assurer cette continuité par une cycloïde, on dira alors que la trajectoire est cycloïde, elle sera donnée par l'équation [Ser 89a]:

$$q_d(t) = \begin{cases} q_d(t) + \frac{D}{2\pi} \left[2\pi \frac{t}{t_f} - \sin\left(2\pi \frac{t}{t_f}\right) \right] & \text{pour } 0 \leq t \leq t_f \\ q_d(t) & \text{pour } t > t_f \end{cases} \dots\dots\dots(II.27).$$

Où $D = q_d(t_f) - q_d(0)$ est le déplacement, t_f est l'instant final du mouvement et q_d est la trajectoire de référence.

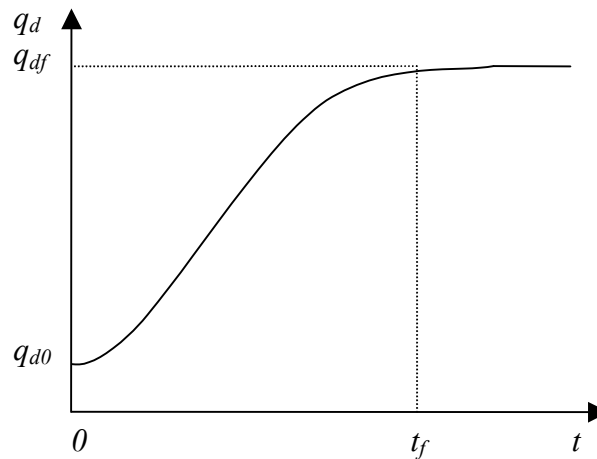


Figure 7. Trajectoire cycloïde

Pour le robot manipulateur, il existe une trajectoire cycloïde test proposée par LEAHVY. Les différentes articulations (art1, art2, art3) se déplacent respectivement de la position $(-50^\circ, -135^\circ, 135^\circ)$ à la position $(45^\circ, -85^\circ, 30^\circ)$ en un temps de mouvement égal à 1.5 secondes. Cette trajectoire est choisie car elle excite toute la dynamique du bras manipulateur.

Si maintenant la consigne n'est pas constante, pour une trajectoire donnée (trajectoire de l'effecteur), on peut aisément déterminer les déplacements et les vitesses dus à chaque articulation $(\theta_i, \dot{\theta}_i)$, ceci en utilisant les équations du modèle géométrique et cinématique inverse des robots.

En effet, à chaque point de coordonnées cartésiennes, on détermine les coordonnées généralisées correspondantes. Suivant la trajectoire de l'effecteur, on trouve une relation donnant les coordonnées généralisées (q_i) qui décrivent cette même trajectoire.

II.8 Conclusion

Dans ce chapitre, nous avons exposé la notion de modélisation, car la mise en œuvre de structures de commande performantes nécessite, généralement, la connaissance des modèles du robot manipulateur. De plus, la validation de toute structure de commande d'un système robotisé doit toujours être réalisée dans une première étape en simulation.

Nous avons présenté dans cette première partie, les modèles géométrique et cinématique permettant d'obtenir les transformations de coordonnées (position et vitesse) entre l'espace articulaire et cartésien du robot redondant à trois degrés de liberté rotoïdes. Afin de tenir compte des différents phénomènes non linéaires engendrés par la dynamique du robot à vitesse élevée, le modèle dynamique est aussi établi.

Le comportement dynamique du robot manipulateur exige une trajectoire spécifique (continue en position, en vitesse et en accélération), permettant ainsi le contrôle du robot avec des commandes physiquement réalisables.

L'implémentation de structure de commande sur un système robotisé, nécessite généralement la conversion des coordonnées cartésiennes en coordonnées articulaires. Cette conversion est réalisée par le modèle géométrique inverse. Dans le cas d'un bras manipulateur redondant le calcul de ce modèle est fastidieux. En effet, le nombre d'actionneurs étant supérieur au nombre de degrés de liberté, il existe une infinité de solutions. Il n'existe pas de méthode analytique générale pour résoudre le modèle géométrique inverse dans le cas de notre système. Ceci dit, nous présenterons dans le chapitre suivant, un algorithme permettant d'établir le modèle géométrique inverse du bras manipulateur redondant considéré.

INVERSION DU MODELE GEOMETRIQUE

III.1 Introduction

Lors de la résolution du modèle géométrique inverse d'un bras manipulateur redondant, on fait généralement appel à des procédures numériques car les solutions inverses n'ont pas, à priori, d'expression analytique. En effet, à une situation donnée de l'organe terminal correspond une infinité de configurations possibles qui permettent à l'effecteur de réaliser la tâche opérationnelle désirée.

Il existe dans la littérature différents algorithmes d'optimisation utilisés pour la résolution du modèle géométrique inverse.

Nous nous proposons dans ce chapitre de présenter certains de ces algorithmes les plus couramment utilisés. Une analyse des caractéristiques de ces méthodes, nous permet de choisir parmi elles. Les algorithmes choisis seront, par la suite, appliqués dans le cas d'un bras manipulateur redondant à trois liaisons rotoïdes.

III.2 Positionnement du problème

Nous rappelons d'abord la problématique à laquelle nous sommes confrontés. Connaissant la position du point terminal du robot X_d (vecteur de dimension $m \times 1$), on désire calculer les positions articulaires correspondantes de ce bras q_d (vecteur de dimension $n \times 1$). Il s'agit donc de déterminer q_d tel que:

$$F(q_d) = X_d \dots\dots\dots(III.1)$$

La fonction F désigne une relation directe générale qui permet de passer des coordonnées articulaires aux coordonnées opérationnelles selon la structure mécanique du robot étudié. La fonction F correspond donc à un modèle bien particulier.

Dans la figure 1, nous présentons le bras manipulateurs redondant utilisé dans notre présente étude. Ce bras est constitué de trois axes rigides articulés par trois liaisons rotoïdes de coordonnées articulaires (q_1, q_2 et q_3) (d'où $n=3$). Les mouvements du bras sont réalisés dans le plan (x, y). La position de l'effecteur est donnée par ces coordonnées (X, Y) (d'où $m=2$).

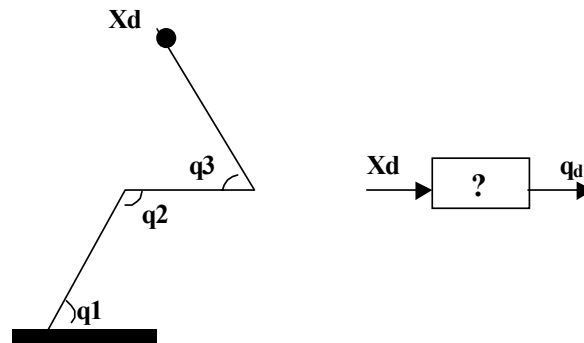


Figure 1. Transformation de coordonnées d'un bras de robot.

Trouver q_d en fonction de X_d revient donc à inverser le système d'équation (III.1) pour obtenir:

$$q_d = F^{-1}(X_d) \dots\dots\dots (III.2)$$

Certes, notre problématique de l'inversion de F s'applique dans le cas de notre étude à des structures mécaniques de robots. Mais, résoudre cette inversion, c'est aussi apporter une solution à tout système d'équation F représentant un phénomène physique, d'où la solution générale apportée à toute application. Les phénomènes physiques modélisés sont si complexes que le modèle géométrique direct est le plus souvent une fonction fortement non linéaire, et même dans le cas particulier des bras de robot, il n'existe pratiquement jamais d'expression analytique de son inverse.

Pour contrer ces difficultés, on fait appel à des procédures numériques. Leur principe est de faire évoluer itérativement les coordonnées articulaires q selon un algorithme donné, jusqu'à ce que q converge vers les coordonnées articulaires désirées q_d tel que:

$$F(q) = F(q_d) = X_d$$

Deux méthodes permettent l'inversion du modèle géométrique « F » [Ama 97]:

- La première méthode réalise une identification du modèle (notée par IP).
- La seconde méthode est dite boucle fermée BF. C'est une boucle dynamique d'asservissement de la variable opérationnelle x dont la consigne est la variable désirée x_d .

A. Méthode utilisant une identification du modèle

Son principe est une identification hors ligne des paramètres du modèle inverse d'une fonction $F(q)$. Dans ce cas, l'erreur cartésienne e permet de modifier les paramètres du modèle inverse pour obtenir sa convergence vers F^{-1} (figure 2) [Ben 02].

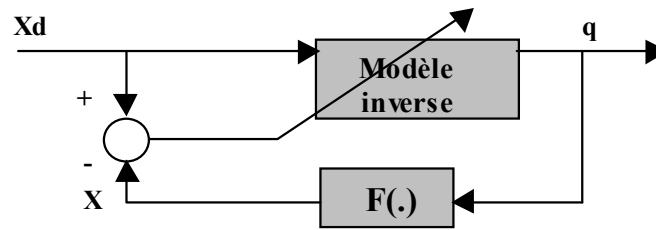


Figure 2. Identification du modèle inverse.

B. Méthode de la boucle fermée

La majorité des méthodes utilisées pour inverser le modèle géométrique sont basées en fait sur le même principe, caractérisé par le schéma de la figure 3, c'est à dire celui d'un asservissement de la variable x . Toutefois, dans les approches de types algorithmiques, ce schéma n'apparaît pas et la solution est introduite sous la forme d'une équation de récurrence [Dro 97]:

$$q_{c+1} = q_c + Me_c \dots \dots \dots (III.3)$$

Où e_c est l'erreur entre x_d et x à l'itération c et Me_c la modification apportée à la position courante q_c .

A contrario, dans le type « asservissement », l'aspect algorithmique n'apparaît pas et l'interprétation devient celle du robot fictif (figure 3).

Nous proposons une interprétation commune de toutes ces méthodes permettant de mieux comparer les avantages et les inconvénients de chacune et de choisir une méthode. Toutes les solutions que nous examinons ici consistent à réaliser un asservissement de la variable opérationnelle x .

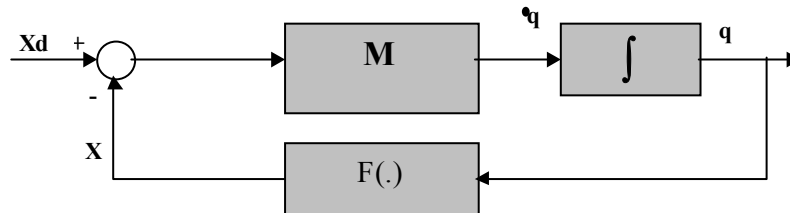


Figure 3. Schéma de base pour l'inversion géométrique (boucle fermée).

Le processus est alors un système statique non linéaire caractérisé par la fonction F . Un intégrateur (continu ou discret) est placé dans la boucle afin d'assurer une erreur statique nulle, c'est à dire d'avoir à la convergence: $x = x_d$ et donc q tel que $F(q) = x_d$. La boucle est alors rendue dynamique et des problèmes classiques de stabilité et de temps de convergence apparaissent.

La période d'échantillonnage de cette boucle est en fait le temps de calcul nécessaire pour toutes les opérations de la chaîne directe. On notera par c l'indice du temps (ou de l'itération correspondante). Le temps de convergence doit naturellement être très inférieur à la période de commande utilisée pour l'asservissement du robot et dont le temps est indicé par k .

Les méthodes analysées maintenant se différencient donc par le choix de la fonction de transfert du bloc M qui joue le rôle de correcteur dans l'asservissement, et selon le cas discret ou continu de leurs applications.

B.3 Etude de stabilité :

D'après la figure 3, l'erreur cartésienne est [Dro 97] [Ama 97]:

$$e = X_d - X \dots\dots\dots (III.4)$$

Sa dérivée est donc:

$$\dot{e} = \dot{X}_d - \dot{X} \dots\dots\dots (III.5)$$

Avec « $\dot{X} = J \dot{q}$ » Où J est la matrice jacobienne.

Si on choisit la fonction de Lyapunov de type :

$$V = \frac{1}{2} e^T K e \dots\dots\dots (III.6)$$

Où K est une matrice de gains

La dérivée de cette fonction donne :

$$\dot{V} = e^T K \dot{e} = -e^T K J \dot{q} \dots\dots\dots (III.7)$$

En posant, $\dot{q} = M e$ l'équation précédente devient :

$$\dot{V} = -e^T [K J M] e \dots\dots\dots (III.8)$$

La stabilité du système représenté sur la figure 3 est garantie si $\dot{V} < 0 \quad \forall q$, c'est à dire si la matrice $P = K J M$ est strictement définie positive. Une étude de stabilité dans le cas discret conduit à la même condition.

Nous allons décrire les différentes méthodes correspondantes aux choix de M garantissant la stabilité.

B.4 Les différents choix de la matrice M :

Le problème de résolution inverse revient à annuler l'erreur e en un nombre fini c d'itérations de telle sorte que q converge vers q_c , solution de [Ama 97] :

$$e_c = X_d - F(q_c) = 0 \dots\dots\dots (III.9)$$

Pour cela, on distingue deux grands types de méthodes : Les méthodes algorithmiques directes et les méthodes basées sur les approximations linéaires.

B.2.3 *Méthodes algorithmiques directes*

Il s'agit de minimiser les critères du type :

$$A(q_c) = \frac{1}{2} e_c^T e_c = \frac{1}{2} (X_d^T X_d - 2X_d^T F(q_c) + F^T(q_c) F(q_c)) \dots\dots\dots (III.10)$$

On peut appliquer un des algorithmes itératifs existants qui doit converger vers le minimum du critère. Parmi ces méthodes, citons :

- **Méthode du gradient**

Le gradient de $A(q_c)$ s'écrit :

$$\alpha \frac{\partial A_c}{\partial q_c} = \left(\frac{\partial e_c^T}{\partial q_c} \right) e_c = -(J^T) e_c \dots\dots\dots (III.11)$$

Un choix tout naturel pour la direction de recherche du minimum ($e_c \rightarrow 0$) est de prendre $\Delta q_c = -\frac{\partial A_c}{\partial q_c}$. La méthode de gradient utilise donc Δq_c dans un processus itératif (figure 4).

$$q_{c+1} = q_c + \alpha (J^T)_c e_c \dots\dots\dots (III.12).$$

« α » est le pas de gradient dont la valeur conditionne le bon fonctionnement de l'algorithme. Le choix de ce paramètre est très délicat.

Si on remplace M par αJ^T dans l'équation (III.8) avec $K=I$ (I matrice identité), la matrice P devient: $P = \alpha J^T J$ est définie strictement positive, si $J^T J$ est définie positive. Cette dernière condition n'est pas toujours vérifiée, le cas $J^T J \geq 0$ peut se présenter.

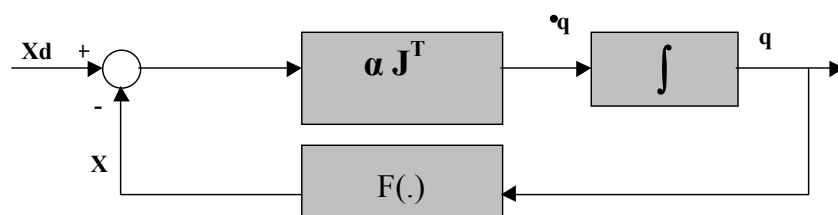


Figure 4. Solution gradient

- **Méthode du second ordre**

Si on introduit un changement de base ($q = Pq'$) sur un processus itératif de type :

$$q_{c+1} = q_c + \alpha \frac{\partial A_c}{\partial q_c} e_c \dots\dots\dots (III.13)$$

On aura donc:

$$Pq'_{c+1} = q_{c+1} = Pq'_c + \alpha P \frac{\partial A_c}{\partial q'_c} e_c \dots \dots \dots (III.14)$$

$$\text{Or } \frac{\partial A_c}{\partial q'_c} = P^T \frac{\partial A_c}{\partial q_c}$$

D'où :

$$q_{c+1} = q_c - \alpha P P^T \frac{\partial A_c}{\partial q_c} \dots \dots \dots (III.15)$$

On pose $B = \alpha P P^T$.

$$q_{c+1} = q_c + B (J^T)_c e_c \dots \dots \dots (III.16)$$

« B » une matrice qui doit être positive. Cette méthode appelée méthode de Davidon-Fletcher-Powell figure 5. Elle utilise le transposée de la matrice jacobienne, elle relève, de la méthode algorithmique du gradient. En remplaçant M par $B J^T$ Dans l'équation (III.8) où $K=I$ (matrice d'identité), la matrice P devient: $P = K I J B J^T$. P est donc définie positive, si $J B J^T$ est positive.

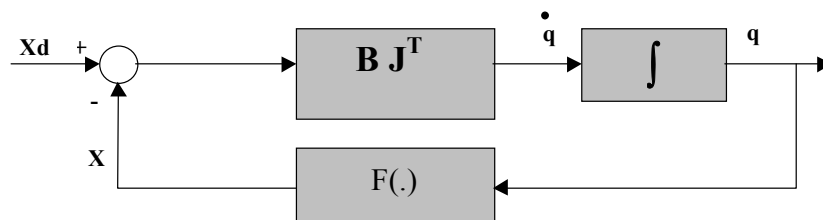


Figure 5. Solution du second ordre.

- Méthode de Newton

Le calcul de Hessian du critère $A(q_c)$ donne :

$$\frac{\partial^2 A_c}{\partial q_c^2} = \frac{\partial}{\partial q_c} \left((J^T)_c e_c \right) = (J^T)_c (J)_c - S_c \dots \dots \dots (III.17)$$

$$\text{Avec } S_c = \sum_{i=1}^m e_i(c) \nabla^2 A_i(c).$$

Le modèle quadratique de $A(q)$ autour de q_c est :

$$Q_c = \frac{1}{2} e_c^T e_c - \partial q_c^T (J^T)_c e_c + \frac{1}{2} \partial q_c^T [(J^T)_c (J)_c + S_c] \delta q_c \dots\dots\dots (III.18)$$

Le problème de trouver une solution (convergence vers la valeur désirée) revient à minimiser le modèle quadratique Q_c de $A(q)$ [Lor 88] [Sic 87]. Par la méthode Newton, on obtient :

$$q_{c+1} = q_c + ((J^T)_c (J)_c + S_c)^{-1} (J^T)_c e_c \dots\dots\dots (III.19)$$

Donc la matrice M est remplacée par la matrice $[J J^T + S]^{-1} J^T$ correspond à la matrice M (figure 6). La stabilité de cette méthode est vérifiée si la matrice $P = J [J J^T + S]^{-1} J^T$ est définie strictement positive.

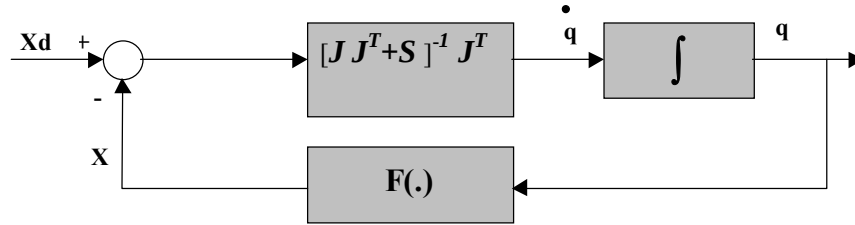


Figure 6. Solution de Newton.

B.2.4 Méthodes basées sur une approximation linéaire

Une approximation linéaire de l'équation (III.9) donne :

$$e_{c+1} = e_c - (J)_c \delta q_c \dots\dots\dots (III.20).$$

En effet, si X_{c+1} sont les coordonnées cartésiennes à l'itération (c+1) :

$$X_{c+1} = F(q_{c+1} = q_c + \delta q_c) = F(q_c) + (J)_c \delta q_c \dots\dots\dots (III.21).$$

On a :

$$X_{c+1} = X_c + (J)_c \delta q_c \dots\dots\dots (III.22).$$

L'idée consiste alors à déterminer δq_c telle que $e_{c+1} = 0$. Selon l'existence de J^l , on distingue deux cas:

- Premier cas :

A chaque itération. La valeur de δq_c est calculée en utilisant le modèle cinématique inverse. Le vecteur q_c est l'état articulaire courant [Che 88].

$$\delta q_c = (J^{-1})_c e_c \dots\dots\dots (III.23).$$

δq_c est utilisé dans un processus itératif $q_{c+1} = q_c + \delta q_c$ jusqu'à la convergence. La matrice J^{-1} correspond à la solution M et doit garantir la stabilité du système.

Si on remplace M par J^{-1} dans l'équation (III.8), la matrice devient : $P = J J^{-1} = I$, ($K = I$) (I matrice d'identité), P est bien définie positive et le système est stable si J^{-1} existe. Dans le cas où J n'est pas carrée, on remplace J^{-1} par la pseudo inverse J^+ .

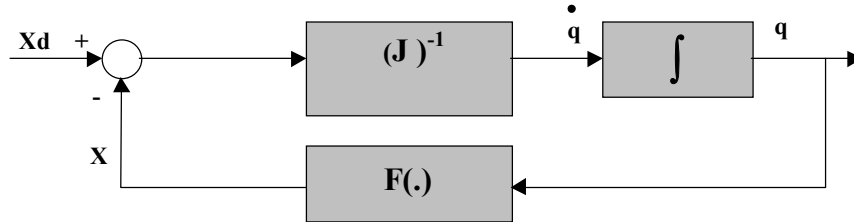


Figure 7. Solution basée sur le modèle différentiel inverse.

- Deuxième cas :

Il consiste à minimiser le critère suivant :

$$A_c = \|e_{c+1}\|^2 = \|e_c - (J)_c(q - q_c)\|^2 \dots\dots\dots(III.24).$$

Cette minimisation peut être faite avec ou sans contrainte :

a) sans contrainte

Le Lagrangien du ce système s'écrit :

$$L = \frac{1}{2} [e_c^T e_c - 2 \delta q_c^T (J^T)_c e_c + \partial q_c^T (J^T)_c (J)_c \delta q] \dots\dots\dots(III.25).$$

La condition de stationnarité de L s'écrit :

$$\frac{\partial L}{\partial \delta q_c} = 0 = -(J^T)_c e_c + ((J^T)_c (J)_c) q_c \dots\dots\dots(III.26).$$

On aura donc:

$$\delta q_c = ((J^T)_c (J)_c)^{-1} (J^T)_c e_c \dots\dots\dots(III.27).$$

Le processus itératif correspondant à cette méthode s'écrit :

$$q_{c+1} = q_c + ((J^T)_c (J)_c)^{-1} (J^T)_c e_c \dots\dots\dots(III.28).$$

La matrice $M = (J^T J)^{-1} (J^T)$ n'est rien d'autre que la pseudo-inverse J^+ . On retrouve le cas précédent.

b) avec contrainte

On peut appliquer la méthode classique du gradient. Mais pour éviter le problème du choix du pas du gradient, on utilise la méthode de Levenberg-Marquardt. Elle minimise le même critère avec une limitation du pas à une valeur Δ :

$$\begin{cases} \min A_c \\ \text{sous la condition } \|q - q_c\| \leq \Delta \end{cases}$$

Où q_c est l'état articulaire courant. Le Lagrangien de ce système s'écrit :

$$L = \frac{1}{2} [e_c^T e_c - 2 \delta q_c^T (J^T)_c e_c + \delta q_c^T (J^T)_c (J)_c \delta q] + \lambda [\delta q_c^T \delta q_c - \Delta^2] \dots \dots (III.29).$$

La condition de stationnarité de L s'écrit :

$$\frac{\partial L}{\partial \delta q_c} = 0 = -(J^T)_c e_c + ((J^T)_c (J)_c + \lambda I) \delta q_c \dots \dots \dots (III.30).$$

On aura donc:

$$\delta q_c = ((J^T)_c (J)_c + \lambda I)^{-1} (J^T)_c e_c \dots \dots \dots (III.31).$$

Le processus itératif correspondant à cette méthode s'écrit :

$$q_{c+1} = q_c + ((J^T)_c (J)_c + \lambda I)^{-1} (J^T)_c e_c \dots \dots \dots (III.32).$$

La stabilité de ce système est assurée si la matrice $P = ((J^T)_c (J)_c + \lambda I)^{-1} (J^T)_c J$ est définie positive.

On distingue deux cas particuliers suivant la valeur λ

- Si λ tend vers 0, M devient égale à $((J^T)_c(J)_c)^{-1}(J^T)$, qui n'est rien d'autre que la pseudo-inverse de la matrice J (cas sans contrainte). La matrice P devient: $P = \left(J^T J \right)^{-1} J^T J$. P est bien positive et le système est stable si $(J^T J)^{-1}$ existe.
- Si λ tend vers ∞ , M devient égale à $\frac{1}{\lambda} J^T$, qui n'est rien d'autre que la matrice utilisée dans la méthode basée sur la transposée de la matrice Jacobienne. La matrice P devient $P = J^T J$. P est positive si $J^T J$ est positive. Cette dernière condition n'est pas toujours vérifiée, le cas $J^T J \geq 0$ peut se présenter.

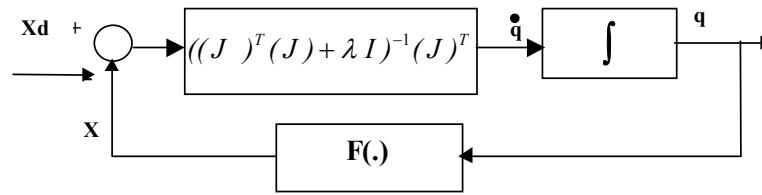


Figure 8. Solution de Levenberg-Marquardt.

III.3 Commentaires et comparaison

Dans ce qui précède, on distingue deux classes: les méthodes algorithmiques directes et les méthodes basées sur une approximation linéaire.

Le principe de la première classe revient à minimiser un critère sur l'erreur opérationnelle. Pour cela, on utilise les algorithmes classiques tels que : algorithme du gradient, l'algorithme du second ordre, l'algorithme de Newton etc. Leur inconvénient majeur est la difficulté de la détermination du pas d'apprentissage « α ».

Le principe de la deuxième classe revient tout d'abord à faire une approximation linéaire du modèle inverse. Selon les cas présentés on cite la méthode basée sur le modèle différentiel inverse et la méthode de Levenberg-Marquardt. L'approximation linéaire du premier ordre ne conduit pas toujours à la solution désirée. Des divergences ou des oscillations peuvent se produire, d'où la nécessité de passer dans ce cas au modèle quadratique du second ordre pour assurer la convergence.

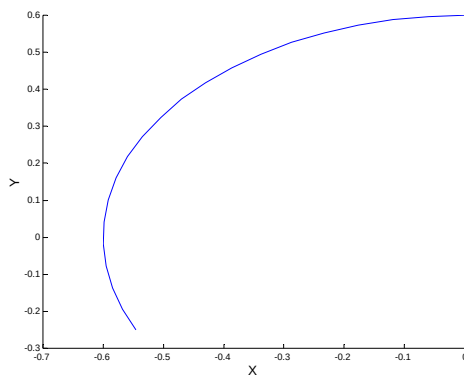
Dans notre présente étude, nous avons utilisé, dans un premier temps, la méthode de Levenberg-Marquardt, compte tenu des avantages qu'elle présente. Cette méthode est une approximation du second ordre d'où elle élimine le problème de la divergence ou des oscillations. Ainsi elle n'utilise pas le modèle cinématique inverse, de plus elle satisfait le critère de stabilité de Lyapunov. Pour ces raisons, nous l'avons adopté dans nos simulations.

III.4 Application de la méthode de levenberg-Marquardt dans le cas d'un bras manipulateur redondant

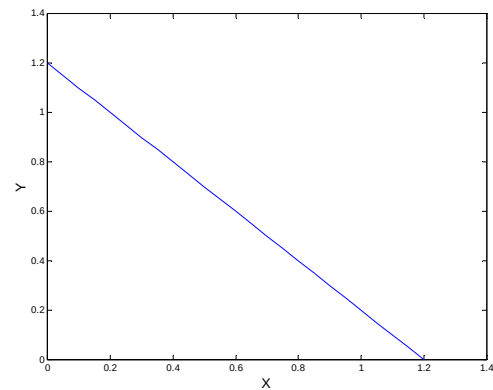
Dans cette partie nous nous intéressons au comportement de l'algorithme Levenberg-Marquardt sur toute une trajectoire. Ceci dit nous présentons les résultats de simulation obtenus lors de l'implémentation de cette méthode.

Afin d'analyser ses performances trois types de trajectoires sont considérés: une trajectoire de type arc de cercle (figure 9.a.), une trajectoire de type droite (figure 9.b.) et une trajectoire cyclique de type cercle (figure 9.c.).

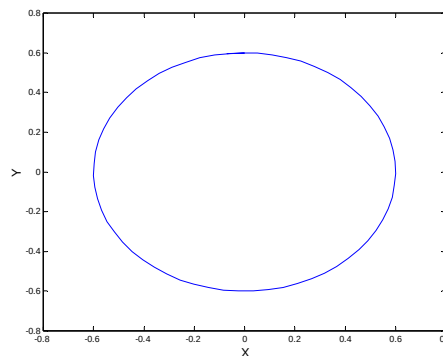
- Dans le cas des deux premières trajectoires le manipulateur débute à partir d'un point « A ». Il parcourt la trajectoire pour aboutir au point « B ».
- Dans le cas de la trajectoire cyclique de type cercle, le manipulateur débute à partir d'un point « A ». Il parcourt la trajectoire dans le sens inverse des aiguilles d'une montre et retourne au point initial « A ».



- a -



- b -



- c -

Figure 9. Les différentes trajectoires désirées

a. Trajectoire arc de cercle, b. Trajectoire droite et c. Trajectoire cercle.

D. Cas de la trajectoire de type arc cercle

La figure 10 illustre les configurations articulaires obtenues par la méthode de Levenberg-Marquardt. Nous pouvons constater l'efficacité de la méthode proposée.

En effet, les courbes (figure 11) représentant l'évolution temporelle des positions articulaires montrent une continuité entre les points obtenus. De plus aucune position singulière n'apparaît.

Ainsi les configurations obtenues sont réalisables par le robot. Elles peuvent donc être utilisées lors d'une commande faisant appel au modèle géométrique inverse.

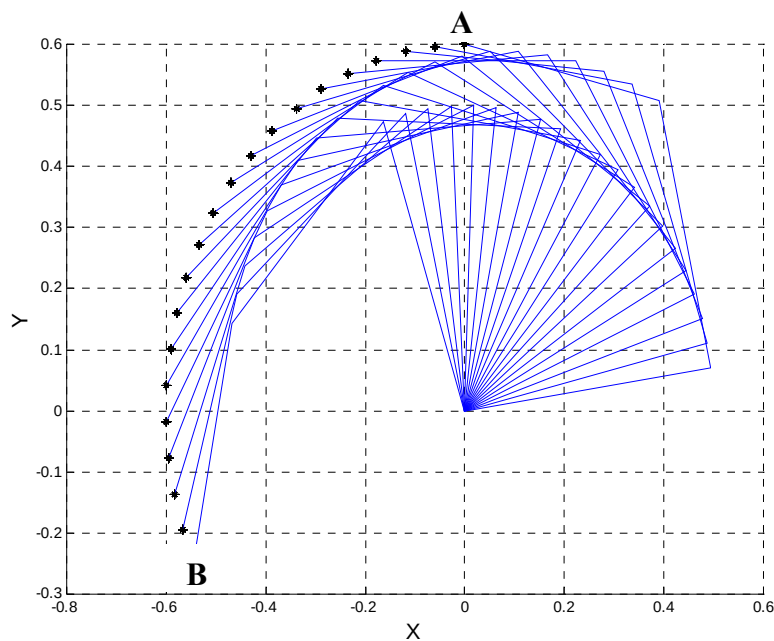
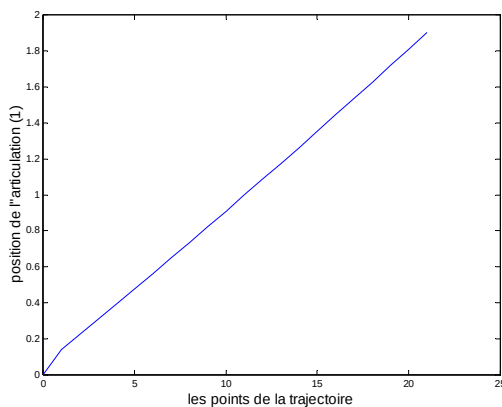
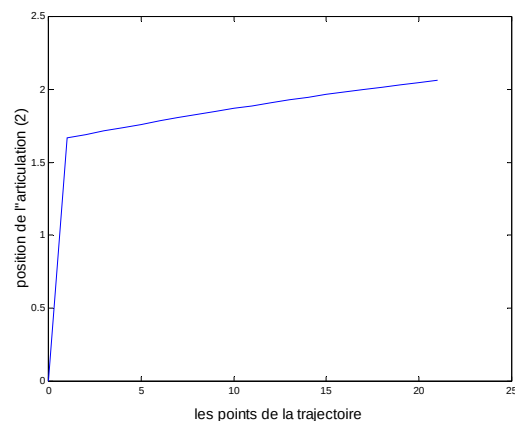


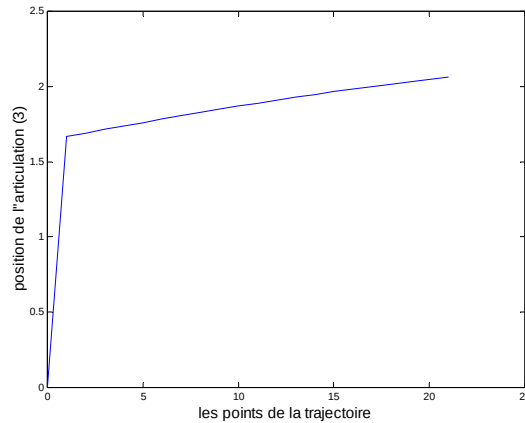
Figure 10. Configurations du bras manipulateur.



- a -



- b -

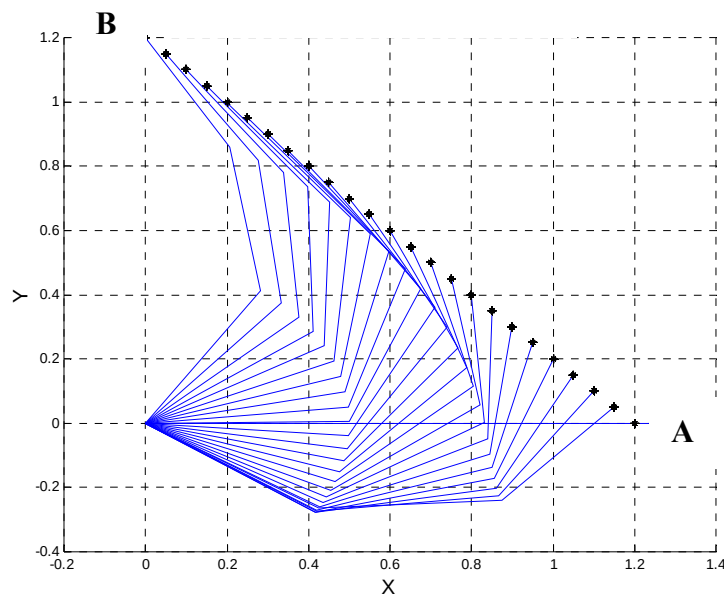


- c -

Figure 11. Positions des articulations*a. Articulation 1, b. Articulation 2 et c. Articulation 3.***E. Cas de la trajectoire de type droite**

Dans ce cas, la trajectoire désirée est de type droite. Connaissant la position de l'effecteur l'algorithme doit calculer les positions articulaires. Ces positions doivent être évaluées de sorte que l'organe terminal du manipulateur redondant se déplace sur la trajectoire désirée du point initial « A » au point final « B ».

La figure 12 illustrant les configurations articulaires obtenues par la méthode de Levenberg-Marquardt, montrent l'efficacité de la méthode.

**Figure 12.** Configurations du bras manipulateur.

De même que pour une trajectoire de type arc de cercle, les évolutions temporelles des positions articulaires (figure 13) montrent une continuité entre les configurations obtenues. De plus aucune position singulière n'apparaît durant le mouvement.

Ainsi, dans le cas d'une commande faisant appel au modèle géométrique inverse, les configurations obtenues sont utilisables. Le programme développé peut donc être utilisé dans une boucle d'asservissement dans le cas de robot redondant à trois axes rotoïdes

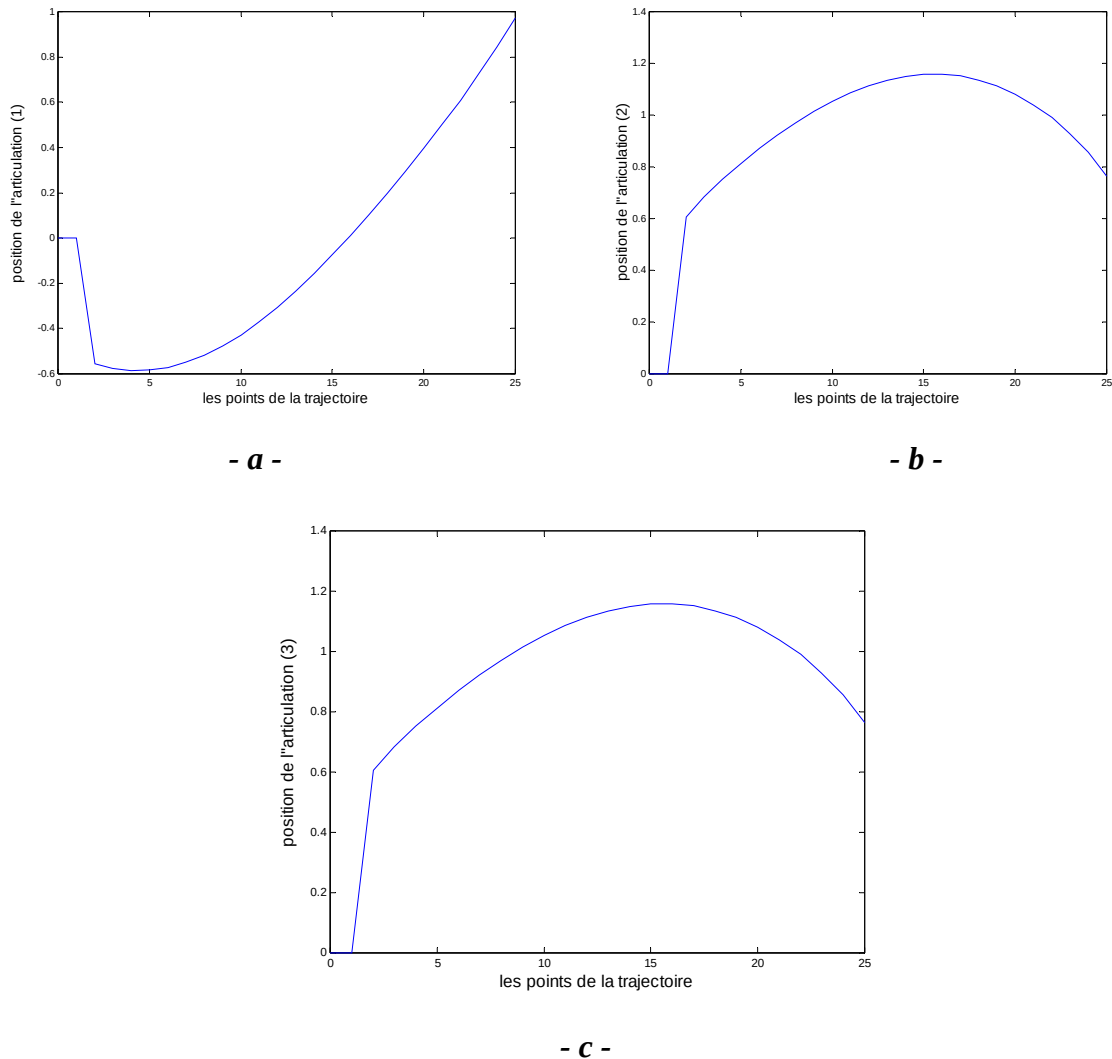


Figure 13. Positions des articulations

a. Articulation 1, b. Articulation 2 et c. Articulation 3.

F. Cas de la trajectoire cercle

Dans cette simulation, nous avons considéré une trajectoire cyclique de type cercle. Au cours de chaque cycle, le manipulateur débute à partir d'un point « A » et parcourt la trajectoire dans le sens inverse des aiguilles d'une montre pour retourner enfin vers le point initial.

Les configurations articulaires obtenues par la méthode de levenberg-Marquardt sont présentées sur la figure 14.

La figure 13 illustre les évolutions temporelles des positions articulaires au cours d'un cycle. Ces résultats font apparaître une continuité entre les différentes configurations engendrées par l'algorithme. De plus aucune position singulière n'apparaît durant le mouvement. Ceci permet de valider l'algorithme de calcul du modèle géométrique inverse d'un robot manipulateur redondant que nous proposons.

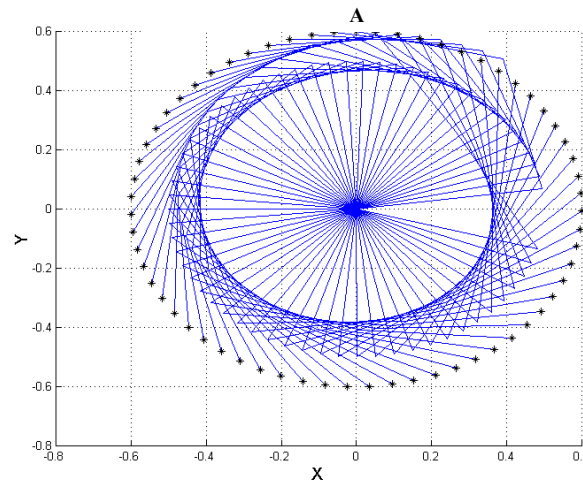
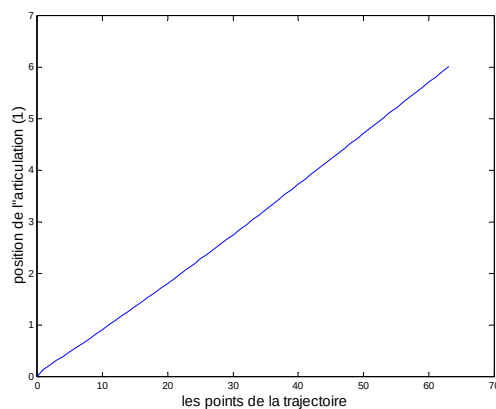
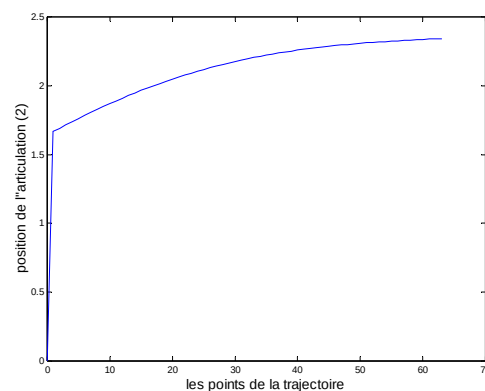


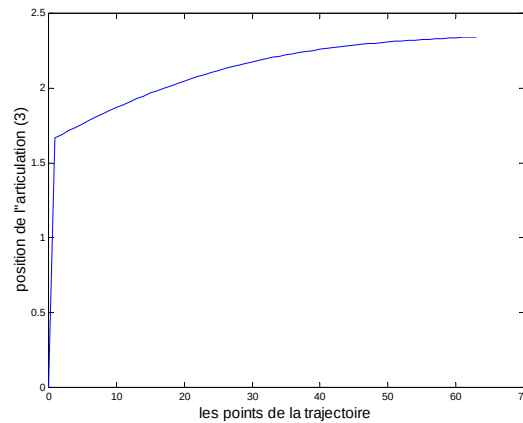
Figure 14. Configurations du bras manipulateur.



- a -



- b -



- c -

Figure 15. Positions des articulations

a. Articulation 1, b. Articulation 2 et c. Articulation 3.

Dans cette première partie, nous avons mis en évidence l'approche de la méthode de la boucle fermée, en utilisant l'algorithme de Levenberg-Marquardt. Afin que notre étude soit complète, nous proposons de présenter et d'évaluer les performances de la méthode d'identification du modèle en utilisant une approche connexionniste basée sur les réseaux de neurones à mémoires.

III.5. Application d'une approche connexionniste

Dans cette partie, nous proposons une approche connexionniste pour résoudre le problème de l'inversion du modèle géométrique d'un bras manipulateur redondant. En effet, nous appliquons les réseaux de neurones à mémoires pour identifier l'inverse de ce modèle. La complexité de l'inversion s'explique par la popularité des solutions et par le choix de celle qui satisfait au mieux le critère donné parfois assujéti à un ensemble de contraintes. L'intérêt de la méthode proposée est qu'elle offre substantiellement une meilleure précision que les méthodes classiques, évite le calcul de l'inverse ou pseudo inverse de la matrice Jacobienne et ne produit pas de problèmes tel que singularité, redondance, et complexité des calculs, etc.. [Ram 94] [Gue 88].

Dans la présente étude, nous utilisons un réseau de neurones à mémoire, l'adaptation des paramètres est réalisée par la méthode du gradient. Le réseau de neurones obtenu après apprentissage hors ligne est introduit comme fonction statique non linéaire dans la structure de commande.

III.5.1 Réseau de neurones à mémoire

L'architecture d'un réseau de neurones à mémoires est présentée sur la figure 16 [Dou 97]. Elle comporte une couche d'entrée E de « n_e » neurones, une couche cachée C de « n_c » neurones et une couche de sortie S de « n_s » neurones. Chaque neurone du réseau est un neurone à mémoire.

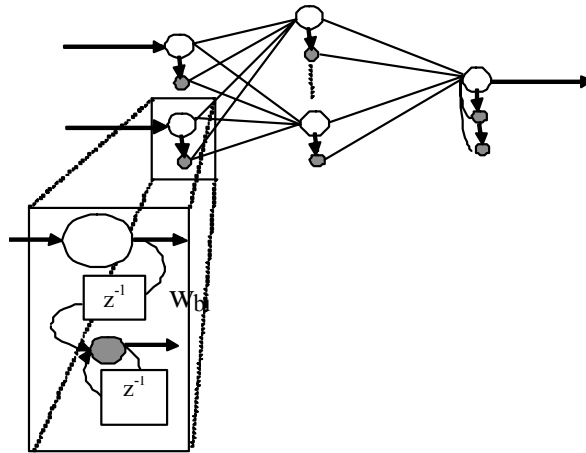


Figure 16. Architecture d'un réseau de neurones à mémoires

On constate qu'il s'agit d'un réseau statique multicouche dont les neurones (en blanc) ont une ou plusieurs cellules mémoires (en gris). La liaison entre chaque neurone et sa mémoire est pondérée par un coefficient W_{bi} qui devra être ajusté lors de l'apprentissage.

Cette mémoire est, à sa sortie rebouclée sur elle même, ce qui rend ce réseau dynamique (par introduction d'un retard), cette boucle étant elle aussi pondérée. Comme pour le réseau statique, les sorties de neurones de chaque couche vont vers les neurones de la couche suivante, on a en outre ici les sorties des cellules mémoires qui vont vers la couche suivante.

L'intérêt d'utiliser des réseaux de neurones à mémoires réside dans la capacité de prise en compte la dynamique du système.

III.5.2. Principe de la méthode proposée

Le principe de cette méthode consiste à définir un nouveau vecteur du modèle géométrique direct: le modèle géométrique direct étendu [Ber 03]. Ce dernier relie les coordonnées cartésiennes « X » aux coordonnées articulaires « q » selon la relation suivante [Ahm 93] [Kle 95]:

$$X = F(q) = \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} f(q) \\ g(q) \end{bmatrix} \dots\dots\dots (III.33).$$

Où « f(q) » est le modèle géométrique direct et « g(q) » est un vecteur de contrainte dans le cas des robots redondant [Bal 84].

$$g(q) = \left[\frac{\partial \phi(q)}{\partial q} \right] \cdot [N]^T = 0 \dots\dots\dots (III.34).$$

Avec : $\phi(q)$ le critère à minimiser,

$$N = \det (J_e) \quad N^\sim \quad avec \quad N^\sim = \begin{bmatrix} J_a^{-1} \cdot J_b \\ -I_{n-m} \end{bmatrix} \dots\dots\dots (III.35).$$

« J_a » est une matrice carrée de dimension (m) constituée des « m » premières colonnes de « J ». « J_b » est une matrice de dimension (m x (n-m)) formé du reste des colonnes. $J = [J_a \ J_b]$. I est la matrice Identité.

Les équations (III.33) et (III.34) montrent donc que le vecteur de position articulaire « q » permet au manipulateur de se positionner à la position cartésienne désirée « X » tout en satisfaisant le critère $\phi(q)$ [Moh 96].

Dans le cas de notre présente étude, le critère à minimiser choisi est le suivant :

$$\phi(q) = 0.5 \sum_{i=1}^n p_i L_i (q_i - q_{0i})^2 \dots\dots\dots (III.36).$$

Où :

« q_i » et « q_{0i} » désignent respectivement les positions articulaires réelles et désirées,
« L_i » est la longueur de l'axe « i ».
« p_i » est un gain constant.

Ce critère permet de minimiser le déplacement articulaire entre deux positions successives. En combinant les équations (III.33), (III.34) et (III.36) on obtient:

$$g(q) = N^T \begin{bmatrix} p_1 l_1 \cdot (q_1 - q_{01}) \\ p w_2 l_2 \cdot (q_2 - q_{02}) \\ \vdots \\ \vdots \\ p_n l_n \cdot (q_n - q_{0n}) \end{bmatrix} = 0 \dots\dots\dots (III.37).$$

Le modèle géométrique étendu d'un bras manipulateur redondant à « n » degrés de liberté dans le plan (« $m=2$ » $\Rightarrow$ le degré de redondance est égal à « $n-2$ ») est donc :

$$F(q)=X(q)=\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} l_1 \cdot c_1 + l_2 \cdot c_{12} + l_3 \cdot c_{123} + \dots + l_n \cdot c_{123\dots n} \\ l_1 \cdot s_1 + l_2 \cdot s_{12} + l_3 \cdot s_{123} + \dots + l_n \cdot s_{123\dots n} \\ g(q) \end{pmatrix} \dots\dots\dots (III.38).$$

Avec:

$$s_{ij} = \sin(q_i + q_j), \quad c_{ij} = \cos(q_i + q_j)$$

« x » et « y » sont les coordonnées cartésiennes de l'effecteur dans le plan ($m=2$).

« z » est un vecteur de contrainte de dimension ($n-2$).

« $l_1, l_2, \dots, l_n$ » sont les longueurs des axes du bras manipulateur.

Le problème de l'optimisation consiste à calculer le vecteur de coordonnée articulaire « q » pour les positions cartésiennes (X, z) données.

III.5.3. Architecture du réseau utilisé

La figure 17 illustre l'architecture du réseau de neurone à mémoire utilisé. Il est constitué de trois couches: Une couche d'entrée de trois neurones une couche cachée de quinze neurones et une couche de sortie de trois neurones. Les vecteurs d'entrées E et de

sortie S du réseau correspondent respectivement aux coordonnées cartésiennes « X, z » et aux coordonnées articulaires « q » du bras manipulateur. Les neurones utilisés ont tous un neurone à mémoire associé (figure 16).

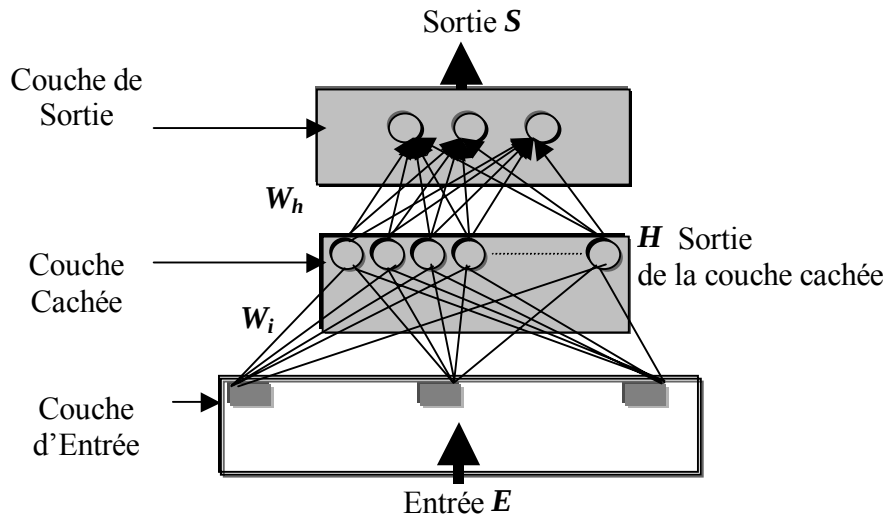


Figure 17. Architecture du réseau de neurones à mémoires utilisé

En utilisant une représentation matricielle, la sortie « H » de la couche cachée et la sortie « S » du réseau sont données comme suit :

$$\begin{aligned} H &= F_h (W_i E) \\ S &= F_o (W_h H) \end{aligned}$$

Avec W_i , W_h les matrices des poids à ajuster respectivement pour la couche d'entrée et la couche de cachée.

« F_h » et « F_o » sont les fonctions d'activation non linéaire respectivement des neurones de la couche cachée et de sortie. Dans notre cas ces fonctions sont choisies sigmoïdales pour les neurones de la couche cachée et linéaire pour les neurones de la couche de sortie.

III.5.4. Construction du réseau de neurones

L'utilisation des réseaux de neurones nécessite une phase de construction où sont déterminés ses paramètres. Cette phase consiste donc à utiliser un ensemble de couples d'entrées sorties pour identifier le système. Cette étape fait donc apparaître trois phases : une phase de construction de la base d'exemples, une phase d'apprentissage du réseau et enfin une phase de test.

Construction de la base d'exemples

Pour générer la base d'apprentissage du modèle géométrique inverse augmenté de contraintes nous utilisons, tout d'abord, la méthode du gradient citée précédemment. Nous utilisons tout d'abord cette méthode pour identifier le modèle inverse et sauvegardons l'ensemble des couples (q, X). La figure 18 illustre le principe de construction de la base d'exemples.

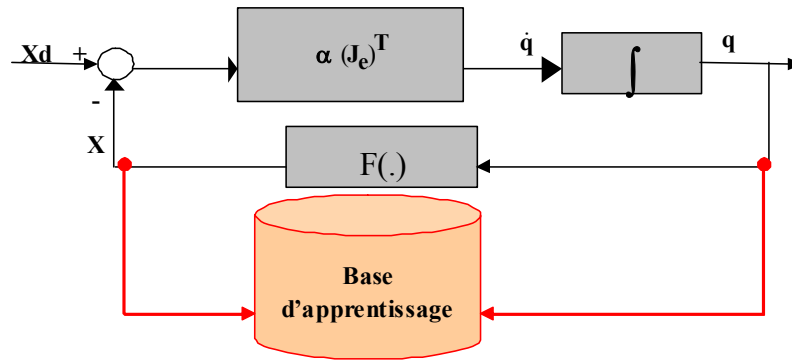


Figure 18. Construction de la base d'apprentissage.

Nous disposons donc d'une base constituée d'un ensemble de coordonnées cartésiennes et des coordonnées articulaires correspondantes. Ces dernières permettent de réaliser à la fois la tâche de positionnement et la tâche de contrainte.

Apprentissage

L'algorithme de la rétro propagation est utilisé pour ajuster les différentes matrices des poids du réseau en minimisant un critère E (W_i , W_h) défini comme la somme des erreurs quadratiques entre les sorties désirées et les sorties réelles du réseau (figure 19).

$$E(q) = 0.5 \sum_{i=1}^n (\hat{q}_i - q_{0i})^2 \dots\dots\dots (III.39).$$

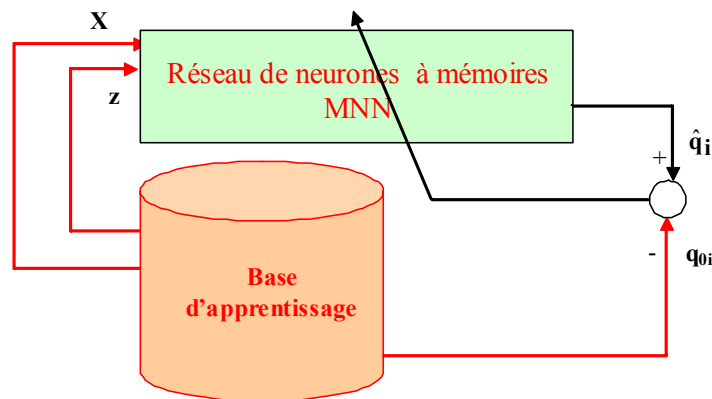


Figure 19. Initialisation du réseau de neurones

Après un nombre d'itération prédéterminé, on réalise l'apprentissage avec l'algorithme du gradient combiné avec une recherche unidimensionnelle du pas d'apprentissage (méthode de Goldstein) [Saa.97]. Le détail des équations d'apprentissage est donné en annexe 2.

L'apprentissage du réseau est de type batch. Les poids synaptiques du réseau sont initialisés aléatoirement entre -1 et +1. Le pas d'apprentissage est initialisé à 0.0001. Le pas d'incrément et de décrémentation du pas d'apprentissage sont respectivement 1.05 et

0.75. Le momentum est de 0.95. Les valeurs d'entrées et de sorties désirées du réseau sont normalisées entre -1 et 1.

Sur l'ensemble d'apprentissage précédemment construit on a pris deux tiers pour effectuer l'apprentissage du réseau de neurones et un tiers pour tester ce réseau soit dans notre cas 1000 points et 500 points.

Les résultats obtenus pour l'apprentissage du modèle géométrique inverse sont représentés par la figure 20. La somme des erreurs quadratiques (SSE) sur l'ensemble d'apprentissage est égale à 0.05 après 1000 itérations. La convergence est très rapide au début des itérations, puis devient très lente. L'erreur finale après 5000 itérations est de l'ordre de 0.02.

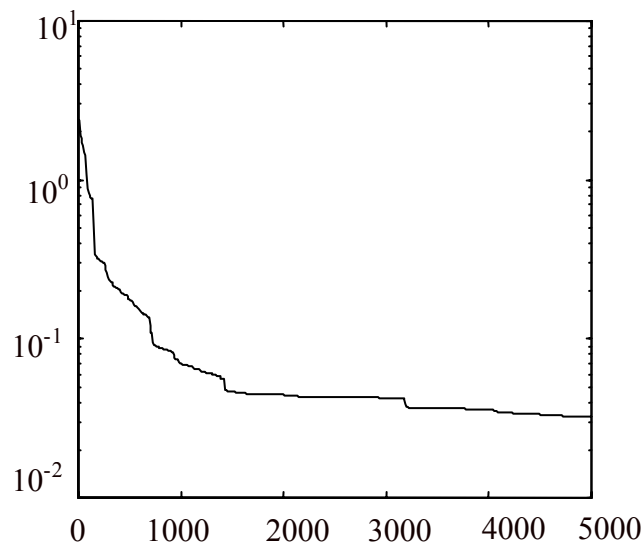


Figure.20: Somme des erreurs quadratiques en fonction du nombre d'itération.

Test de généralisation

Le réseau utilisé dans cette partie est celui obtenu après 1000 itérations d'apprentissage. La base d'exemples utilisé pour le test de généralisation du réseau est constituée de 500 points. Lors de ce test on a remarqué que la somme des erreurs quadratiques obtenues est très proche de la valeur trouvée sur les points appris. Elle est égale à 0.08. L'analyse des résultats des différents points utilisés lors du test de généralisation montre que l'erreur absolue entre les valeurs désirées et le résultat est d'autant plus faible si les points non appris sont très proches des points appris.

Le réseau de neurones pourra donc être introduit comme fonction statique non linéaire dans une structure de commande.

III.5.5. Simulation

Afin de tester les performances de la méthode connexionniste proposée, nous l'avons appliqué à un bras manipulateur redondant à trois degrés de liberté. Nous avons utilisé une loi de

commande simple en position pour faire suivre à notre manipulateur une trajectoire définie par une droite de type « $y=-x+0.63$ » en minimisant le critère choisi:

$$\phi(q) = 0.5 \sum_{i=1}^n p_i L_i (q_i - q_{0i})^2$$

Le réseau de neurones à mémoire obtenu précédemment par apprentissage est implémenté pour la résolution du modèle géométrique inverse du bras manipulateur. En effet, il est utilisé pour donner les positions articulaires $q_d(t)$ à partir des positions cartésiennes désirées $x_d(t)$. La figure 21 illustre l'évolution du bras manipulateur le long de la trajectoire.

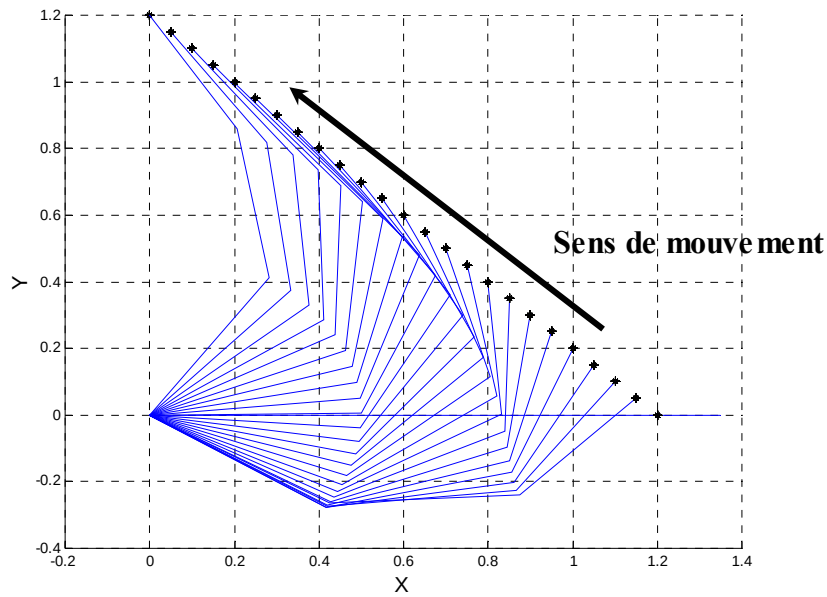


Figure 21. Configurations du robot

L'erreur de suivi opérationnel (e_x , e_y) et l'évolution du critère « $g(q) = N^T \frac{\partial \phi(q)}{\partial q}$ » reste très faible le long de la trajectoire (figure 21). Le réseau de neurones à mémoire a donc permis de fournir à la loi de commande, les positions articulaires nécessaires au suivi de la trajectoire désirée en minimisant le critère donné.

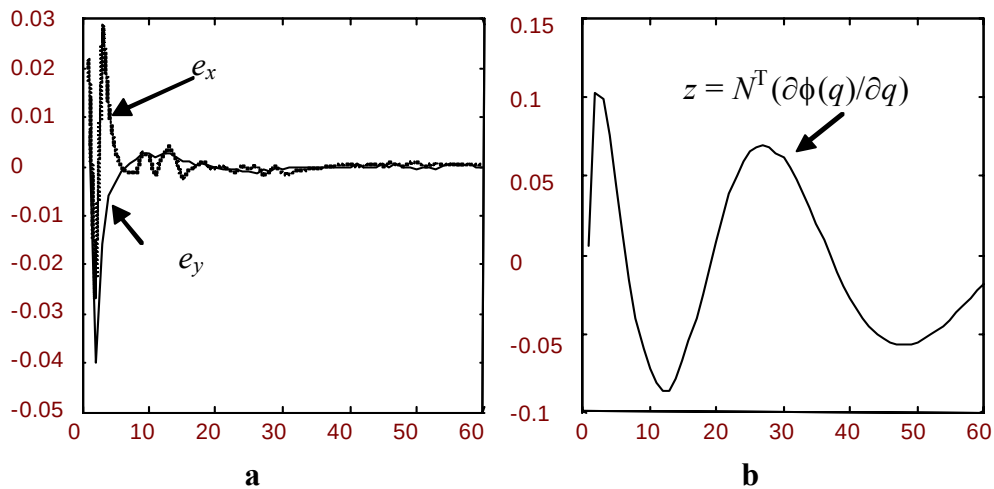


Figure 22. (a) Erreur de suivi opérationnel et (b) Evolution du critère le long de la trajectoire.

III.6 Conclusion

Dans ce chapitre, nous avons présenté deux solutions au problème de la résolution inverse d'un système d'équation. La première est basée sur la méthode de la boucle fermée en utilisant l'algorithme de Levenberg-Marquardt. La seconde utilise une approche connexionniste pour l'identification du modèle. Ces deux méthodes permettent d'obtenir le modèle géométrique inverse d'un robot manipulateur redondant assujetti à un ensemble de contraintes.

Le principe de la première méthode réside en une approximation locale de « F^{-1} » effectuée sur chaque point de la trajectoire. Une étude de la stabilité menée via une fonction de Lyapunov, nous a conduit au choix d'une fonction stable.

L'application de l'algorithme de Levenberg-Marquardt proposé dans le cas de différentes trajectoires a permis de montrer son efficacité compte tenu de la continuité entre les différentes configurations obtenues. De plus les solutions données par cet algorithme ne présentent aucune singularité.

Le principe de la deuxième méthode, consiste à appliquer des réseaux de neurones à mémoires pour l'inversion du modèle géométrique d'un robot redondant, les tests réalisés alors et l'analyse de notre méthode et de ces résultats nous ont permis de mettre en évidence des avantages, certes, mais aussi des inconvénients liés à la résolution statique de notre problème. En effet, la base d'apprentissage du réseau est extrêmement difficile à déterminer car elle doit englober un ensemble de points suffisamment significatif de l'espace de travail du manipulateur. Son choix induit directement une bonne ou une mauvaise convergence de l'apprentissage et de la généralisation. Un réseau statique correspond à la structure du bras manipulateur, tout changement de cette structure nécessite un nouvel apprentissage et donc le choix d'une nouvelle base. De plus, les paramètres du réseau: pas d'apprentissage, nombre d'itérations maximal, nombre de couches, nombre de neurones, choix de la fonction d'activation et l'initialisation des coefficients synaptiques, sont difficiles à déterminer et influent fortement sur la qualité d'apprentissage. Les résultats obtenus sont satisfaisants, toute fois la stabilité du système dépend des paramètres du réseau et la modification de la tâche à réaliser nécessiterai la reconstruction de la base d'exemple et un réapprentissage. Il est toutefois possible d'introduire un mécanisme d'adaptation en ligne. Ceci pourra faire l'objet de travaux futures.

Compte tenu de l'analyse des deux méthodes proposées, de la diversité des tâches considérés dans la présente étude, nous choisissons d'utiliser l'algorithme Levenberg-Marquardt dans la structure de commande d'un bras manipulateur redondant. Ainsi, les coordonnées articulaires pourront être déduites des coordonnées cartésiennes dès que cela sera nécessaire.

commande hybride force-position

IV.1. INTRODUCTION:

Au cours de certaines tâches, il est souvent nécessaire que le point d'application de la force exercée se déplace sur la surface de l'environnement. Le robot doit donc suivre une trajectoire sur la surface tout en maintenant un effort désiré normal à cette surface. Il faut donc, à la fois, contrôler la position du robot manipulateur et l'effort exercé par celui-ci sur l'environnement.

Lorsque le robot est contraint par l'environnement suivant toutes les directions, aucun déplacement de l'organe terminal n'est possible, il ne peut qu'exercer des efforts sur l'environnement. Par contre, lorsque le manipulateur opère dans l'espace libre, l'absence de tout contact empêche la création d'efforts.

L'exécution d'une tâche de commande hybride force-position résulte d'une stratégie dont la synthèse a été formalisée par Mason [Mas 81]. Il apparaît que l'espace doit être subdivisé en deux sous espaces complémentaires, l'un constitué des directions commandables en position et l'autre de celles commandables en effort.

A partir de cette décomposition de l'espace cartésien, de nombreuses méthodes de commande hybride force-position ont été proposées. Ces commandes sont constituées de deux boucles parallèles d'asservissement: une boucle de position et une boucle d'effort dont les effets sont sommés sous forme de couple d'effort ou de vitesse pour commander des actionneurs.

Les structures de commande proposées dans la littérature diffèrent essentiellement par l'enchaînement des transformations géométriques et cinématiques [Saa 97]. La réalisation simultanée d'une consigne d'effort et d'une consigne de position par un même manipulateur est la définition de la commande hybride force-position.

Ce chapitre a pour objectif de présenter quelques structures de commande hybride force-position les plus rencontrées dans la littérature, d'analyser leurs avantages et inconvénients, afin d'en adopter une pour la suite de notre travail.

IV.2. PRINCIPE DE LA COMMANDE HYBRIDE FORCE-POSITION

La commande hybride force-position consiste à utiliser simultanément les informations d'effort et de position afin de satisfaire à la fois, des contraintes en position et en efforts.

Les directions qui sont contraintes en effort sont commandées en position tandis que celles qui sont contraintes en position sont commandées en effort. En effet lorsque le

manipulateur est contraint par l'environnement suivant toutes les directions, aucun déplacement du point terminal n'est possible; il ne peut qu'exercer des efforts sur l'environnement. Par contre, lorsque le manipulateur évolue dans l'espace libre, l'absence de tout contact empêche la création d'effort. Le contrôle de force et le contrôle de position sont donc deux notions duales [Deg 93].

Pour simplifier la présentation de ce problème complexe, nous présentons l'exemple ci dessous (figure 1) :

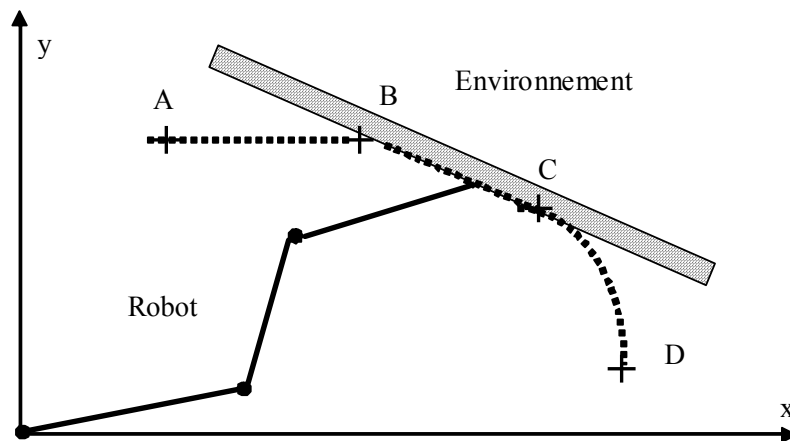


Figure 1. Exemple de tâche de commande hybride
(Suivi de contour dans le plan)

Cet exemple permet de voir le comportement du robot dans les différentes phases d'une tâche de commande hybride. En effet, dans la tâche à réaliser, le manipulateur se déplace dans le plan (repère $\{x, y\}$) du point «A» au point «D».

Le mouvement de l'effecteur se décompose comme suit:

1) Phase d'approche

Du point «A» au point «B», le robot se déplace dans un espace libre de toute contrainte. Aucune force d'interaction n'est créée. Le contrôle du déplacement est réalisé par un asservissement pur de position.

2) Mise en contact

Au point «B», l'effecteur entre en contact avec la surface de l'environnement. Ce point correspond donc à une transition entre un contrôle de position pur et un contrôle d'effort qui doit compenser éventuellement l'effet d'un choc lorsque la position du point est méconnue.

3) Phase hybride

Du point «B» au point «C», l'effecteur doit longer le contour de l'environnement. La géométrie de la tâche provoque des contraintes sur le robot qui engendre des forces d'interactions importantes. Celles-ci doivent être contrôlées par le suivi simultané d'une consigne de force normale à l'environnement (figure 2).

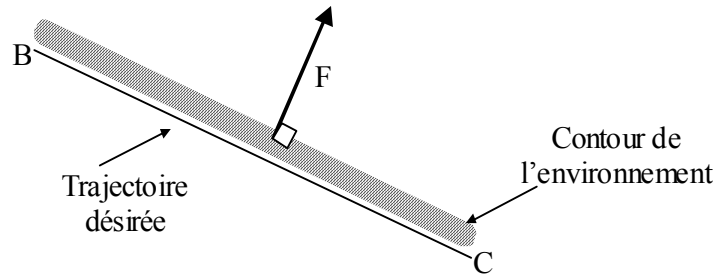


Figure 2. Phase hybride

4) Phase de retour

Du point «C» au point «D», le robot quitte l'environnement, le déplacement est dans l'espace libre.

La réalisation de cette tâche peut être décrite, dans l'espace de référence, par deux vecteurs fonction du temps caractérisant la trajectoire $X_d(t)$ et consigne de force $F_d(t)$

IV.3. STRUCTURES DE COMMANDE HYBRIDE FORCE-POSITION:

On présente dans cette partie deux familles de structures de commande avec asservissement simultané de position et d'effort. La première famille est constituée de la commande de Craig et Raibert [Rai 81], la commande de Reboulet et Robert [Reb 85] et enfin la commande de khatib [Kha 87]. La seconde, consiste en la commande externe [Per 91].

IV.3.1. COMMANDES HYBRIDES FORCE-POSITION

Nous allons maintenant étudier plus particulièrement quelques structures choisies pour leur représentativité par rapport à l'ensemble des structures proposées dans la littérature. Il existe essentiellement trois formulations concernant la commande hybride force-position.

Ces structures de commande dépendent du type de variables qui sont sommées.

- Dans la structure proposée par CRAIG et RAIBERT [Rai 81], la sommation s'effectue au niveau du vecteur couple articulaire.
- Dans celle de KHATIB [Kha 87], la sommation se fait au niveau du vecteur force exprimée dans l'espace opérationnel.
- Enfin, REBOULET et ROBERT proposent une structure dont la sommation s'effectue au niveau du vecteur vitesse exprimé dans l'espace opérationnel [Reb 85].

L'idée originale de ce type de commande est déduite de la constatation suivante: un même degré de liberté ne peut être commandé à la fois en force et en position.

Le système subissant une série de contraintes physiques, la commande hybride impose à l'organe terminal une série duale de contraintes dites artificielles choisies de façon à imposer une trajectoire désirée et à appliquer sur l'environnement les forces souhaitées.

La commande est réalisée dans l'espace cartésien et les auteurs montrent, en utilisant la description de tâche de Mason [Mas 81], que l'on peut commander le robot manipulateur en position et en effort en ayant deux boucles d'asservissement distinctes et en sommant le résultat des erreurs au niveau de l'espace articulaire.

L'espace de tâche étant scindé en deux sous-espaces orthogonaux, le schéma global est constitué de deux boucles parallèles et complémentaires disposant chacune de son propre contrôleur. Elles sont asservies indépendamment suivant une structure de commande classique en force ou en position. La sommation des efforts fournit les couples de commande nécessaires à la réalisation des consignes.

Notons que les consignes, les erreurs de suivi et la matrice de sélection sont spécifiées dans l'espace généralisé.

b) Structure de commande de Craig et Raibert

Il y a déjà quelques années que des chercheurs, dans le domaine de la robotique, se sont penchés sur le principe de la commande hybride et le premier schéma qui est apparu est celui de CRAIG et RAIBERT [Rai 81].

Cette structure de commande peut être représentée par le schéma fonctionnel de la figure 3.

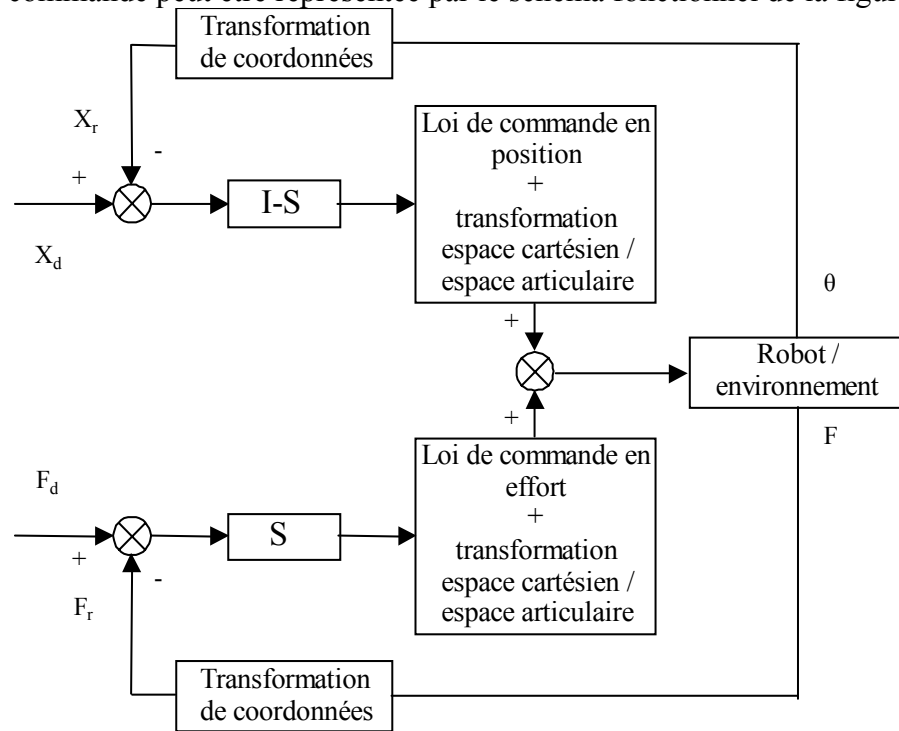


Figure 3. Commande hybride d'après Craig et Raibert

La plupart des schémas de commande hybride force-position que l'on peut rencontrer actuellement dans la littérature s'inspirent de cette structure.

Pour assurer l'orthogonalité des commandes issues de chacun des asservissements et éviter tout conflit au niveau des actionneurs, il est nécessaire d'inclure dans ces structures une matrice de sélection S , apportée par la théorie de Mason.

La matrice de sélection S et sa complémentaire $I-S$ (I matrice identité) décomposent l'espace cartésien en deux sous-espaces orthogonaux associés l'un à l'asservissement en effort et l'autre à celui de position. Elles sont diagonales. Un élément de cette diagonale vaut 1 si la composante correspondante du vecteur de force est commandable. Il vaut 0, si la composante correspondante du vecteur vitesse (position) est contrôlable. On assure de ce fait qu'une direction ne peut être commandée, à la fois, en position et en effort. Cette matrice dépendra donc de la configuration de la tâche [Kha 99].

d) Structure de commande de Reboulet et Robert

Dans cette méthode proposée par Reboulet et Robert [Reb 85b] la somme des effets est une somme de vitesses. Cette méthode utilise directement la formalisation de tâche de Mason.

Comme les consignes sont décrites dans le repère tâche, la structure de commande est organisée sur trois niveaux de repères différents.

- Les erreurs de suivi sont formées dans le repère de tâche pour appliquer directement la matrice de sélection.
- Les mesures de force et de position sont fournies dans le repère de référence et le robot est commandé dans l'espace généralisé.
- Des transformations géométriques sont introduites à toutes les étapes de l'asservissement.

Le schéma de principe est représenté par la figure 4.

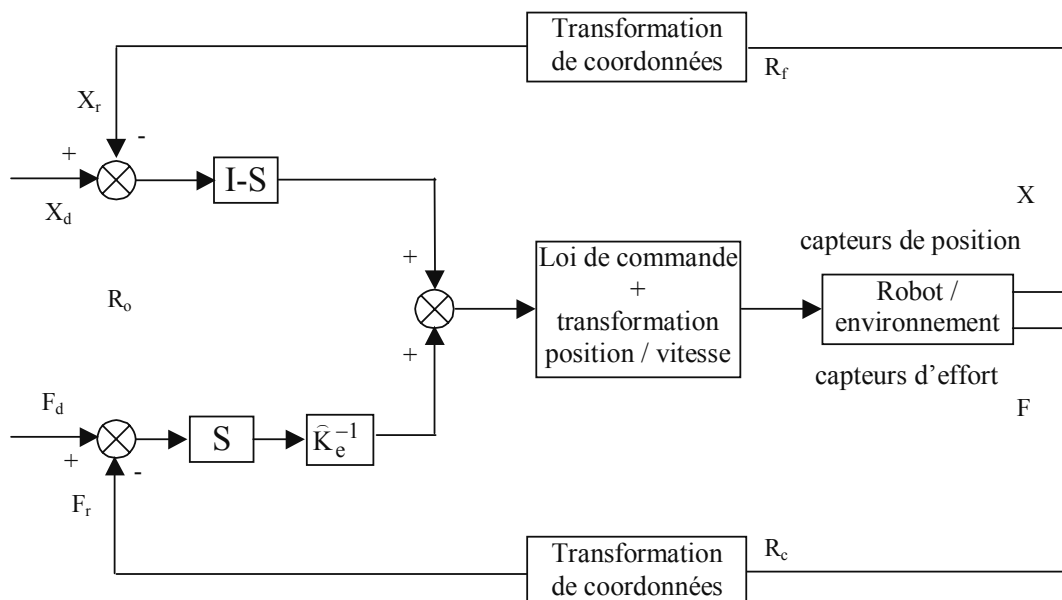


Figure 4. Structure hybride d'après Reboulet et Robert

Par l'intermédiaire de l'inverse de la raideur estimée, l'erreur de force devient une erreur de position. La loi de commande, commune aux deux boucles est implantée dans l'espace généralisé. Son entrée est un écart articulaire.

e) Structure de commande de Khatib

D'autres chercheurs se sont également intéressés à la commande hybride force-position et de nouveaux schémas ont été élaborés [Kha 86]. Khatib et Craig proposent une formulation de la commande hybride force-position dans l'espace opérationnel. Tous les niveaux de cette structure sont placés dans le repère de référence, que ce soit la somme des effets de chacune des boucles sous forme d'efforts cartésiens, la génération de consignes ou l'implantation de correcteurs.

Le schéma de principe est donné par la figure 5.

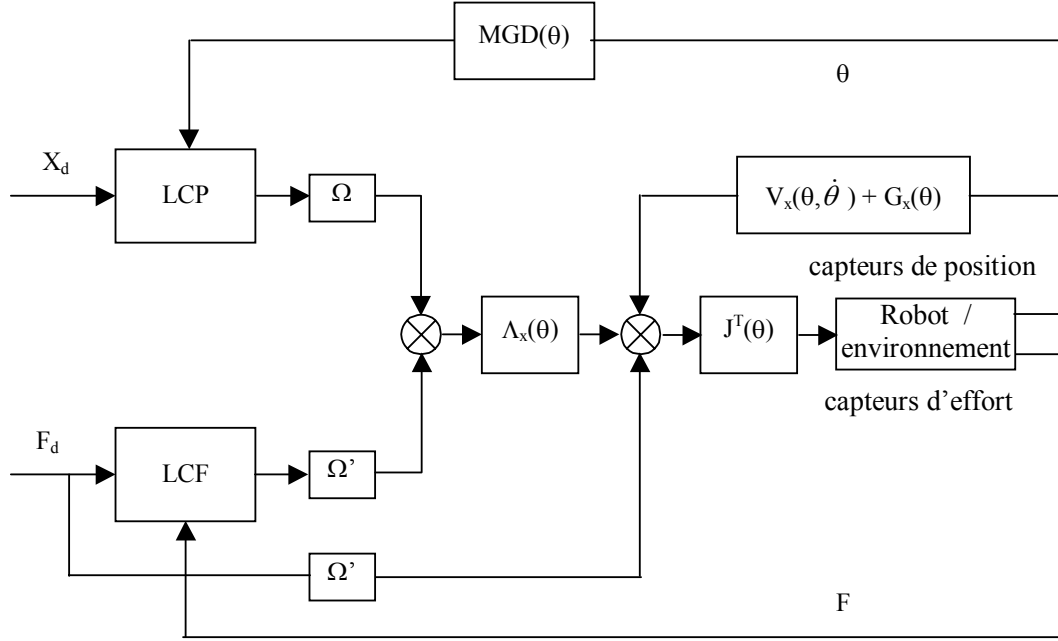


Figure 5. Structure hybride dynamique (Khatib).

Où :

- LCP et LCF désignent respectivement la loi de commande en position et la loi de commande en effort.
- Λ_x est le découplage par la matrice inertielle calculée dans le repère opérationnel, J^T est la matrice Jacobienne transposée.
- V_x et G_x sont les termes de Coriolis et de gravité du modèle dynamique du robot dans l'espace opérationnel.
- Ω et Ω' sont les matrices de spécification de tâches. Elles sont constituées par la matrice de sélection S et des rotations directes et inverses qui permettent le passage entre le repère de référence et le repère des contraintes. Ces matrices assurent l'orthogonalité des commandes issues des contrôleurs.

$$\Omega = R^T (I-S) R \dots \dots \dots (IV.1)$$

$$\Omega' = R^T S R \dots \dots \dots (IV.2)$$

A partir d'une description du contour de l'environnement dans l'espace de référence, la consigne de position X_d est la trajectoire sur la surface. La consigne de force est un vecteur normal à cette surface à chaque instant.

Cette solution propose de sommer les erreurs de force dans l'espace cartésien, de faire ensuite un découplage non-linéaire et le changement de coordonnées par l'intermédiaire de la matrice Jacobienne transposée. Aucune transformation géométrique sur les mesures fournies par les capteurs n'est nécessaire.

Une analyse de tâche devra être mise en œuvre pour déterminer les axes du repère de tâche contrôlés en position et les axes contrôlés en force.

Khatib a introduit la matrice inertielle $\Lambda_x(\theta)$ uniquement dans l'asservissement de position pour découpler les déplacements du point terminal dûs aux commandes issues du contrôleur de position, dans l'espace opérationnel.

IV.3.2. STRUCTURE DE COMMANDE EN EFFORT EXTERNE

Cette structure fait apparaître une hiérarchisation de la boucle d'effort (appelée boucle externe) sur la boucle d'asservissement en position au lieu d'une mise en parallèle des contrôleurs [Per 03a].

La figure 6 permet de visualiser cette nouvelle disposition des contrôleurs, le caractère hiérarchiquement supérieur de la boucle en effort apparaissant ici clairement [Kha 99].

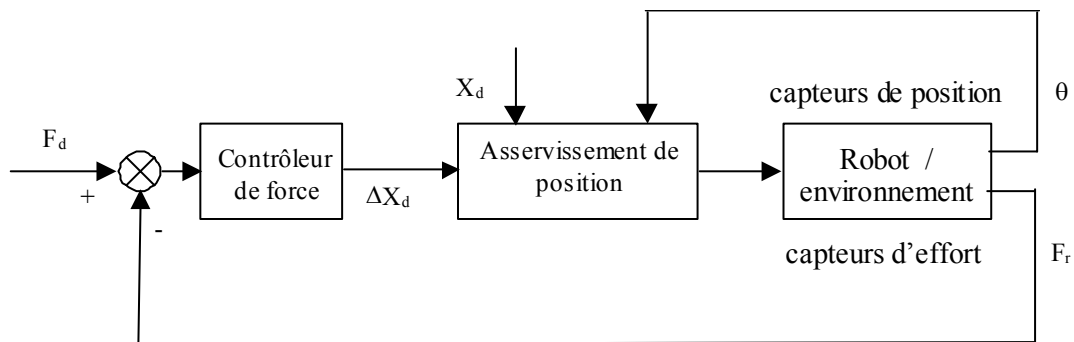


Figure 6. Structure de commande en effort externe

Où :

- F_d et X_d sont les consignes respectives de force et de position dans l'espace opérationnel,
- F_r est l'effort réel mesuré sur l'effecteur,
- θ est la position articulaire réelle,
- ΔX_d est un incrément de situation (position/orientation) engendré par le bloc de commande des forces.

L'asservissement de l'effort est réalisé à travers une conversion de l'erreur en force en une consigne de position « ΔX_d ».

L'idée maîtresse de cette structure est que, modélisant le contact du robot avec l'environnement par un ressort, une variation de l'effort peut être obtenue par un déplacement infinitésimal du point terminal. En effet, lorsque le robot se déplace d'une quantité ΔX dans une direction orthogonale à l'environnement (figure 7) l'effort « F » produit est proportionnel à la raideur de l'environnement K_e est au déplacement ΔX :

$$F = K_e \Delta X \dots \dots \dots (IV.3)$$

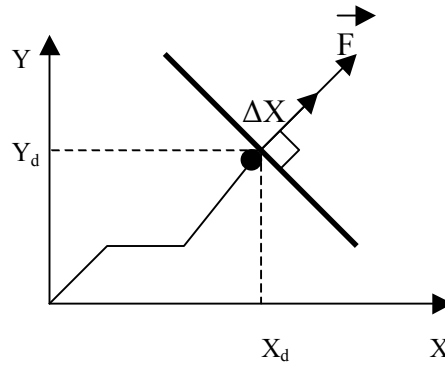


Figure 7. Création d'effort

Le but est donc d'ajouter à une consigne de position X_d un déplacement relatif ΔX_d issu de la boucle d'effort. A partir des efforts de consigne F_d et des efforts de retour F_r exercés au niveau de l'outil, nous calculons un incrément de situation (position/orientation) ΔX_d qui vient s'ajouter à la consigne de position désirée X_d pour donner une nouvelle consigne de position envoyée au système. D'ores et déjà, on constate qu'il est possible d'imposer un effort désiré, ce qui laisse envisager bon nombre d'applications.

L'incrément de situation ΔX_d issu de la différence $(F_d - F_r)$ est obtenu à partir d'une loi de commande en effort qu'il faut définir en fonction de la tâche à réaliser et des résultats escomptés [Per 91].

IV.3.3 ETUDE COMPARATIVE

L'analyse des structures de commande de Craig et Raibert, Reboulet et Robert et Khatib fait apparaître un certain nombre d'inconvénients dont [Per 03a]:

- Utilisation de plusieurs transformations géométriques, ces transformations et les commutations entre la boucle d'asservissement de force et de position, compliquent la mise en œuvre de cette structure.
- La mise en œuvre de ces structures nécessite la connaissance précise du modèle dynamique et celui de son interaction avec l'environnement
- La dynamique des réponses en force et en position dépend de la synthèse des deux correcteurs.
- L'utilisation de la matrice de sélection peut engendrer l'instabilité du système.

La structure de commande externe, quant à elle n'utilise pas de matrice de sélection. En effet, elle se base sur une hiérarchisation des deux boucles d'asservissements, plutôt que sur une parallélisation. Elle reste donc stable quelque soit la configuration de la tâche à réaliser [Saa 95].

De plus, cette structure n'utilise pas de transformation géométrique d'où la simplification et la facilité de sa mise en œuvre. Compte tenu de ces avantages, nous adopterons cette structure dans notre présente étude. Nous présentons donc dans le paragraphe suivant la synthèse de la structure de commande externe.

IV.4. SYNTHESE DE LA STRUCTURE DE COMMANDE EN EFFORT EXTERNE

L'équation d'évolution du système est donnée par le modèle dynamique qui relie les accélérations effectivement réalisées de chacune des articulations aux couples fournis.

Le vecteur « Γ_p » perturbations contient l'effet des efforts extérieurs exercés à l'extrémité du manipulateur et éventuellement l'effet de la pesanteur.

Les phénomènes inertiels sont caractérisés par une matrice $M(\theta)$ qui donne l'expression du modèle dynamique hors contact:

$$\ddot{\theta} = M^{-1}(\theta) [\Gamma_u - C(\dot{\theta}, \theta) \dot{\theta} - \Gamma_p] \dots \dots \dots (IV.3)$$

Où:

θ est le vecteur des positions et orientations articulaires,

Γ_u est le vecteur des couples de commande,

Γ_p est le vecteur des couples de perturbations.

Hors contact le robot peut donc être schématisé comme suit:

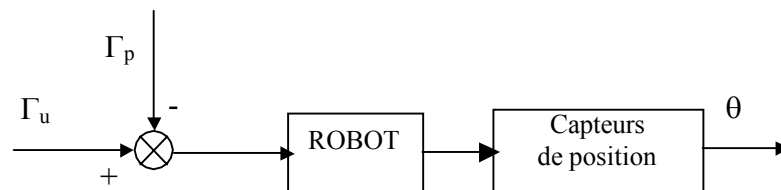


Figure 8. Modèle du robot hors contact

Les couples de commande sont fournis par les actionneurs des articulations, modélisés chacun par un gain, de façon simplifiée. Ainsi, nous pouvons écrire:

$$\Gamma_u = K_m \cdot U \dots \dots \dots (IV.4)$$

Où:

U est le vecteur des tensions de commande

K_m est la matrice diagonale des gains.

Lorsque le robot est en contact avec l'environnement, la force d'interaction « F_e » produite par l'environnement, agit comme un couple perturbateur sur le système mécanique. Elle est modélisée par un rebouclage de l'effort par l'intermédiaire de la transformation cinématique nécessaire " J^T ", un capteur de force fournit une mesure de l'effort « F_c » au point terminal dans le repère capteur (figure 9).

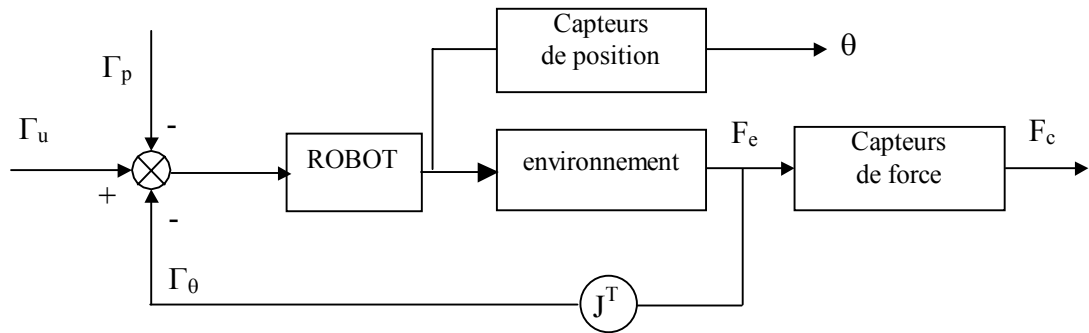


Figure 9. Modèle du robot en contact

Les déformations de l'environnement et du capteur, suivant les directions contraintes, créent un effort supposé ici proportionnel au déplacement:

$$F_e = K_e (X - X_e) \quad \text{si } X > X_e \quad \dots\dots\dots (IV.5)$$

Où:

$X - X_e$ est l'écart entre la position du point terminal et la position initiale de l'environnement sur une normale au contour dans un repère cartésien.

K_e est la raideur équivalente capteur-environnement

LE CONTROLEUR DE POSITION

Lorsque le manipulateur se déplace vers une position désirée X_d , son contrôle est exclusivement assuré par l'asservissement de position.

Lors d'un contact, les forces d'interaction n'agissent pas seulement comme des perturbations sur les actionneurs, mais modifient aussi la dynamique de l'asservissement par un bouclage de l'effort. Le déplacement ΔX_d issu de la loi de commande en effort, vient s'ajouter à la consigne de position X_d pour constituer l'entrée de commande de l'asservissement en position. Ceci n'apparaît pas explicitement dans la figure 10 car plusieurs possibilités sont envisageables; elles se distinguent par les transformations cinématiques ou géométriques qu'elles mettent en jeu.

Tout d'abord, l'asservissement peut être cartésien et porter sur la position X comparée à la consigne globale X_d^* , ($X_d^* = \Delta X_d + X_d$). Dans ce cas, les transformations utilisées sont internes à la boucle de position.

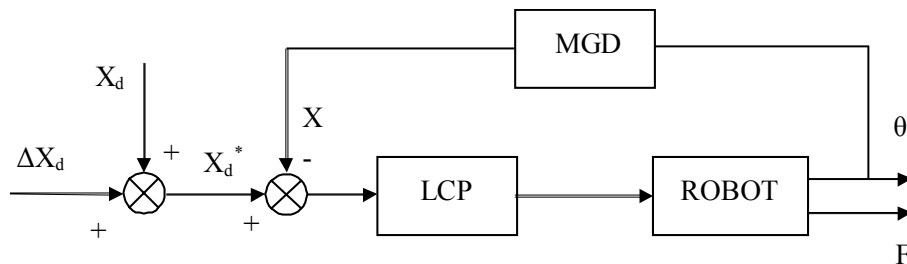


Figure 10. Structure de commande externe avec asservissement cartésien de la position

L'asservissement peut aussi être articulaire et porter sur la position articulaire θ qui est comparée à la position désirée θ_d issue des consignes X_d et ΔX_d . Dans ce cas, les consignes doivent être transformées, soit indépendamment (figure 11), soit directement après formation de la consigne globale (figure 12). Il est important de constater que dans ces deux cas, aucune transformation géométrique ou cinématique n'intervient dans la boucle interne de position.

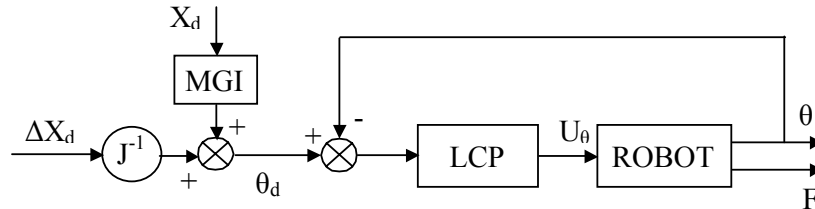


Figure 11. Asservissement articulaire de la position
(Transformation indépendante des consignes)

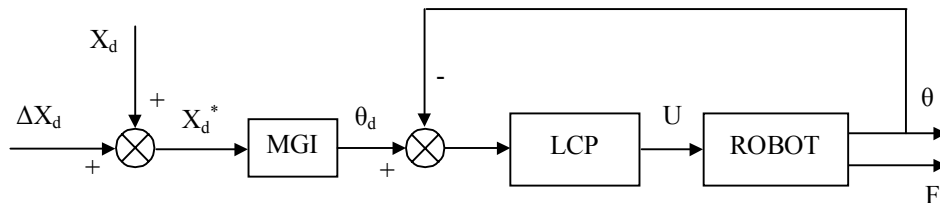


Figure 12. Asservissement articulaire de la position
(Transformation globale des consignes)

La possibilité d'asservir la position par un contrôle articulaire est l'un des intérêts majeurs de cette méthode. En effet, aucune transformation géométrique ou cinématique n'intervient dans la boucle de position. L'algorithme de contrôle s'en trouve grandement simplifié, ce qui diminue le temps de calcul pour un meilleur suivi de la consigne. De plus, le réglage de l'asservissement est totalement indépendant de la position instantanée du robot, évitant tout problème de stabilité. A partir de cette analyse, il reste maintenant à choisir entre les structures des figures 11 et 12.

Le calcul du modèle géométrique inverse dépend de la résolvabilité du modèle du robot. La non-linéarité des équations entraîne plusieurs solutions. Dans le cadre de notre étude, nous utilisons la méthode basée sur l'algorithme de Levenberg-Marquardt présentée dans le chapitre III.

Ainsi, la structure de la figure 12 sera retenue car seule le modèle géométrique inverse intervient dans les calculs ce qui réduit le temps de calcul.

LE CONTROLEUR D'EFFORT

Dans la structure de commande en effort externe, les incréments de positions viennent s'ajouter à la consigne de position. Ils sont issus de la loi de commande en effort (figure 13). Le correcteur utilisé peut être plus ou moins complexe en fonction des résultats escomptés.

En se plaçant du point de vue de l'asservissement de l'effort F , la consigne X_d intervient comme une perturbation. Une erreur statique apparaît et l'asservissement d'effort est affecté par la consigne de position. Mais pour ce cas on a besoin d'utiliser quelques transformations géométriques tel que le modèle géométrique direct et inverse pour passer d'un espace articulaire vers un espace cartésien. Le schéma global d'asservissement est donc :

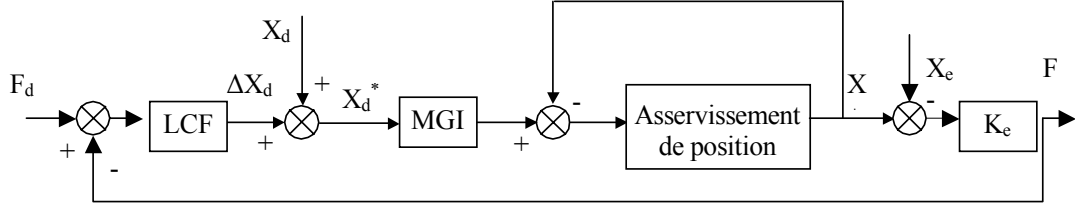


Figure 13. Asservissement de force.

SCHEMA GLOBAL DE LA COMMANDE EN EFFORT EXTERNE

Compte tenu de l'analyse faite précédemment, la synthèse de la commande externe est donc particulièrement simple. De plus, la structure de commande est constante quelque soit la configuration de la tâche. En effet, aucune matrice de sélection n'est nécessaire à l'orthogonalité des commandes, ni aucune transformation géométrique pour la description de la tâche dans le repère de référence. Ainsi, il n'y a aucun risque d'instabilité lié aux transitions mouvements libres-mouvements contraints.

Lorsque l'asservissement de position est bien réglé, la synthèse des contrôleurs ne dépend pas de la Jacobienne, c'est à dire de la configuration géométrique du robot. Finalement, le schéma global de commande est donné par la figure 14.

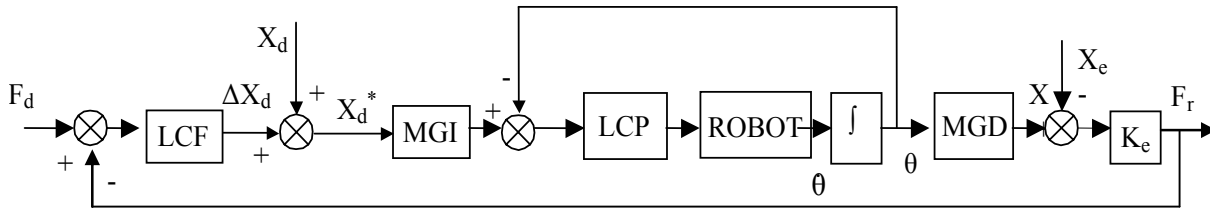


Figure 14. Structure finale adoptée de la commande externe.

IV.5. CONCLUSION

Dans ce chapitre nous avons présenté quatre structures de commande hybride force-position : la commande de Craig et Raibert, la commande de Reboulet et Robert, la commande de Khatib et enfin la commande externe.

L'analyse des schémas de ces commandes fait apparaître deux familles. La première utilise deux boucles d'asservissement parallèles. C'est le cas des structures proposées par Craig et Raibert, Reboulet et Robert et Khatib. La seconde est, quant à elle, basée sur le

principe de la hiérarchisation d'une boucle d'asservissement par rapport à l'autre. C'est le cas de la structure de commande externe.

Les structures de commande de Craig et Raibert, de Reboulet et Robert et de Khatib sont constituées de deux boucles parallèles, l'une sur l'erreur de position et l'autre sur l'erreur des efforts. Elle diffère essentiellement par la nature des grandeurs sommées provenant des deux boucles d'asservissement, et par l'enchaînement des transformations géométriques et cinématiques.

La présence de la matrice de sélection et les transformations géométriques nécessaires rend leur synthèse et leur mise en œuvre difficile. Des problèmes de stabilité interviennent également lors de commutations de structures ou dans certaines configurations du système.

La structure externe rend le contrôle des positions et des forces totalement indépendantes de la position instantanée du bras. En effet, la hiérarchisation de la boucle d'effort sur la boucle de position permet un contrôle permanent de l'effort et de la position sans aucune commutation. Sa grande flexibilité, quant à la réalisation de tâches robotique comprenant une mise en contact avec l'environnement, ouvre de nombreuses possibilités, tout particulièrement pour la coopération multi-robots.

Compte tenu, des avantages que présente la structure de commande externe, nous proposons dans le chapitre suivant d'analyser ses performances dans le cas de la commande d'un bras manipulateur redondant. Ainsi, nous allons évaluer ses performances pour différentes tâches robotisées. La synthèse des lois de commande en position et en effort est réalisée en utilisant la commande par découplage non linéaire.

Synthèse de commandes hybrides Force / Position

Introduction

De nombreuses applications nécessitent une évolution rapide du robot et une bonne précision de suivi de trajectoires simultanée de positions et des efforts d'interactions robot - environnement. Il est donc indispensable de concevoir un système de commande permettant la prise en compte de forces d'interaction durant la réalisation de la tâche [Lor 93]. Une solution à ce problème est la commande hybride force / position [Sici 01].

Ce chapitre porte donc sur la synthèse de la commande hybride force /position d'un bras manipulateur redondant basée sur la structure de commande en effort externe. Le schéma de cette structure de commande (figure 1) fait apparaître un contrôleur de position et un contrôleur d'effort [Per 20].

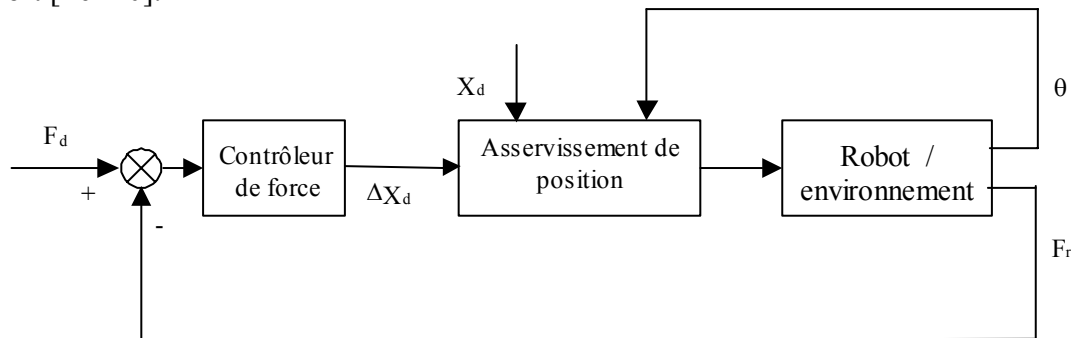


Figure 1. Structure de la commande en effort externe.

Afin de mettre en œuvre le contrôleur de position nous proposons d'utiliser différentes approches pour la commande du robot considéré soit : la commande dynamique, la commande à base des concepts de la logique floue et enfin à base des concepts de la logique neuro-floue.

L'utilisation de la commande par découplage non linéaire constitue une bonne approche compte tenu qu'elle prend en compte toutes ou une partie des forces d'interactions avec l'environnement [kha 99], [Rai 81]. Ce type de commande est aussi connu sous le nom de commande dynamique (ou « computed torque » dans la littérature) [khal.99]. Cette structure de commande assure le découplage et la linéarisation des équations du modèle dynamique. Ceci engendre une réponse uniforme quelle que soit la configuration du robot.

Cependant la difficulté de l'analyse mathématique de l'ensemble robot-environnement constitue un inconvénient majeur à l'utilisation des approches de commande conventionnelle. De plus les réglages des gains des PID sont réalisés expérimentalement.

Cet inconvénient est particulièrement critique lorsque l'on cherche à obtenir de bonnes performances en terme d'optimisation des réponses et d'adaptativité vis à vis des paramètres de la tâche. Il apparaît donc difficile dans ces conditions de concevoir une commande performante d'un système non linéaire inconnu, ce qui nous amène à opter pour des concepts de la technique de l'intelligence artificielle. Pour notre présent travail, on va utiliser les concepts

flous et neuro-flou pour la commande d'un bras manipulateur afin de suivre une trajectoire désirée.

Pour remédier à ce point, on propose de s'orienter dans une seconde étape vers les méthodes d'intelligence artificielles (IA) tel que les concepts de la logique floue et ceux de la logique neuro-floue.

Afin de bien mener notre synthèse de commande hybride force/ position, nous diviserons ce chapitre en quatre parties :

- La première partie est consacrée à la synthèse de la commande dynamique.
- La deuxième partie est consacrée à la synthèse de la commande en utilisant les concepts de la logique floue.
- La troisième partie est consacrée à la synthèse de la commande en utilisant les concepts des réseaux neuro-flous.
- La quatrième partie est consacrée à l'étude comparative des structures de commandes proposées.

Première partie

Commande hybride Force - Position utilisant l'approche de La commande dynamique

Dans cette première partie, nous présentons le principe de la commande dynamique par découplage non linéaire. Afin d'évaluer et d'analyser les performances de la commande proposée, nous l'appliquons en simulation à un bras manipulateur redondant à trois liaisons rotoïdes. Les tâches à réaliser seront des mouvements dans des espaces libres et des mouvements dans des espaces contraints.

V.I.3 Principe de la commande dynamique

Il a été montré dans le chapitre II, qu'un bras manipulateur peut être modélisé par un système d'équations différentielles du second ordre, non linéaires et couplées comme suit:

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q). \dots\dots\dots(V.1)$$

Le principe de la commande dynamique est d'utiliser cette même équation, obtenue par la méthode de Lagrange, plus connue sous le nom de modèle dynamique inverse du bras manipulateur, afin de le commander [Mez 03] [Yos 90].

Cette méthode s'appuie sur les techniques de découplage non linéaire développées par O.Khalil [Khal.99]. Elle consiste à transformer la commande du système, non linéaire et couplé, en un problème de commande d'un système linéaire et découplé en utilisant un retour d'état.

Le système décrit par l'équation (V.1) peut être linéarisé par une commande de la forme :

$$\Gamma = AW + B \dots\dots\dots(V.2)$$

La méthode de Khalil consiste à prendre : $A = \tilde{M}(q)$, $B = \tilde{C}(q, \dot{q})\dot{q} + \tilde{G}(q)$.

Où $\tilde{M}(q)$, $\tilde{C}(q, \dot{q})$ et $\tilde{G}(q)$ sont respectivement les matrices M, C et G estimées ;

Dans le cas idéal. Nous obtenons un nouveau système, dont l'entrée est W et la sortie est q . ce système est un système du second ordre, linéaire et découplé, pour lequel nous avons [Khal.99] [Mez.03] :

$$W = \ddot{q} \dots\dots\dots(V.3)$$

Donc $W(t)$ peut être considéré comme un nouveau vecteur de commande. On se ramène donc à un problème de commande de n systèmes linéaires, découplés et du second ordre. Plusieurs choix peuvent être envisagés pour le vecteur de commande W . La solution la plus couramment utilisée consiste à introduire des gains proportionnels K_p et dérivés K_v . Leurs comportements peuvent être spécifiés par un amortissement ξ et une pulsation w_n .

$$K_p = w_n^2 \quad \text{et} \quad K_v = 2\xi w_n \dots\dots\dots (V.4).$$

Dans notre étude la tâche à réaliser consiste en des suivis de trajectoires où seule la position finale est spécifiée. Dans ce cas le but à atteindre est la position q_d . Le vecteur commande W est donc donné par l'équation suivante [Las 02]:

$$W = K_p (q_d - q) + K_v \dot{q}_d \dots\dots\dots (V.5).$$

A partir des équations (V.3) et (V.5), si la modélisation est parfaite et pour des erreurs initiales nulles, l'équation du système en boucle fermée est comme suit :

$$\ddot{q} + K_v \dot{q} + K_p (q_d - q) = 0 \dots\dots\dots (V.6).$$

Le système obtenu est un système du second ordre, linéaire et découplé compte tenu des matrices diagonales K_p et K_v . Ces dernières sont choisies de sorte à imposer la dynamique désirée à $q(t)$.

L'équation 6 est une équation différentielle du deuxième ordre linéaire et découplée et est facile à résoudre. Cette dernière représente un régulateur proportionnel dérivateur (PD). Nous avons donc pu transformer le modèle de commande complexe en un modèle de commande de base pour chaque articulation indépendamment des autres [khal.99].

Cette méthode est la plus utilisée dans l'industrie vu sa simplicité et ses performances. Toutefois, elle nécessite une bonne estimation des paramètres en particulier la matrice d'inertie « M », la matrice de Coriolis « C » et le vecteur de gravité « G ». La figure 1 présente le principe de découplage non linéaire [Xav 92] :

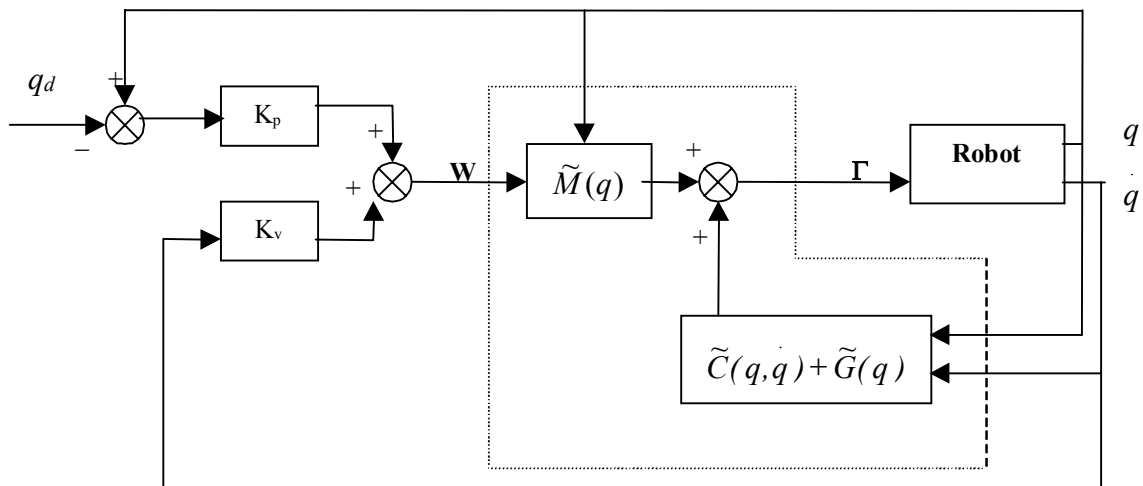


Figure 1. Découplage non linéaire.

Généralement, pour les robots industriels, les régulateurs sont de type PID, d'où l'ajout d'un terme d'intégration de l'erreur de position angulaire. Le nouveau vecteur W doit être écrit sous la forme suivante :

$$W = K_p (q_d - q) + K_v \dot{q} + K_i \int (q_d - q) dt \dots\dots\dots (V.7)$$

Cette dernière équation représente le réglage par action de type PID. Nous adapterons cette structure dans nos simulations (figure 2).

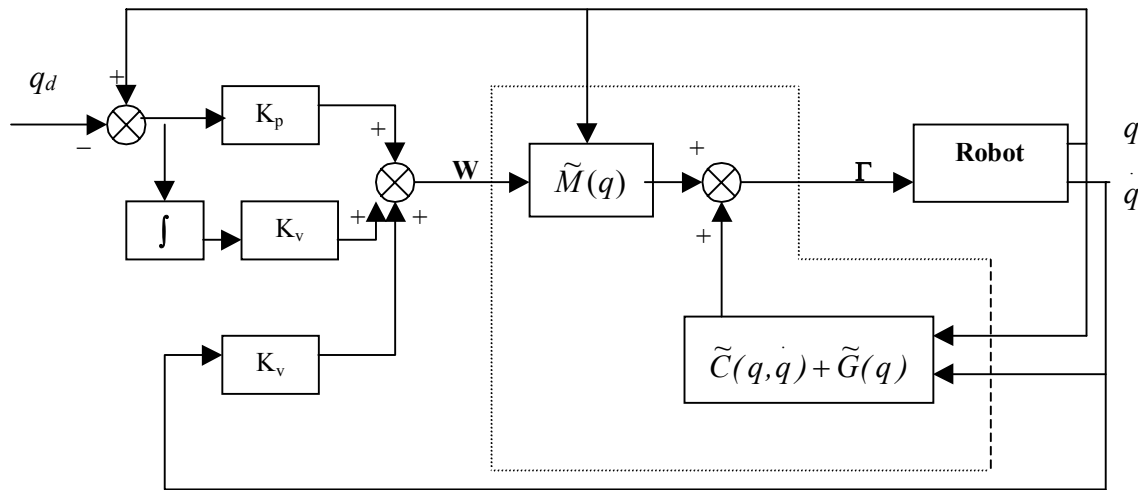


Figure 2. Découplage non linéaire par réglage PID

Nous remarquons qu'il s'agit d'une commande par retour d'état puisque les termes calculés dans l'élaboration de la consigne dépendent de l'état actuel du système dynamique. La représentation d'état du système donné par l'équation (V.1) est comme suit :

$$\begin{cases} X_2 = \dot{X}_1 \\ \dot{X}_2 = M^{-1}(X_1)[\Gamma - C(X_1, X_2)X_2 - G(X_1)] \end{cases} \dots\dots\dots (V.8)$$

$$\text{Avec: } X_1 = q \quad \text{et} \quad X_2 = \dot{q}$$

Nous adopterons cette représentation d'état pour nos simulations de la commande du système.

V.I.4 Simulation de la commande dynamique

Pour évaluer les performances de la commande proposée, nous l'appliquerons sur d'un robot redondant à trois liaisons rotoïdes. La tâche à réaliser consiste en un suivi de trajectoire, nous avons choisi la trajectoire de LEAHVY comme référence pour les différents axes du robot, compte tenu qu'elle excite toute la dynamique du robot. Cette trajectoire consiste à faire varier dans un intervalle de 5 secondes la position de [Ser 89a] :

- L'articulation 1 de -50° à 45° .
- L'articulation 2 de -135° à -85° .
- L'articulation 3 de 135° à 30° .

La figure 3 représentent les trajectoires de référence respectivement pour la première, la deuxième et la troisième articulation utilisées dans nos simulations.

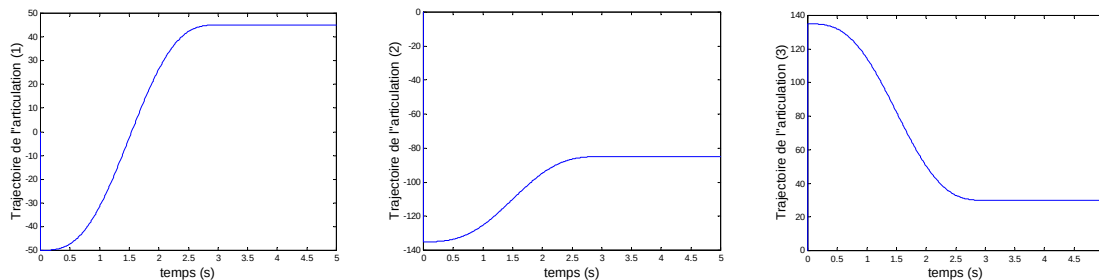


Figure 3. Trajectoires de références des trois articulations.

La figure 4 présente la trajectoire désirée de l'organe terminal dans le plan (X, Y), ainsi les trajectoires désirées suivant l'axe X et l'axe Y.

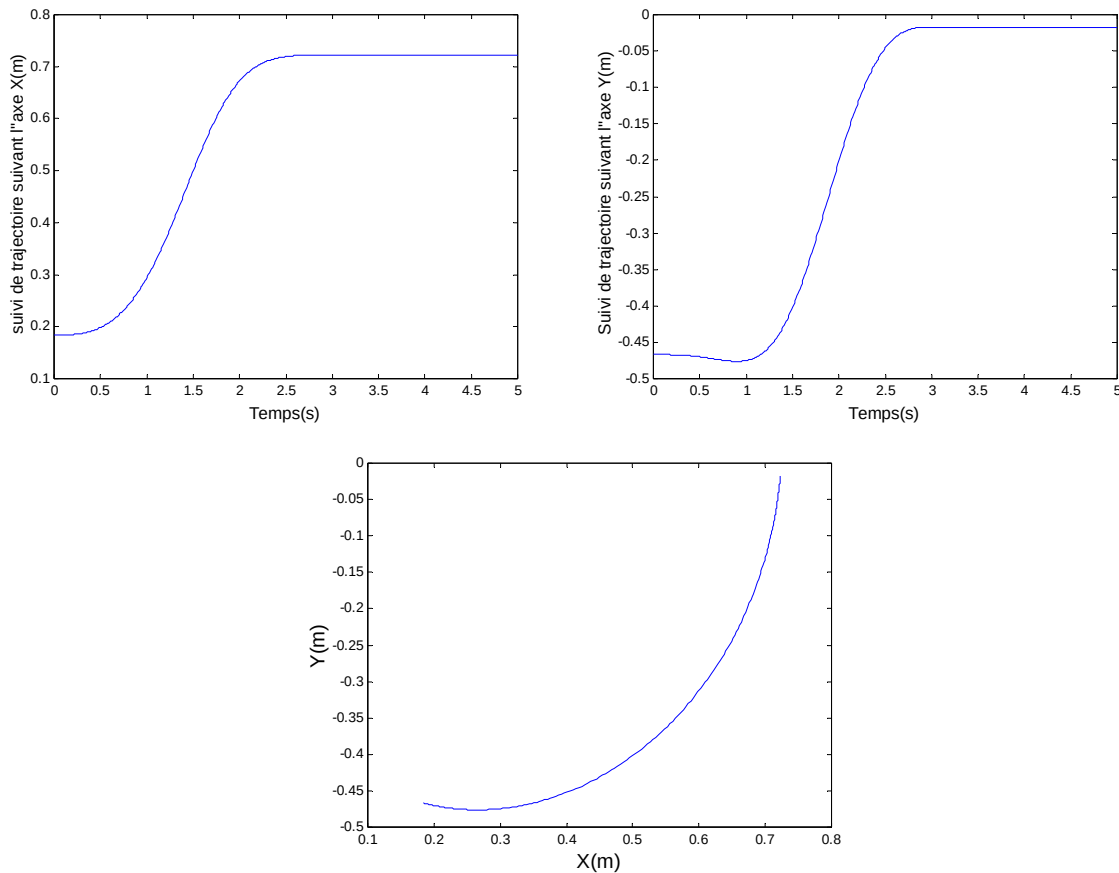


Figure 4. Trajectoires de l'organe terminal dans le plan.

Les simulations des lois de commande ont été réalisées par programmation en langage MATLAB. Nous n'utilisons aucune fonction préprogrammée de MATLAB. L'algorithme de résolution Rung-Kutta d'ordre 4 est programmé pour la résolution de l'équation différentielle du second ordre. La période d'échantillonnage utilisée dans nos simulations est égale à 0.01 secondes.

Afin d'évaluer les performances de notre système. Dans la présente étude, nous avons considéré deux configurations de mouvements possibles :

- **Les mouvements dans un espace articulaire** : Dans cette première étape, nous allons étudier le mouvement du robot dans un espace articulaire libre. Dans cet environnement, la tâche consiste à suivre une trajectoire articulaire désirée sans contrainte d'effort.
- **Les mouvements dans un espace cartésien** : Dans cette seconde étape le mouvement du robot est un suivi de trajectoire dans l'espace cartésien. En effet, le robot manipulateur devra se déplacer dans un espace libre puis rentrer en contact avec un mur et le suivre en exerçant un effort constant sur ce dernier. Dans cette partie nous allons utiliser la structure de la commande en effort externe pour assurer l'asservissement de position et d'effort.

➤ Simulation de la commande dynamique dans un espace articulaire

Nous entendons par commande en position dans l'espace articulaire toute commande permettant d'asservir, indépendamment ou de façon synchronisée, la position articulaire d'un bras manipulateur (figure 5).

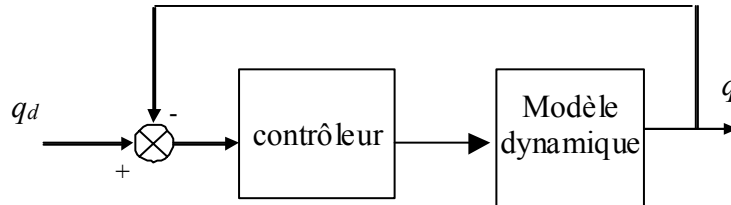


Figure 5. Asservissement de la position dans l'espace articulaire

Trois principales expérimentations sont considérées :

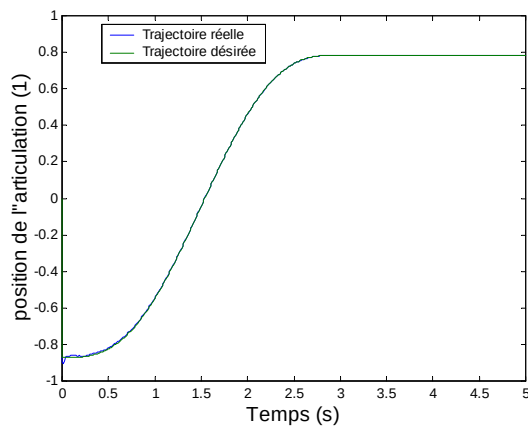
- La première consiste en l'étude du mouvement du bras manipulateur dans un espace libre sans charge.
- La seconde étudie le mouvement du même bras manipulateur avec une charge.
- Dans la dernière expérimentation, nous considérons le cas d'une défaillance sur l'un des actionneurs du bras manipulateur.

• Mouvement dans un espace libre sans charge

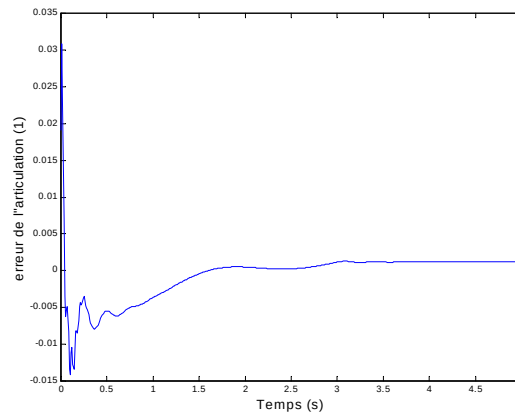
Nos simulations ont pour but l'évaluation des erreurs de suivi du robot sans charge dans un environnement de performance sans contraintes d'effort décrit par la trajectoire de LEAHVEY. Les gains des PID associés à chaque articulation assurant un bon suivi sont :

$$P = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 3.75 & 0 \\ 0 & 0 & 2 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 6 \end{bmatrix} N/m.s \quad D = \begin{bmatrix} 430 & 0 & 0 \\ 0 & 350 & 0 \\ 0 & 0 & 200 \end{bmatrix} m.s/N$$

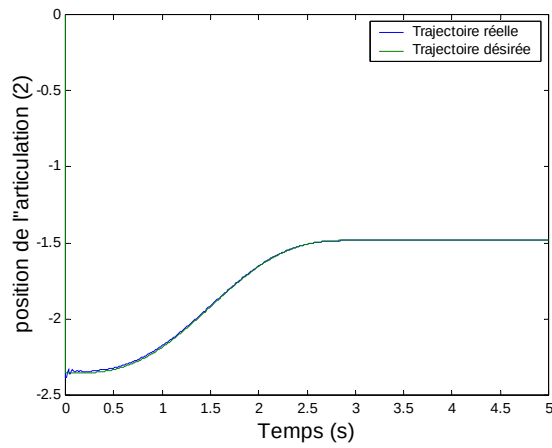
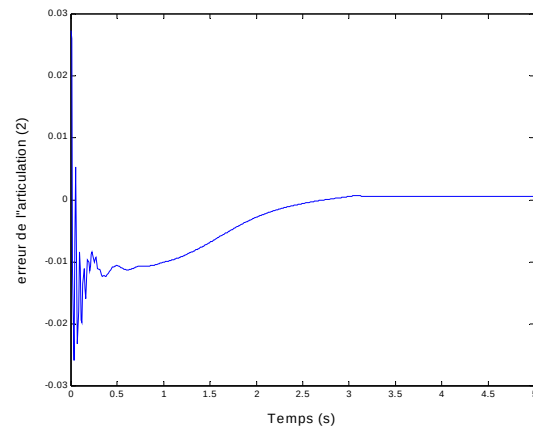
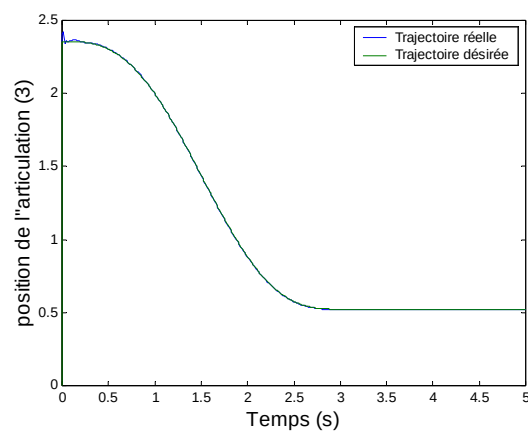
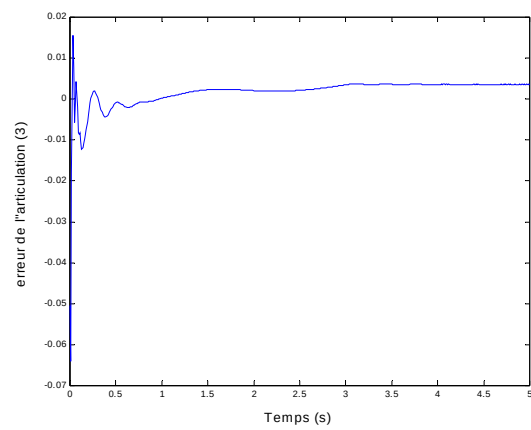
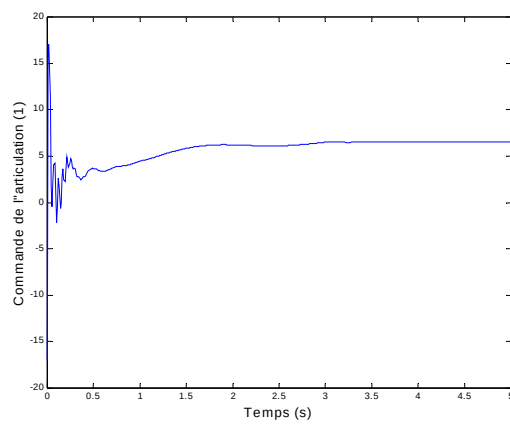
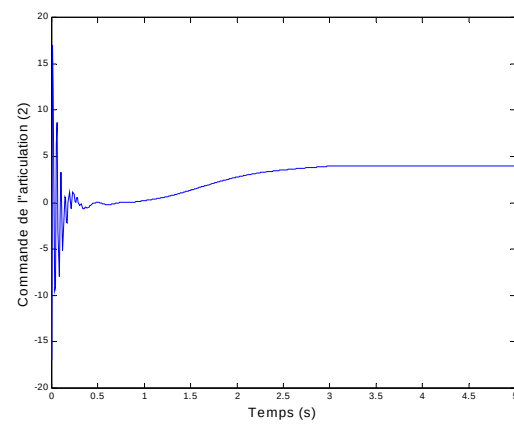
Les résultats obtenus pour cet essai à vide sont consignés sur la figure 6, ces résultats font apparaître un bon suivi de trajectoire.

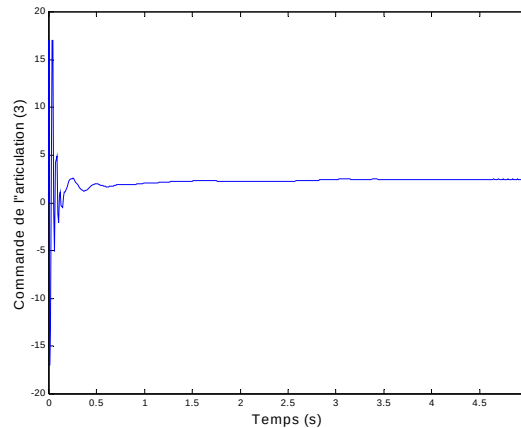


- a1 -



- b1 -

**- a2 -****- b2 -****- a3 -****- b3 -****- c1 -****- c2 -**



- c3 -

Figure 6. Mouvement dans un espace libre sans charge.

- $a1$, $a2$ et $a3$ Trajectoires respectivement de l'articulation 1,2 et 3.
- $b1$, $b2$ et $b3$ Erreurs de suivis respectivement de l'articulation 1,2 et 3.
- $c1$, $c2$ et $c3$ Couples appliqués respectivement à l'articulation 1,2 et 3.

Les figures 6.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement des articulations 1, 2 et 3. Ces courbes montrent un bon suivi de trajectoires. En effet, les courbes des erreurs de suivis articulaires (figures 6.b1, b2 et b3) montrent que les erreurs initiales sont de 0.0191 rad pour l'articulation 1, 0.0274 rad pour la deuxième et -0.0504 rad pour la troisième. Ces erreurs sont dues essentiellement à l'inertie des bras et au fait qu'initialement les positions des bras ne coïncident pas avec les positions initiales désirées. En régime permanent les erreurs sont de 0.0012 rad pour l'articulation 1, 0.0006 rad pour la deuxième et 0.0036 rad pour la troisième.

Les figures 6.c1, c2 et c3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant considéré. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit: 17 N.m pour la première articulation, 16,4 N.m pour la deuxième et de 15.5 N.m pour la troisième. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.50 N.m pour la première articulation, 3.94 N.m pour la deuxième et 2.48 N.m pour la troisième. Nous pouvons aussi constater qu'il existe une relation quasi-linéaire entre les erreurs articulaires et les vecteurs de commandes.

Ces premiers résultats nous permettent de valider l'approche de la commande proposée. Afin de tester la robustesse de la commande proposée nous avons effectué une seconde expérience qui consiste à étudier le mouvement du même bras manipulateur avec une charge.

• Mouvement dans un espace libre avec charge

Afin de tester la robustesse de la commande vis à vis de la variation de la charge, nous considérons dans cette deuxième expérience le mouvement non contraint en effort d'un bras manipulateurs redondant soumis à une charge variant instantanément de 1.5 Kg à 0 Kg.

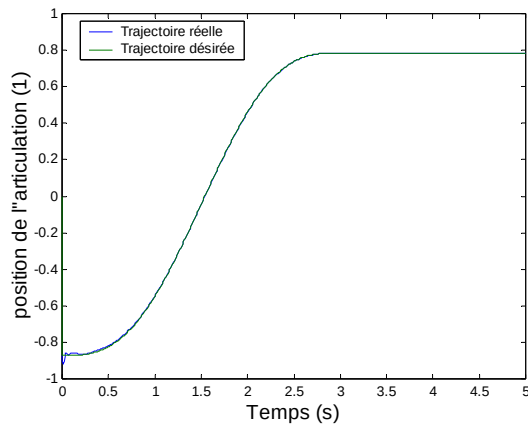
Remarque: La charge maximale que peut porter un bras manipulateur ne peut dépasser le tiers de son poids. Dans notre cas elle est de 1.5 Kg.

Les gains des PID associées à chaque articulation assurant un bon suivi sont :

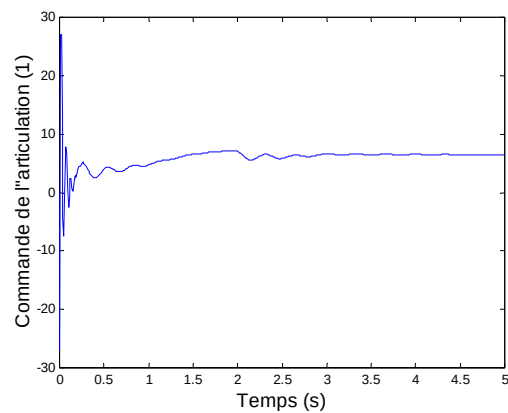
$$P = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 3.75 & 0 \\ 0 & 0 & 2 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 4 \end{bmatrix} N/m.s \quad D = \begin{bmatrix} 430 & 0 & 0 \\ 0 & 350 & 0 \\ 0 & 0 & 130 \end{bmatrix} m.s/N$$

Les figures 7.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement de la première, la deuxième et la troisième articulation. Ces courbes montrent un suivi de trajectoires satisfaisant. En effet, Les courbes des erreurs de suivis articulaires (figures 7.b1, b2 et b3) montrent que les erreurs initiales sont de 0.04 rad pour la première articulation, 0.039 rad pour la deuxième et -0.095 rad pour la troisième.

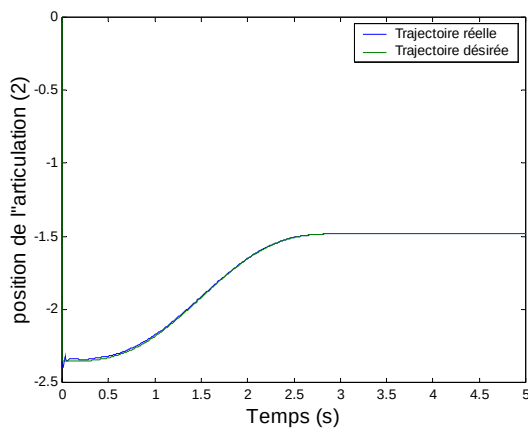
Ces résultats sont sensiblement identiques à ceux obtenus lors du mouvement du bras manipulateur sans charge sauf pour la première articulation où l'erreur articulaire est plus grande (0.03 rad au lieu de 0.017 rad). Ceci s'explique par le fait qu'initialement les positions des bras ne coïncident pas avec les positions initiales désirées et que l'effet de la charge est plus grand sur cette articulation. Les erreurs articulaires sont essentiellement dûes à l'inertie du bras et à l'erreur de positionnement initial. En régime permanent, à l'instant « $t = 2s$ », les courbes des erreurs articulaires font apparaître de faibles oscillations. Les erreurs articulaires sont maximales à cet instant soit : $2.23 \cdot 10^{-3}$ rad pour la première articulation, $-1.2 \cdot 10^{-3}$ rad pour la deuxième et $5.2 \cdot 10^{-3}$ rad pour la troisième. Ceci s'explique par l'annulation de la charge qui agit sur le système comme une perturbation. Ces erreurs articulaires se stabilisent à l'instant « $t = 3s$ » aux valeurs: $1.2 \cdot 10^{-3}$ rad pour la première articulation, $0.6 \cdot 10^{-3}$ rad pour la deuxième et $3.6 \cdot 10^{-3}$ rad pour la troisième.



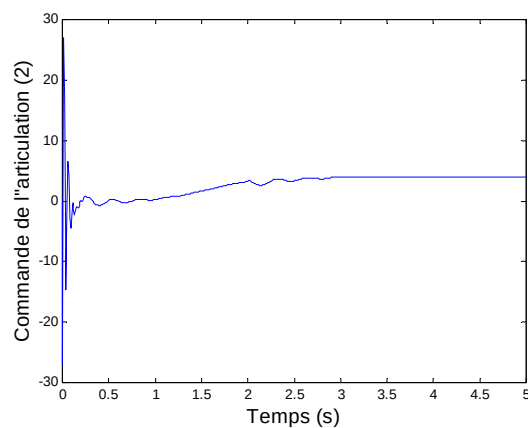
- a1 -



- c1 -



- a2 -



- c2 -

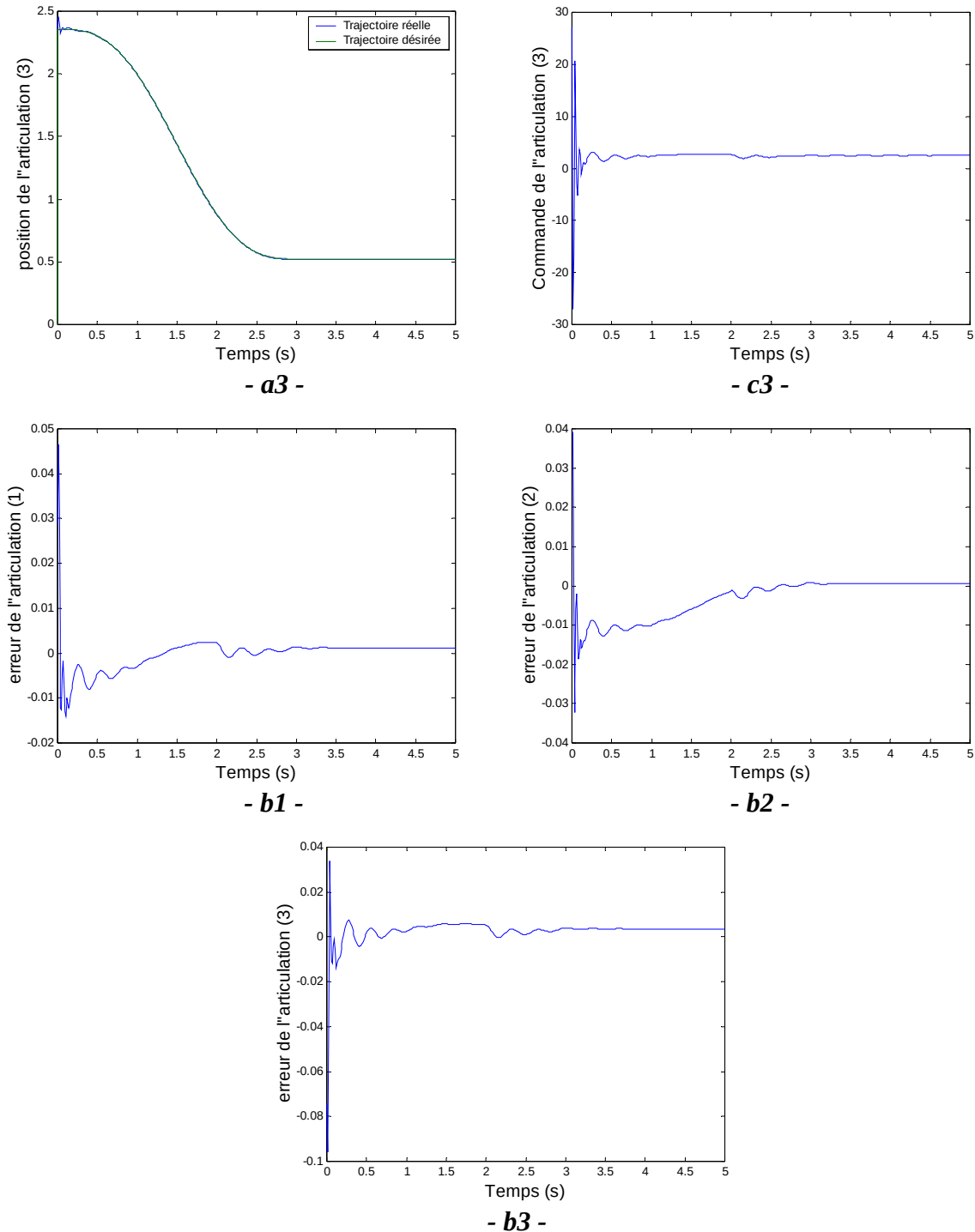


Figure 7. Mouvement dans l'espace libre du bras manipulateur redondant soumis à une charge initiale de 1.5 Kg variant instantanément ($t = 2s$) à 0 Kg

- $a1$, $a2$ et $a3$ Trajectoires respectivement de l'articulation 1,2 et 3.
- $b1$, $b2$ et $b3$ Erreurs de suivis respectivement de l'articulation 1,2 et 3
- $c1$, $c2$ et $c3$ Couples appliqués respectivement à l'articulation 1,2 et 3.

Les figures 7.c1, c2 et c3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant utilisé. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit: 28.6 N.m pour la première articulation, 26.3 N.m pour la deuxième et de 27 N.m pour la troisième. Ces couples sont sensiblement plus élevés que ceux

obtenus lors d'un mouvement libre sans charge. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras avec la charge (initialement la charge portée par l'effecteur est de 1.5 Kg). En régime permanent, à l'instant « $t = 3 \text{ s}$ », les courbes représentant les couples (figures 8.c1, c2 et c3) font apparaître de faibles oscillations. Les valeurs maximales des couples à cet instant sont respectivement: 7.5 N.m pour la première articulation, 4.6 N.m pour la deuxième et 2.95 N.m pour la troisième. Ceci s'explique par l'annulation de la charge qui agit sur le système comme une perturbation. Les vecteurs commandes finissent par se stabiliser à l'instant « $t = 3 \text{ s}$ » à la valeur de 6.50 N.m pour la première articulation, 3.95 N.m pour la deuxième et 2.46 N.m pour la troisième.

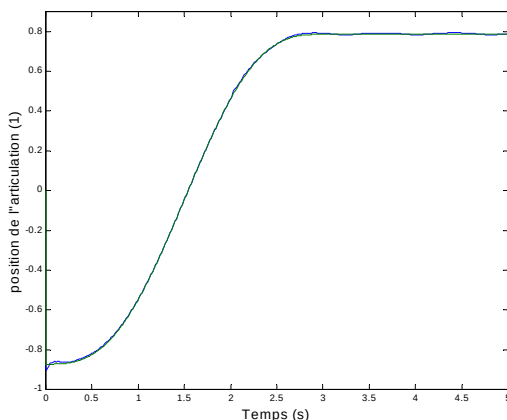
Les résultats obtenus montrent la robustesse de la commande vis à vis de la variation de la charge. En effet, la commande proposée permet de compenser les perturbations engendrées par la chute de la charge de 1.5 Kg pendant le mouvement.

Les deux expériences précédentes, montrent la validité de la commande proposée lors d'un mouvement non contraint en effort d'un bras manipulateur redondant avec ou sans charge. Nous nous intéresserons dans la suite de notre travail au comportement de la structure de commande proposée au cas où une défaillance apparaîtrait sur l'un des régulateurs (ou actionneurs) des bras du robot.

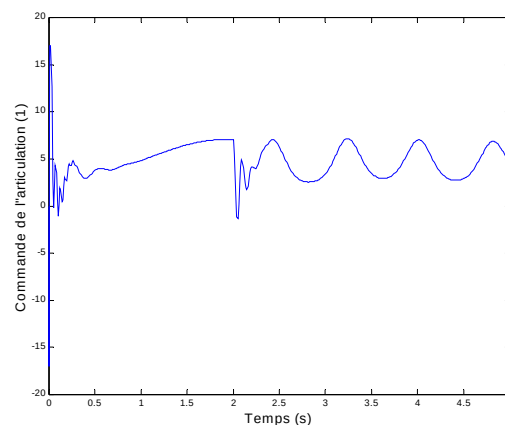
• Défaillance de l'un des régulateurs (ou actionneur) des bras du robot manipulateur redondant

Ce test consiste à simuler une défaillance d'un des trois régulateurs qui commande le robot pendant son mouvement. Ceci dit, nous introduisons à l'instant « $t = 2 \text{ s}$ » une défaillance sur le troisième régulateur. Les gains des PID associés à chaque articulation assurant un bon suivi sont :

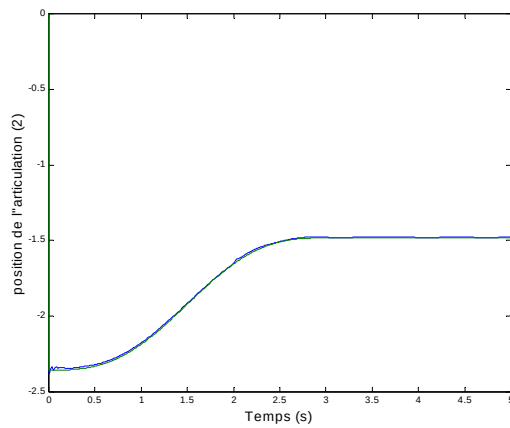
$$P = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 3.75 & 0 \\ 0 & 0 & 2 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 5 \end{bmatrix} N/m.s, \quad D = \begin{bmatrix} 430 & 0 & 0 \\ 0 & 350 & 0 \\ 0 & 0 & 100 \end{bmatrix} m.s/N$$



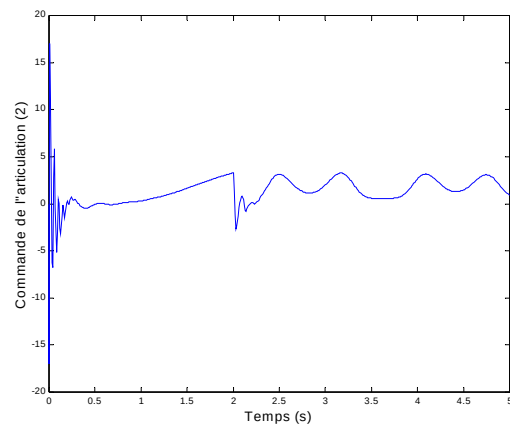
- a1 -



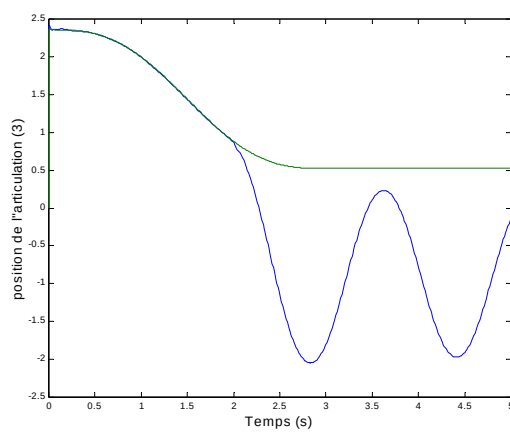
- b1 -



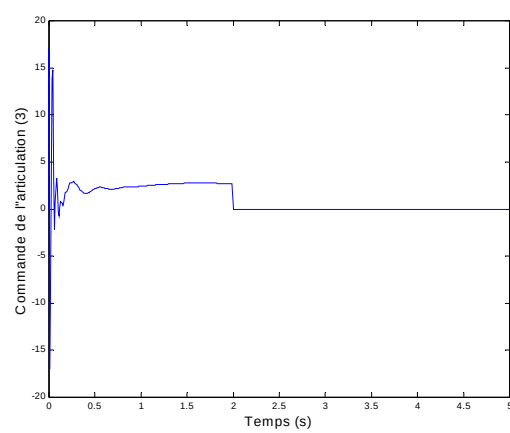
- a2 -



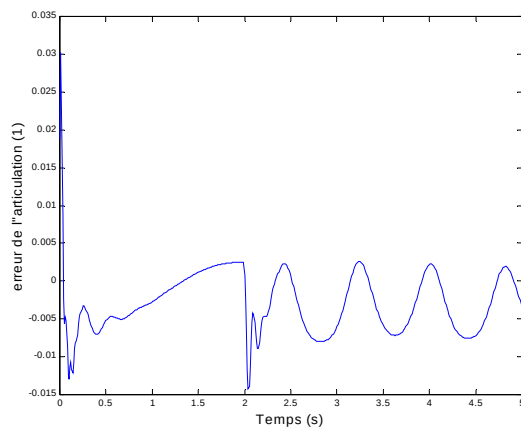
- b2 -



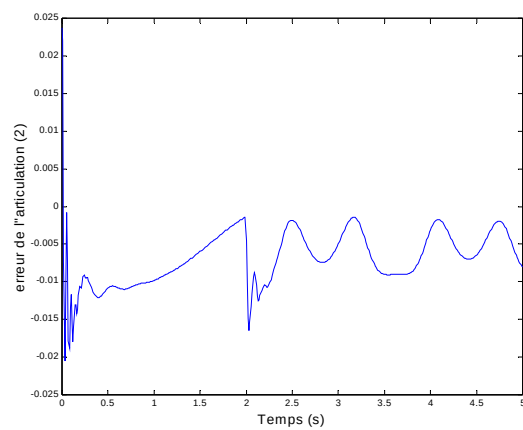
- a3 -



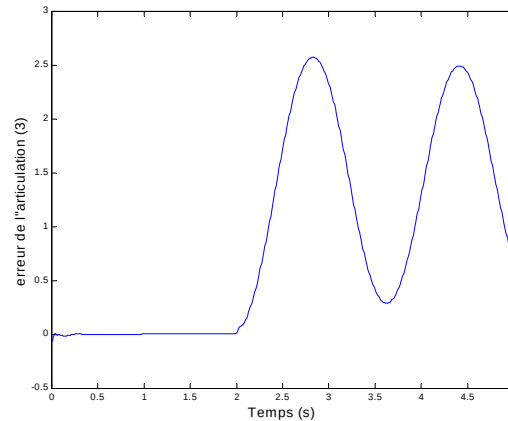
- b3 -



- c1 -



- c2 -



- c3 -

Figure 8. Mouvement dans l'espace libre du bras manipulateur redondant sans charge cas d'une rupture de la commande sur la troisième articulation

- $a1$, $a2$ et $a3$ Trajectoires respectivement de l'articulation 1,2 et 3.
- $b1$, $b2$ et $b3$ Erreurs de suivis respectivement de l'articulation 1,2 et 3.
- $c1$, $c2$ et $c3$ Couples appliqués respectivement à l'articulation 1,2 et 3.

Les figures 8.a1, a2 et a3 montrent que le suivi de trajectoires est satisfaisant pour la première et la deuxième articulation. Dans le cas de la troisième articulation une divergence apparaît à partir de l'instant « $t = 2s$ ». En effet, à cet instant une défaillance de ce régulateur est introduite. Les courbes des erreurs de suivi articulaire (figures 8.b1, b2 et b3) montrent que les erreurs initiales sont: 0.017 rad pour la première articulation, 0.028 rad pour la deuxième et -0.05 rad pour la troisième. Ces résultats sont sensiblement identiques à ceux obtenus lors du mouvement du bras manipulateurs sans charge. Ces erreurs articulaires sont dues essentiellement à l'inertie du bras et aux erreurs initiales de positionnement des bras du manipulateur. A l'instant « $t = 2s$ », les courbes des erreurs articulaires font apparaître un pic d'amplitude de - 0.014 rad pour la première articulation suivi de faibles oscillations, -0.0165 rad pour la deuxième suivi de faibles oscillations. Ces dernières restent oscillantes autour de $-2 \cdot 10^{-3}$ rad pour la première articulation et $-4 \cdot 10^{-3}$ rad pour la deuxième articulation. Dans le cas de la troisième articulation nous observons des oscillations croissantes d'où la divergence du système. Ces résultats s'expliquent en l'occurrence par une défaillance sur ce régulateur. En effet, une défaillance sur une articulation entraîne des contraintes sur les autres articulations en agissant comme une perturbation. De ce fait, les deux régulateurs seront plus sollicités à cet instant et essaient de compenser l'effet de cette perturbation.

Les figures 8.c1, c2 et c3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit: 17 N.m pour la première articulation, 16.4 N.m pour la deuxième et de 15.5 N.m pour la troisième. Ces couples sont identiques à ceux obtenus lors d'un mouvement libre sans charge. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. A l'instant « $t = 2s$ », les courbes représentant les couples (figures 8.c1, c2) font apparaître un pic d'amplitude - 2 N.m et -3 N.m suivi d'oscillations respectivement sur la première et la deuxième articulation. Dans le cas de la troisième articulation la figure 8.c3, à partir de l'instant « $t = 2s$ » le vecteur commande s'annule. Ceci est du à la défaillance du régulateur (ou actionneur) lié à cette articulation.

Les résultats de simulation obtenus (figure 8) montrent que malgré la divergence de la troisième articulation causée par la défaillance introduite sur son régulateur, les autres articulations suivent leurs trajectoires de référence. Ceci illustre l'avantage de la structure de commande décentralisée choisie par rapport à la structure de commande centralisée.

Conclusion 1

Les résultats obtenus dans ces premières simulations montrent un bon suivi de la trajectoire articulaire avec des erreurs très faibles en régime permanent, ainsi que la robustesse de la commande vis à vis de la variation de la charge. De plus, elles illustrent l'avantage de la structure décentralisée de commande proposée. Ainsi l'approche de commande proposée est validée.

➤ Simulation de la commande dynamique dans l'espace cartésien

Généralement, la spécification de la tâche à réaliser est donnée dans l'espace opérationnel. Le robot manipulateur doit se déplacer dans cet environnement grâce à des commandes opérant sur les variables articulaires. Ainsi, le bras manipulateur se déplace vers une position désirée, son contrôle est exclusivement assuré par un asservissement défini dans le repère tâche. Dans ce cas des transformations géométriques sont nécessaires et elles sont internes à la boucle d'asservissement afin de permettre le passage d'un espace à un autre. Le passage d'un espace à un autre est donc très important et exige l'évaluation des modèles géométriques direct et inverse du robot. Ce dernier est souvent très difficile à obtenir compte tenu de la non-linéarité du système, les singularités et l'infinité de solutions existantes. La résolution à ce problème a fait l'objet du chapitre V.

Afin d'évaluer les performances de notre structure de commande, deux principales expérimentations sont considérées :

- La première consiste en l'étude du mouvement du bras manipulateur redondant sans charge dans un espace cartésien libre.
- La seconde considère le mouvement d'un bras manipulateur redondant dans un espace cartésien libre puis contraint. En effet, le robot manipulateur redondant devra se déplacer dans un espace libre puis rentrer en contact avec un environnement et il va le suivre en maintenant sur sa surface un effort constant. Dans cette partie nous allons utiliser la structure de la commande en effort externe pour assurer l'asservissement simultané de position et d'effort.

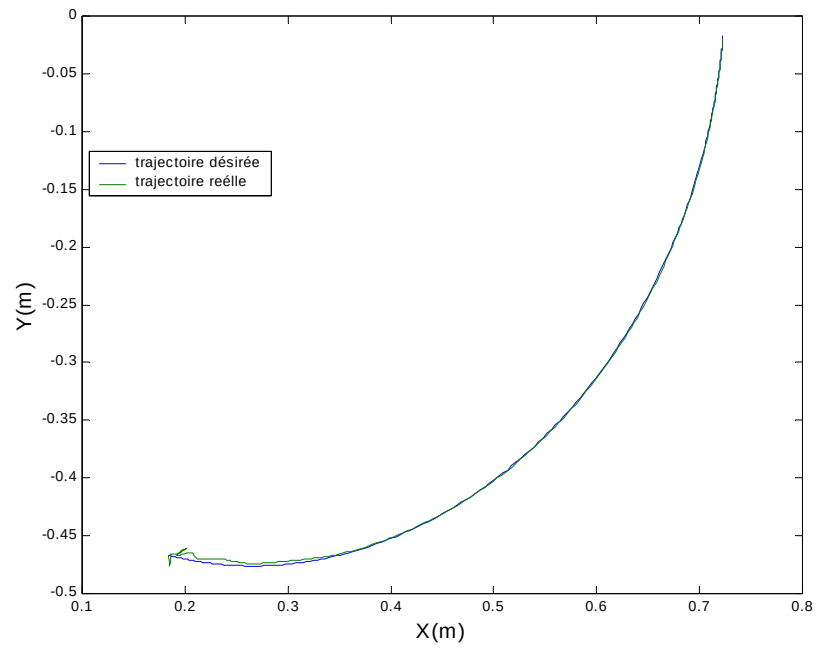
• Mouvement dans un espace cartésien libre sans charge

Le but de ces premières simulations est d'évaluer les erreurs de poursuite du robot manipulateur considéré lors d'un mouvement dans un espace libre sans contrainte d'effort.

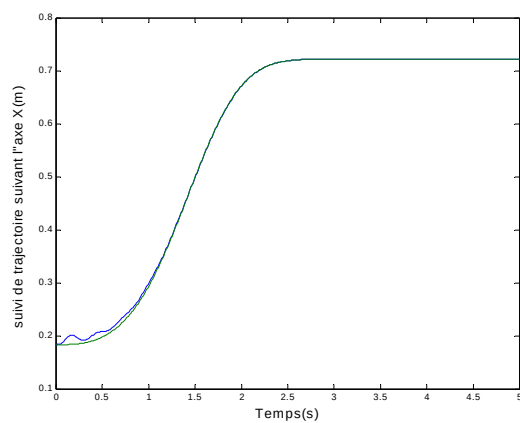
Les gains des PID associées à chaque articulation assurant un bon suivi sont :

$$P = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 4 \end{bmatrix} N/m.s \quad D = \begin{bmatrix} 260 & 0 & 0 \\ 0 & 350 & 0 \\ 0 & 0 & 200 \end{bmatrix} m.s / N$$

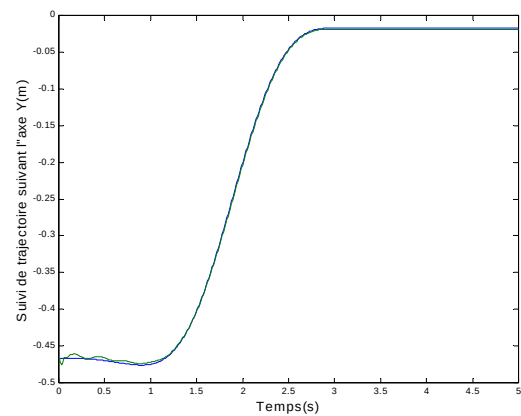
Les résultats obtenus pour cet essai à vide sont consignés sur la figure 9.



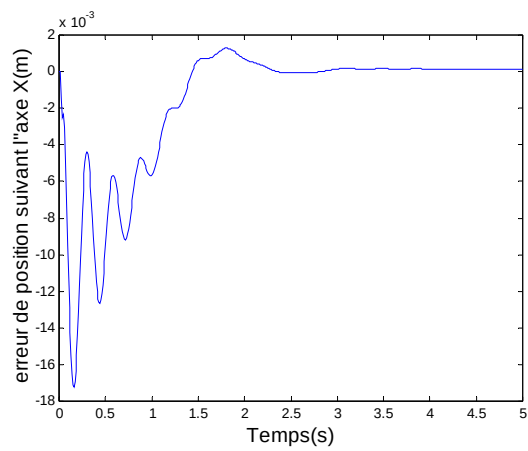
- a -



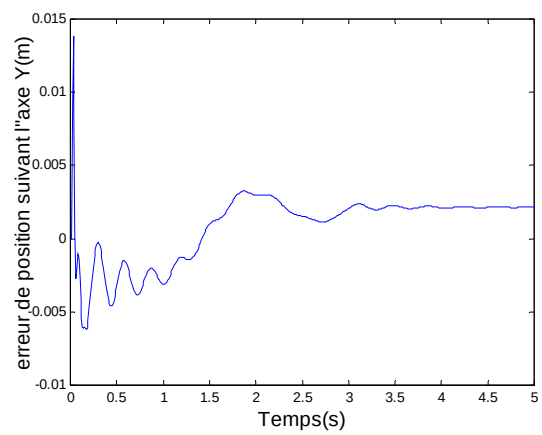
- b1 -



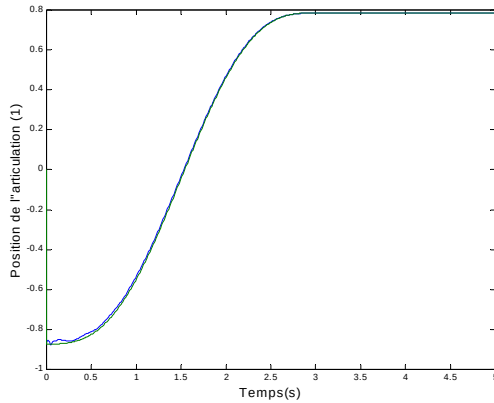
- b2 -



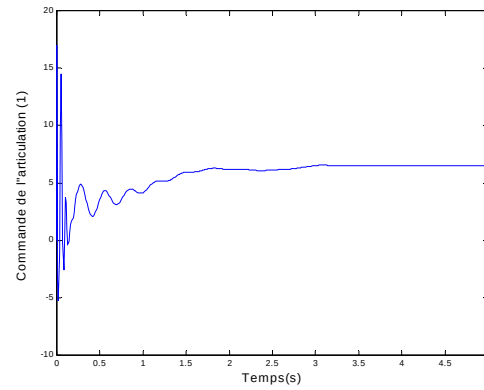
- c1 -



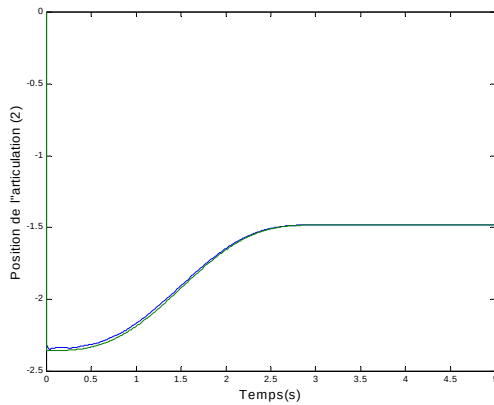
- c2 -



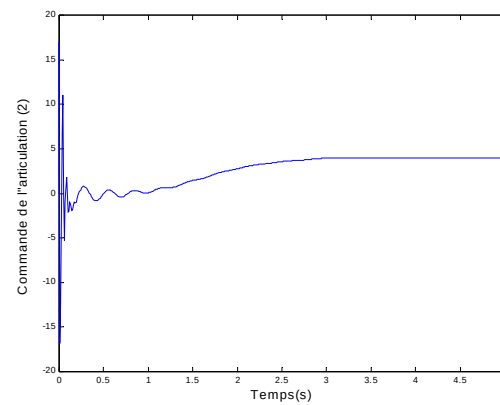
- d1 -



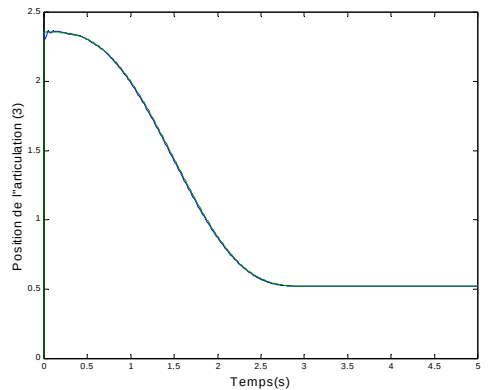
- e1 -



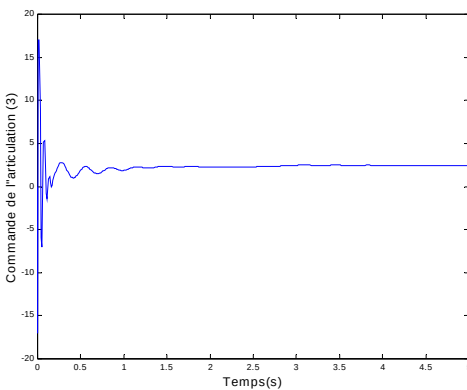
- d2 -



- e2 -



- d3 -



- e3 -

Figure 9. Mouvement dans l'espace libre cartésien du bras manipulateur redondant sans charge.

- a Trajectoire de l'organe terminal dans le plan (X, Y)
- b1 et b2 Trajectoires suivant l'axe X et l'axe Y
- c1 et c2 Erreur de suivi selon les axes X et Y
- d1, d2 et d3 Trajectoires respectivement des articulations 1, 2 et 3
- e1, e2 et e3 Couple appliqué respectivement aux articulations 1, 2 et 3

La figure 9.a illustre la trajectoire réelle et désirée de l'organe terminal dans l'espace opérationnel. Les courbes obtenues font apparaître un bon suivi de trajectoire. Les figures 9.b1 et b2 illustrent les trajectoires réelles et désirées de l'effecteur dans l'espace tâche

respectivement suivant l'axe X et suivant l'axe Y. Ces courbes montrent aussi que le suivi de trajectoire est de bonne qualité. En effet, dans les courbes des erreurs de suivi (figures 9.c1 et c2) nous remarquons que l'erreur initiale est égale à -0.0015 m suivant l'axe X et à $9.2 \cdot 10^{-3}$ m suivant l'axe Y. Ces erreurs sont dues essentiellement à l'inertie des bras et à l'erreur initiale de positionnement du robot. En régime permanent les erreurs sont faibles ($1.4 \cdot 10^{-4}$ m suivant l'axe X, $2.3 \cdot 10^{-3}$ m suivant l'axe Y).

Les figures 9.d1, d2 et d3 donnent les trajectoires réelles et désirées respectivement des articulations 1,2 et 3. Ces courbes montrent un bon suivi de trajectoire.

Les figures 9.e1, e2 et e3 représentent les couples appliqués aux différentes articulations du bras manipulateur considéré. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit: 17 N.m pour la première articulation, 16,4 N.m pour la deuxième et 15.5 N.m pour la troisième articulation. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.50 N.m pour la première articulation, 3.94 N.m pour la deuxième articulation et 2.48 N.m pour la troisième articulation.

- **Mouvement dans un espace cartésien libre puis contraint**

Nous considérons dans cette deuxième expérience un bras manipulateur redondant se déplaçant dans un espace libre puis rentrant en contact avec un environnement de raideur « K_e » donnée.

Trois tâches différentes ont été attribuées au robot considéré :

- a- Suivi d'un contour rectiligne en maintenant un effort constant.
- b- Suivi d'un contour curviligne en maintenant un effort constant.
- c- Prise et transport d'un objet d'un point « A » à un point « B ».

Nos simulations ont pour but l'évaluation des erreurs de poursuite du robot dans l'environnement spécifié. les gains de normalisation et les gains de commandes assurant une dynamique globale satisfaisante sont obtenus expérimentalement.

- a. Suivi d'un contour rectiligne en maintenant un effort constant**

Cette tâche (figure 10) fait apparaître une mise en contact et un suivi de contour. En effet, le robot se déplace initialement dans un espace libre selon une trajectoire désirée puis il rentre en contact avec un environnement de raideur K_e ($K_e = 500$ N/m).

Pour se faire à l'instant « $t = 1.75s$ » on simule la présence d'un mur situé à la position $X = 0.6m$, avec lequel il rentre en contact en maintenant sur sa surface un effort normal de 50N.

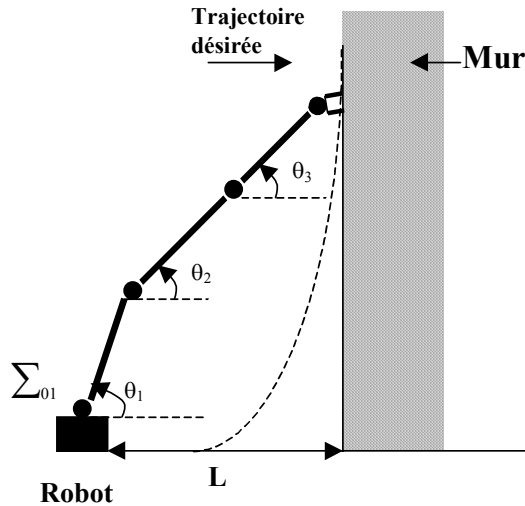


Figure 10. Suivi d'un contour rectiligne en maintenant un effort constant..

Lors du contact avec l'environnement extérieur, les déformations de ce dernier et du capteur, créent un effort que l'on suppose pour notre présente étude proportionnel au déplacement :

$$F = K_e(X - X_e) \quad \text{si } X > X_e \dots\dots\dots (V.9)$$

Où : $X - X_e$ est l'écart entre la position du point terminal et la position de l'environnement sur une normale au contour dans un repère cartésien.

L'effort externe est orthogonal avec la position absolue, il paraît nécessaire de réaliser un asservissement global décrit dans le chapitre IV. La structure de commande utilisée est :

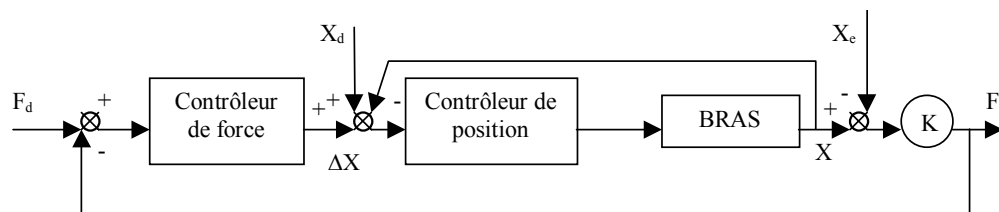


Figure 11. Structure de commande en effort externe

Les gains des PID associées à chaque articulation assurant un bon suivi sont :

$$P = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix} \text{ N/m}, \quad I = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 4 \end{bmatrix} \text{ N/m.s}, \quad D = \begin{bmatrix} 260 & 0 & 0 \\ 0 & 250 & 0 \\ 0 & 0 & 200 \end{bmatrix} \text{ m.s/N},$$

La réponse en position du système en boucle fermée est représentée par la figure 12. Dans l'intervalle du temps $[0s, 1.75s]$, le robot manipulateur se déplace dans un espace libre. Dès que l'effecteur entre en contact avec l'environnement (mur) à l'instant « $t=1.75s$ », il suit sa surface en se déplaçant suivant l'axe Y. Cette première figure fait apparaître un bon suivi de trajectoire.

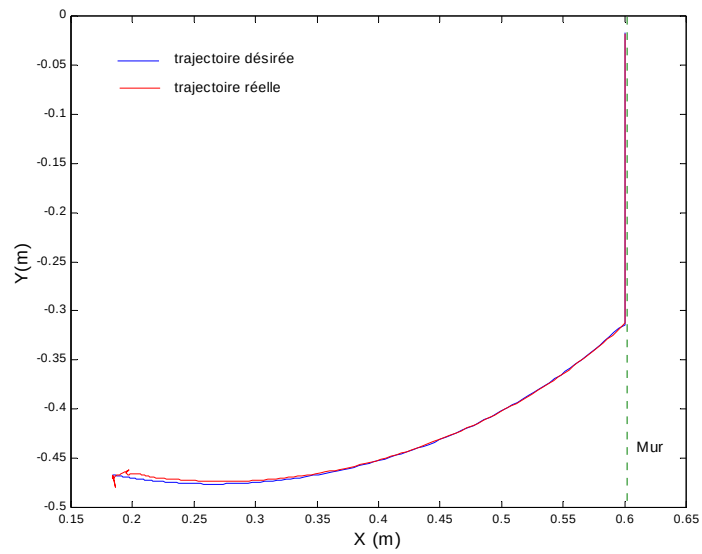
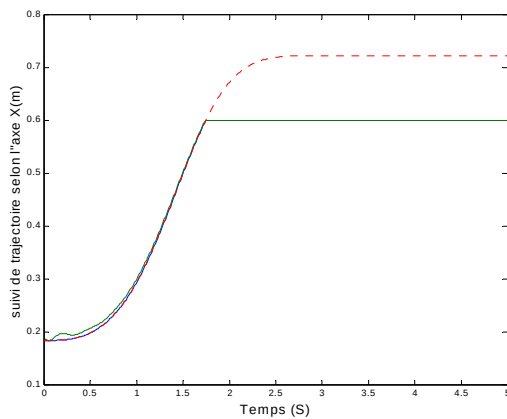
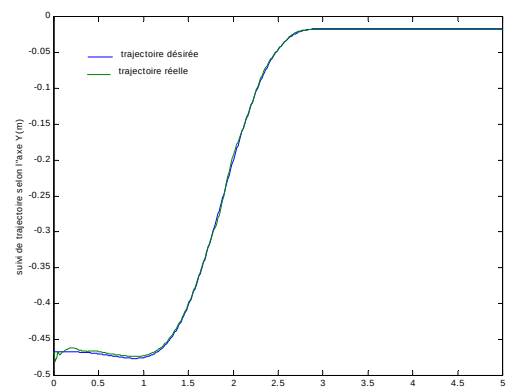


Figure 12. Suivi de contour rectiligne dans le plan (X, Y).

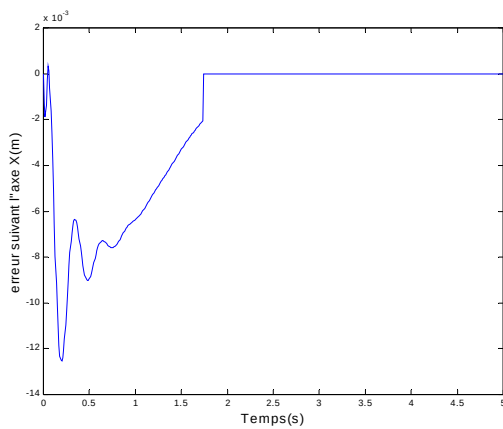
Les figures 13.a1 et a2 illustrent les trajectoires réelles et désirées respectivement suivant l'axe X et suivant l'axe Y.



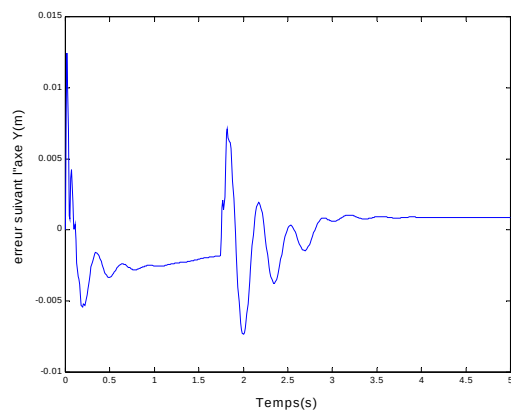
- a1 -



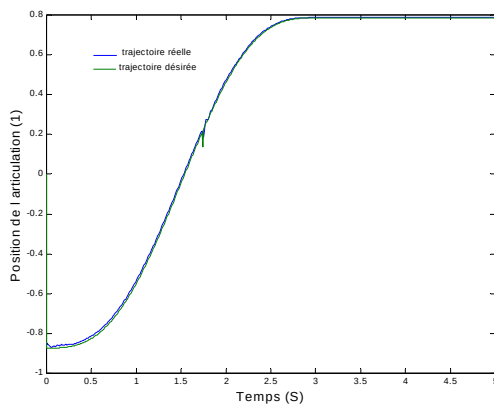
- a2 -



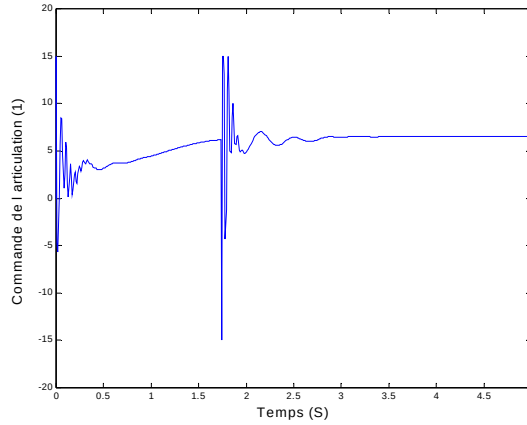
- b1 -



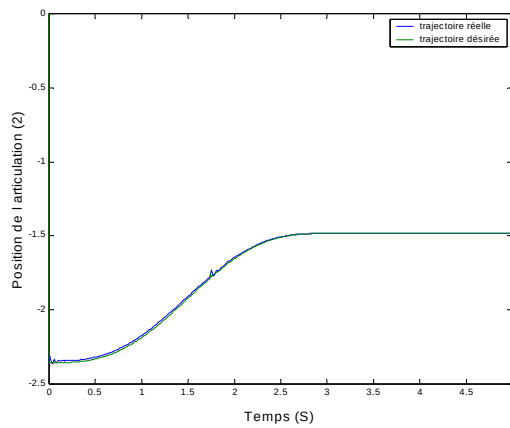
- b2 -



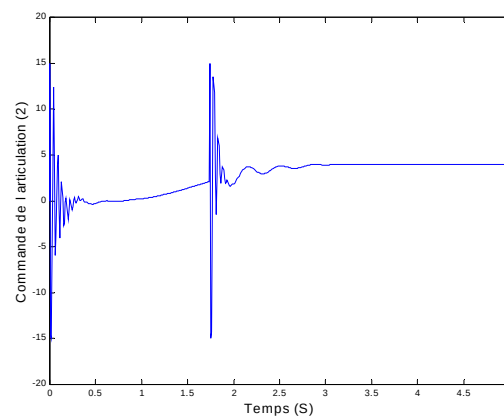
- d1 -



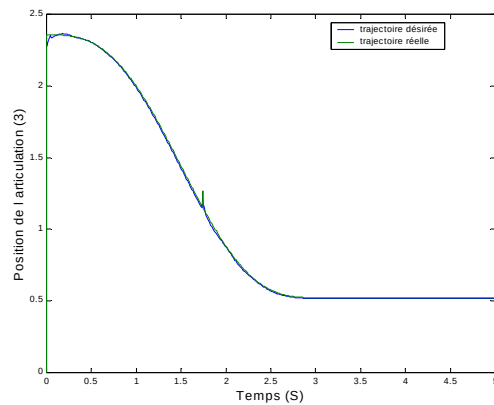
- e1 -



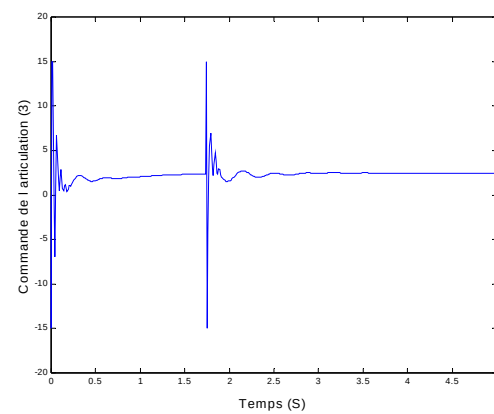
- d2 -



- e2 -



- d3 -



- e3 -

Figure 13. Suivi d'un contour rectiligne en maintenant un effort constant

- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- d1, d2 et d3 Trajectoires respectivement des articulations 1, 2 et 3
- e1, e2 et e3 Couple appliqué respectivement aux articulations 1, 2 et 3

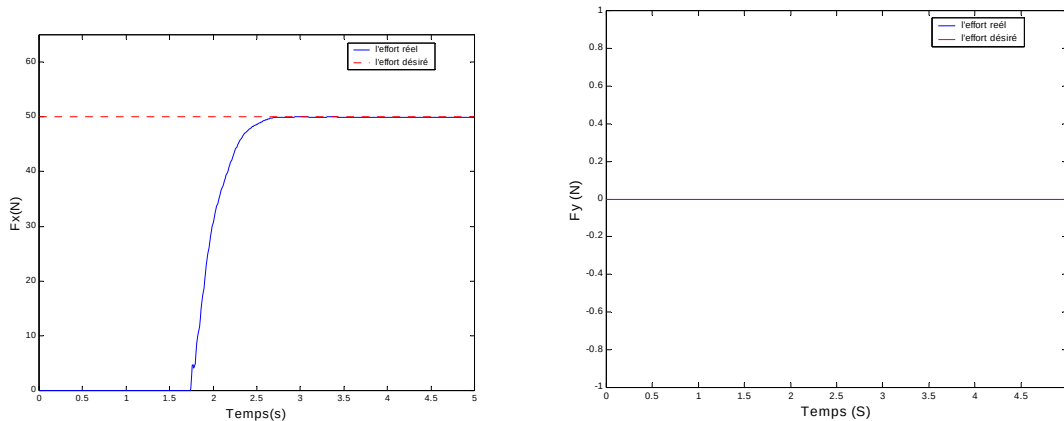
La poursuite suivant l'axe X, est satisfaisante. En effet, la figure 13.b1 montre initialement que l'erreur dans l'espace cartésien est $0.4 \cdot 10^{-3}$ m selon l'axe X. Cette erreur est due à la position initiale de l'effecteur. Une fois que l'effecteur est en contact avec l'environnement ($t=1.75$ s), cette erreur décroît jusqu'à atteindre $-1.8 \cdot 10^{-6}$ m. Ceci s'explique

par le fait que pour assurer un bon suivi de consignes, l'effecteur prend une configuration lui permettant de maintenir un effort désiré normal à la surface de l'obstacle, tout en garantissant un déplacement suivant les directions non contraintes. La figure 13.b2 montre que le suivi suivant l'axe Y est de bonne qualité. Toutefois, un pic d'amplitude 7.10^{-3} m à $t=1.75$ s apparaît puis l'erreur se stabilise à la valeur 8.10^{-4} m à partir de l'instant $t = 3$ s. Ceci peut s'expliquer par la mise en contact à $t=1.75$ s du bras manipulateur avec l'environnement.

Les figures 13.d1, d2 et d3 présentent respectivement le suivi de la trajectoire de la première, deuxième et la troisième articulation. Ces courbes montrent un bon suivi de trajectoires au cours de tout le mouvement. Toutefois, des pics apparaissent à l'instant $t = 1.75$ s. Ceci s'explique, par l'occurrence du contact.

Les figures 13.e1, e2 et e3 présentent les couples appliqués aux différentes articulations du bras manipulateurs redondant considéré. Au démarrage les valeurs des couples sont élevées soit 17 N.m, 16.4 N.m et 15.5 N.m respectivement pour la première, deuxième et la troisième articulation. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. Lors du contact avec l'environnement, quelques oscillations apparaissent au niveau des couples de commande. Les valeurs maximales des oscillations sont 16.2 N.m, 15.4 N.m et 14.8 N.m respectivement pour la première, deuxième et troisième articulation.

Les réponses en effort obtenues (figure 14) montrent que le mouvement du robot dans l'espace libre annule tout effort de contact (figure 14.a1). En effet, Dans l'intervalle de temps $[0 \text{ s}, 1.75 \text{ s}]$, les forces d'interactions sont nulles. A partir de l'instant « $t = 1.75$ s » le bras manipulateur rentre en contact avec l'environnement. Dans le cas de notre présent travail les paramètres des régulateurs sont choisis de sorte à minimiser ce pic. Ceci dans le but de minimiser au maximum les phénomènes d'instabilité engendrés par le contact. Une fois le contact établi, le robot maintient un effort désiré de 50N, normal à la surface de l'environnement.



- a1 -

- a2 -

Figure 14. La réponse en effort.
a1 et a2 Effort respectivement selon l'axe X et l'axe Y

Ainsi la courbe des efforts suivant l'axe « X » est initialement nulle puis, dès la mise en contact, elle tend vers l'effort désiré (50 N) qu'elle l'atteint au bout de 0.75s après le contact. Les efforts selon l'axe Y (figure 14.a2) sont nuls compte tenu de la tâche considérée. En effet, dans cette première tâche le déplacement de l'effecteur s'effectue suivant cette direction.

Dans le but de mieux évaluer les performances de notre structure de commande, nous proposons de considérer une tâche de suivi d'un mur de forme curviligne. En effet, ce type de trajectoire est complexe et son suivi excite toute la dynamique du robot. De plus, lors du mouvement contraint les efforts de contact peuvent apparaître selon les deux directions X et Y.

b- Suivi d'un contour curviligne en maintenant un effort constant.

Cette deuxième partie consiste en l'étude du mouvement d'un bras manipulateur redondant suivant une trajectoire curviligne. La tâche de suivi de trajectoire consiste en un déplacement de l'effecteur dans un espace libre puis il rentre en contact avec un mur curviligne, de raideur K_e ($K_e = 500 \text{ N/m}$), en maintenant un effort sur la surface (figure 15).

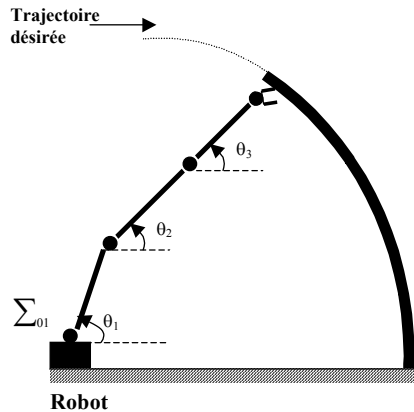
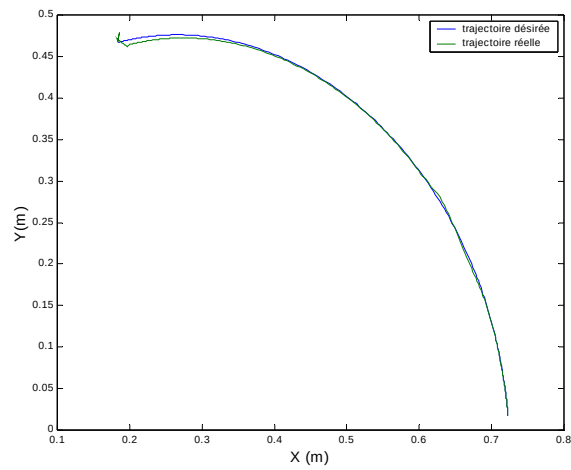


Figure 15. Suivi d'un contour curviligne en maintenant un effort.

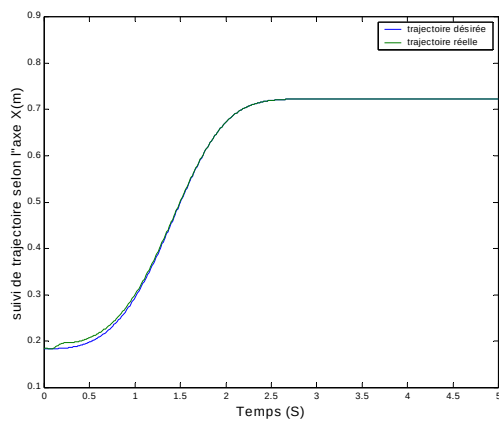
A l'instant « $t = 1.75s$ » on simule la présence d'un mur curviligne situé à la position (0.6m, 0.3m), avec lequel il rentre en contact en maintenant un effort constant sur sa surface. Les gains des PID associées à chaque articulation assurant un bon suivi sont :

$$P = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 4 \end{bmatrix} N/m.s, \quad D = \begin{bmatrix} 360 & 0 & 0 \\ 0 & 290 & 0 \\ 0 & 0 & 90 \end{bmatrix} m.s/N$$

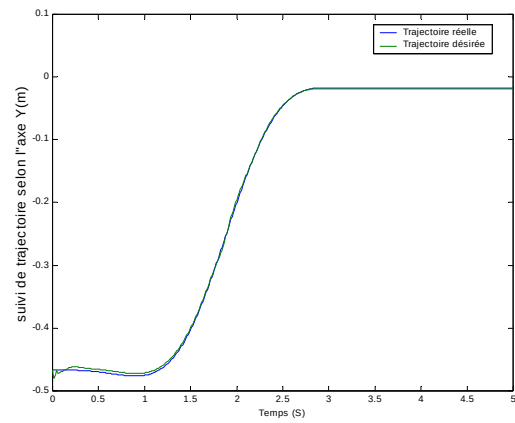
Les résultats obtenus sont consignés sur la figure 16.



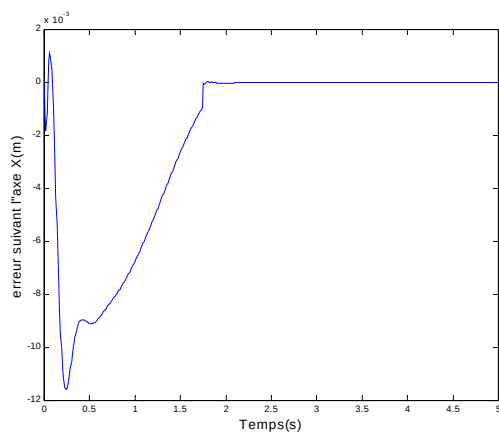
- a-



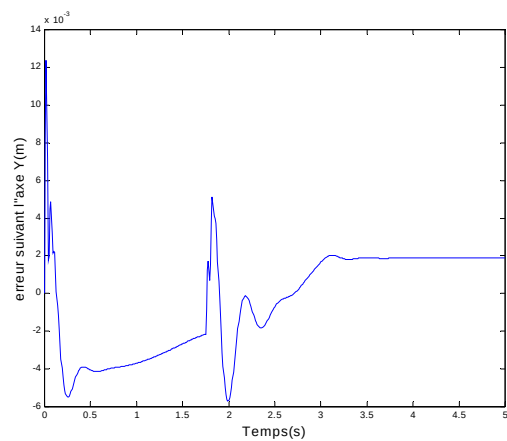
- a1-



- a2-



- b1-



- b2-

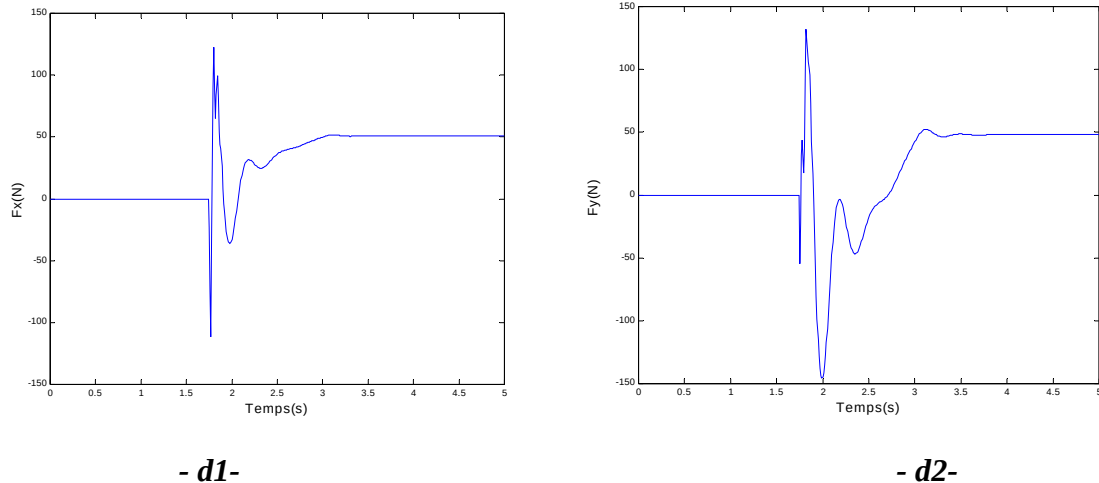


Figure 16. Suivi d'un contour curviligne en maintenant un effort.

- a. Trajectoire de l'organe terminal dans le plan (X, Y)
- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- d1, d2 Effort selon l'axe X et l'axe Y

Dans l'intervalle du temps $[0s, 1.75s]$, le robot manipulateur se déplace dans un espace libre. Dès que l'effecteur entre en contact avec le mur curviligne à « $t=1.75s$ », il suit sa surface en se déplaçant suivant les deux axes X et Y (figure 16.a). Cette première courbe montre qu'il y'a une bonne poursuite de trajectoire avant l'impact à $t = 1.75s$. L'occurrence du contact correspond à la petite déformation de la trajectoire réelle qui est visible sur la figure V.16.a.

Les figures 16.a1 et a2 illustrent les trajectoires réelles et désirées respectivement suivant l'axe X et suivant l'axe Y. La poursuite suivant l'axe X, est satisfaisante. En effet, la figure 16.b1 montre initialement que l'erreur dans l'espace cartésien est $0.4 \cdot 10^{-3} \text{ m}$ selon l'axe X. Cette erreur est due à la position initiale de l'effecteur. Une fois que l'effecteur est en contact avec l'environnement ($t=1.75s$), cette erreur décroît jusqu'à atteindre $-1.8 \cdot 10^{-6} \text{ m}$. Ceci s'explique par le fait que pour assurer un bon suivi de consignes, l'effecteur prend une configuration lui permettant de maintenir un effort désiré normal à la tangente de la surface de l'obstacle, tout en garantissant un déplacement suivant les directions non contraintes.

La figure 16.b2 montre que le suivi suivant l'axe Y est de bonne qualité. Toute fois, un pic d'amplitude $7 \cdot 10^{-3} \text{ m}$ à $t=1.75s$ apparaît puis l'erreur se stabilise à la valeur $8 \cdot 10^{-4} \text{ m}$ à partir de l'instant $t = 3s$. Ceci peut s'expliquer par la mise en contact à $t=1.75s$ du bras manipulateur avec l'environnement.

La figure 16.d1 et d2 montrent les réponses en effort obtenues durant le suivi de trajectoire. Avant la collision, le robot se déplace dans l'espace libre, aucun effort n'est mesuré par le capteur de force. En effet, dans l'intervalle de temps $[0 \text{ s}, 1.75 \text{ s}]$, les forces d'interactions suivant l'axe X et l'axe Y sont nulles. A partir de l'instant « $t = 1.75s$ » le bras manipulateur rentre en contact avec l'environnement. En effet, les courbes d'effort suivant les deux axes présentent des dépassements par rapport à la consigne désirée à cet instant suivis de quelques oscillations dues à la modélisation robot-environnement par un ressort. Ainsi, il

apparaît le phénomène d'élasticité, ces oscillations se stabilisent à la valeur 50N à l'instant $t = 3s$ suivant chaque direction. Ceci s'explique par le fait que le contact n'a pas eu lieu suivant la perpendiculaire à la surface de l'environnement. En effet, l'effort d'interaction appartient à une direction oblique à cette surface curviligne. Elle aura donc deux composantes, chacune suivant un axe.

Afin que notre étude soit la plus complète possible, nous nous intéressons dans la prochaine partie à une tâche de saisie et transport d'un objet d'un point « A » à un point « B ».

c- Saisie et transport d'un objet d'un point « A » à un point « B »

L'objectif de la présente étude, est d'évaluer le comportement de l'approche de commande proposée vis à vis d'une tâche courante des bras manipulateurs. Cette tâche consiste à la prise et au déplacement d'un objet ($K_e = 160 \text{ N/m}$).

Ainsi, le robot manipulateur redondant considéré devra :

- dans une première phase suivre une trajectoire désirée (défini par un planificateur de trajectoire),
- dans une seconde phase, prendre un objet en maintenant un effort de serrage de 20N
- et enfin soulever l'objet jusqu'à une position « B »

Dans notre étude, l'objet à manipuler est de forme carrée (côté de 0.2 m) et de masse 1.5Kg. Il se situe sur une table à la position (0.6m, 0.36m) dans le repère tâche (figure 17).

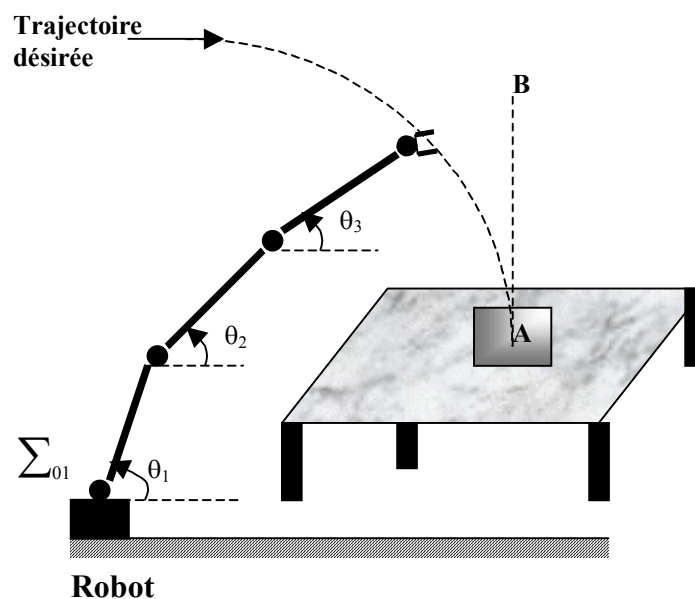


Figure 17. Tâche de prise et transport d'un objet.

Nous considérons que le robot manipulateur redondant est instrumenté par des capteurs d'effort au niveau de son effecteur, modélisés par un ressort linéaire de raideur équivalente « K_e » et de longueur au repos « L_r » (figure 18).

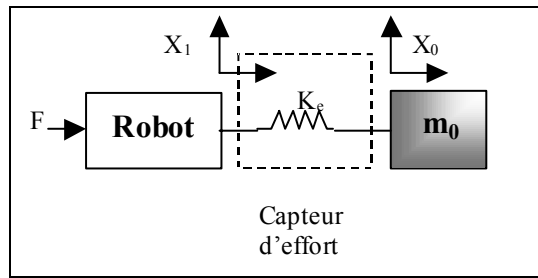


Figure 18. Modélisation de la structure mécanique des capteurs.

Avec: « m_0 » la masse de l'objet, « F » le vecteur des efforts de contact du robot avec l'objet.

$$F = K_e (X_1 + L_r - X_0) \dots \dots \dots (V.10).$$

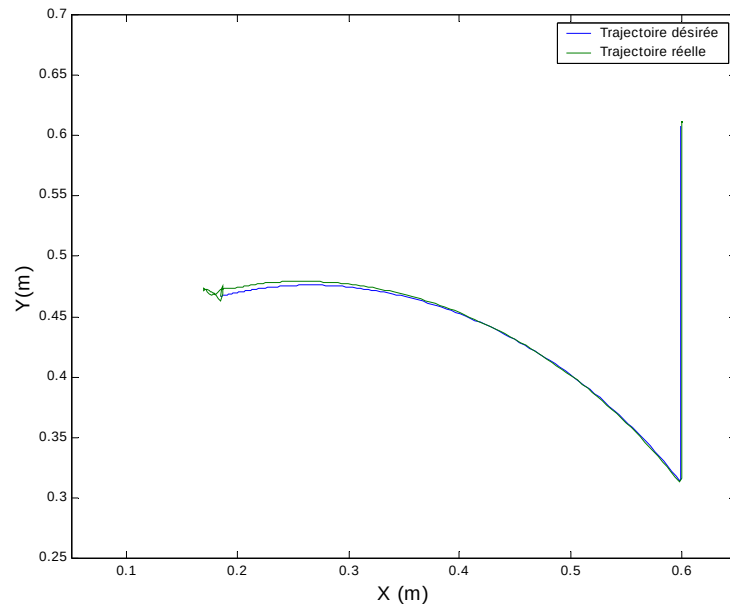
Où « X_1 » et « X_0 » sont respectivement la position du point terminal du bras et de l'extrémité de l'objet.

L'effort appliqué sur l'objet est orthogonal à sa surface. Le bon déroulement de la tâche considérée nécessite un asservissement simultané de l'effort et de la position de l'effecteur. Ainsi nous utiliserons notre approche de commande basée sur la structure de commande en effort externe.

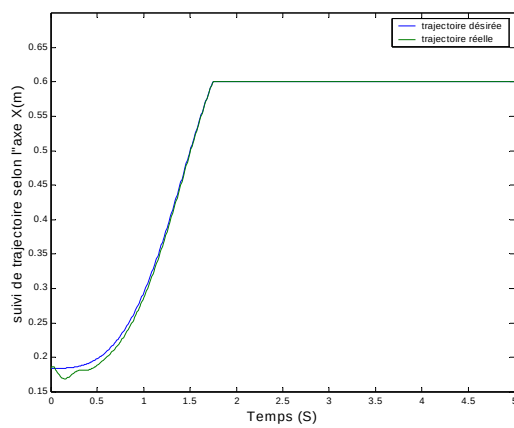
Un réglage expérimental des paramètres des PID associés aux régulateurs donnant de bons résultats de poursuite a été effectué. Les gains utilisés sont :

$$P = \begin{bmatrix} 7 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1.7 \end{bmatrix} N/m, \quad I = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 4 \end{bmatrix} N/m.s \quad D = \begin{bmatrix} 390 & 0 & 0 \\ 0 & 380 & 0 \\ 0 & 0 & 80 \end{bmatrix} m.s/N$$

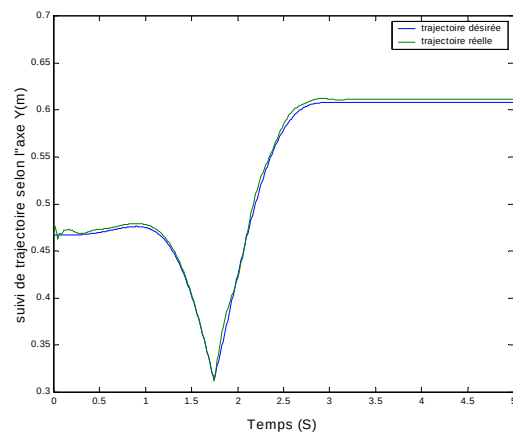
Les résultats obtenus sont consignés sur la figure 19. La figure (19.a) présente le mouvement réel et désiré du robot manipulateur redondant dans le plan (X, Y). Cette courbe montre les deux phases du mouvement, durant la première phase ($t \in [0, 1.74s]$) l'effecteur se déplace sans contrainte, à l'instant $t = 1.75s$ il rentre en contact avec l'objet. Il le saisit en exerçant un effort désiré de 20N et le déplaçant verticalement du point « A » vers le point « B » la position finale spécifiée. Cette première courbe montre un bon suivi de trajectoire.



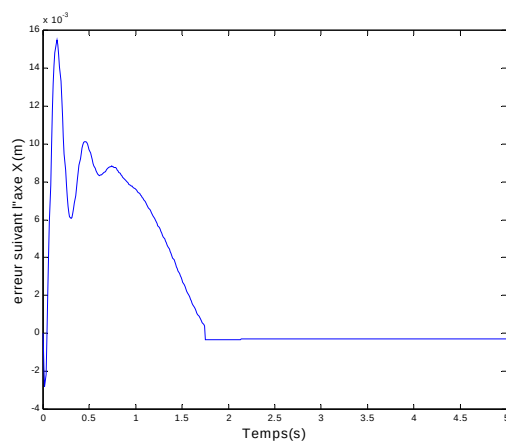
- a -



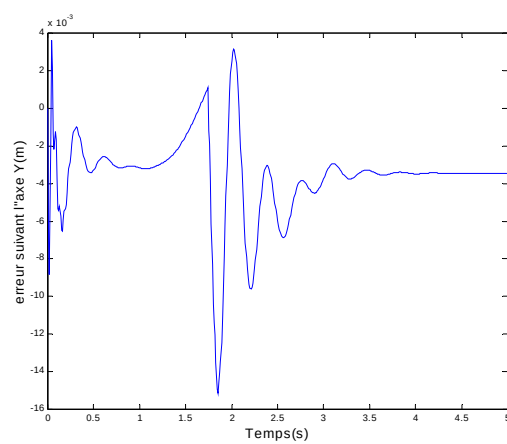
- b1 -



- b2 -



- c1 -



- c2 -

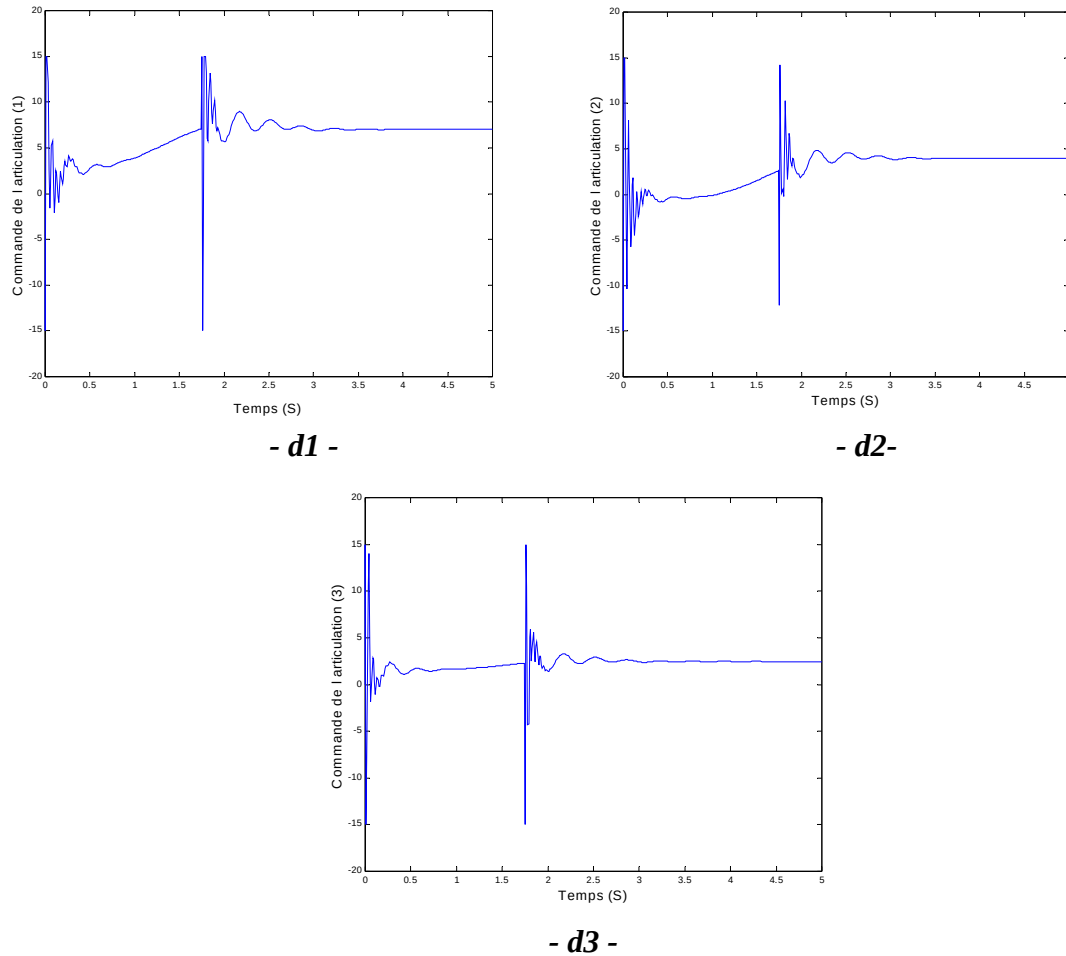


Figure 19. Transport d'un objet par un bras manipulateur redondant.

- a. Trajectoire de l'organe terminal dans le plan (X, Y)
- b1 et b2 Trajectoires suivant l'axe X et l'axe Y
- c1 et c2 Erreur de suivi selon les axes X et Y
- d1, d2 et d3 Couple appliqué respectivement aux articulations 1, 2 et 3

En effet, les courbes des erreurs suivant l'axe X et l'axe Y (figure 19.b1 et b2) montrent que les erreurs initiales sont $-2.95 \cdot 10^{-3}$ m suivant l'axe X et $-7.4 \cdot 10^{-3}$ m suivant l'axe Y. ces erreurs sont dûes essentiellement à l'inertie des bras et au fait qu'initialement la position de l'effecteur ne coïncide pas avec la position initiale désirée. A l'instant $t = 1.75$ s, les courbes des erreurs suivant les deux axes font apparaître des oscillations, les erreurs cartésiennes sont maximales à cet instant soit $-1.26 \cdot 10^{-4}$ m suivant l'axe X et $-1.2 \cdot 10^{-3}$ m suivant l'axe Y. Ceci s'explique par la mise en contact du bras manipulateur avec l'objet. Ces erreurs se stabilisent en régime permanent à $-2.4 \cdot 10^{-7}$ m suivant l'axe X et $-3.5 \cdot 10^{-3}$ m suivant l'axe Y.

Les figures 19.d1, d2 et d3 donnent les couples appliqués aux différentes articulations du bras manipulateurs redondant considéré. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit : 15 N.m pour la première articulation, 14.5 N.m pour la deuxième et de 14.5 N.m pour la troisième. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. Lors de la saisie de l'objet quelques oscillations apparaissent au niveau des couples de commande.

Les valeurs maximales sont 15 N.m, 14.7 N.m et 15 N.m respectivement pour la première, la deuxième et la troisième articulation.

La figure 20 donne les courbes des efforts obtenus.

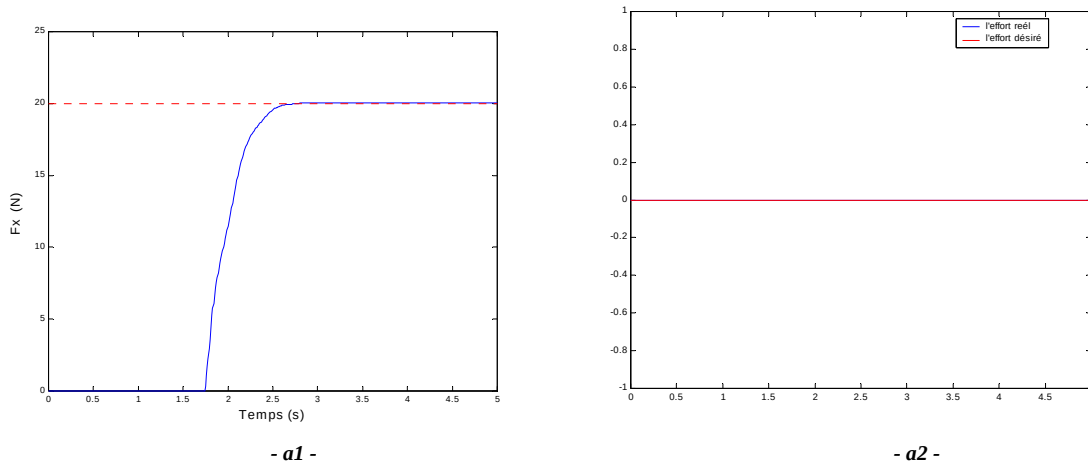


Figure 20. la réponse en effort.
a1 et a2 Effort respectivement selon l'axe X et l'axe Y.

Dans l'intervalle de temps [0 s, 1.75 s], les forces d'interactions sont nulles suivant les deux axes. Ceci est dû au mouvement non contraint d'où l'absence de tout contact. Une fois le contact établi avec l'objet, le robot maintient un effort désiré normal à la surface de l'objet de 20 N.

Les efforts selon l'axe Y sont nuls compte tenu que le déplacement s'effectue suivant cette direction. Tandis que les efforts suivant l'axe « X » qui étaient nuls en espace libre, tendent vers la valeur d'effort désirée qui est de 20 N dès la mise en contact, elle est atteinte au bout de 0.75s. Ceci s'explique par le fait que la saisie de l'objet a lieu selon cette direction.

Conclusion 2

Afin de tester la robustesse de la commande proposée dans le repère tâche nous avons étudié le mouvement d'un bras manipulateur dans un espace libre puis contraint. A cet effet, trois cas sont considérés: le suivi d'un mur vertical, le suivi d'un mur curviligne et la saisie et le transporter d'un objet. Les résultats obtenus montrent un bon suivi de trajectoire en espace libre avec des erreurs initiales un peu grandes qui sont essentiellement dues à l'erreur initiale de positionnement de l'effecteur. Nous pouvons remarquer aussi, que dès que le contact est établi avec l'obstacle la loi de commande tend à compenser les efforts mesurés, ce qui a permis de corriger le suivi de trajectoires. Ainsi, les résultats obtenus, permettent de valider l'approche de commande proposée.

L'implémentation de la commande en simulation sur notre système nous a permis d'apprécier sa simplicité de conception. Toutefois, la détermination des gains de commande nécessite la connaissance du modèle.

V.I.4 Conclusion

Au terme de cette première partie, nous avons mis en évidence les performances de la commande par découplage non linéaire à base d'un régulateur classique de type PID, en utilisant la structure de commande en effort externe pour assurer l'asservissement simultané de la position et de la force d'un bras manipulateur redondant à trois degrés de liberté.

Pour évaluer les performances de la commande proposée, plusieurs tests sont effectués. Nous avons étudié les performances de cette commande dans un espace libre et dans un espace libre puis contraint. Lorsque le manipulateur évolue en espace libre, seule la boucle d'asservissement de position contribue à la génération des consignes de positions fournies au système. Les suivis de trajectoire sont satisfaisants et pour tester la robustesse de cette commande en espace articulaire libre, nous avons simulé le mouvement du bras manipulateur redondant avec une charge de 1,5 kg. Les résultats de simulation obtenus sont satisfaisants. En effet, nous obtenons un bon suivi de trajectoire. Les erreurs sont faibles en régime permanent pour les cas de mouvements sans ou avec charge. Nous avons aussi simulé une défaillance sur l'un des régulateurs. Dans ce cas les résultats obtenus sont de bonne qualité malgré la divergence de la trajectoire de troisième articulation. Ceci illustre l'avantage de la commande décentralisée adoptée par rapport la commande centralisée.

Dans le cas des autres tâches considérées, lorsque le robot entre en interaction avec un environnement de raideur « K_e », la transition mouvement libre - mouvement contraint se traduit par un choc. Dans cette phase de mise en contact, la structure de commande en effort externe présente de meilleures performances grâce à sa boucle d'asservissement de position qui est interne à la boucle d'effort (le système reste toujours asservi en position).

L'analyse globale des résultats obtenus montre que la commande proposée permet d'asservir correctement notre système pour toutes les tâches proposées. Cependant, la difficulté de l'analyse mathématique de l'ensemble robot- environnement constitue un inconvénient majeur à l'utilisation des approches de commande conventionnelle. De plus les réglages des gains des PID sont réalisés expérimentalement.

Pour remédier à ce point, on propose de s'orienter vers les méthodes d'intelligence artificielles (IA) qui fera l'objet des deux prochains chapitres. Ainsi, dans la partie suivante (partie II), nous présentons les principes de base d'un raisonnement par la logique floue, la synthèse d'un contrôleur flou. En fin, nous appliquerons cette approche à une structure de commande en effort externe.

Deuxième partie

Commande hybride Force - Position utilisant les concepts de la logique floue

La synthèse d'une loi de commande performante et la difficulté de l'analyse mathématique de la dynamique du robot et celle de son interaction avec l'environnement, rendent dans certains cas l'application de l'automatique classique difficile, ce qui nous amène à opter pour des concepts Flous, cette approche est particulièrement intéressante car elle permet d'affecter au système de commande une adaptabilité à la variation des paramètres de la tâche.

L'objectif de cette deuxième partie est donc la synthèse de la loi de commande à base de la logique floue, le système de commande résultant est dit intelligent. Toutefois, la structure de commande reste la même que celle utilisée dans la commande conventionnelle. Ceci dit, il est nécessaire de remplacer le régulateur classique (PID) par un régulateur flou assurant simultanément l'asservissement de position et de force (figure 1).

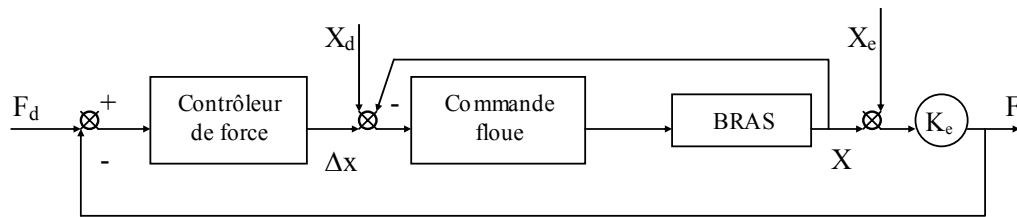


Figure 1. Structure de commande en effort externe basée sur la logique floue.

V.II.5 Commande par Logique Floue

1.1. Structure d'un régulateur logique flou

L'objectif de la commande par logique floue consiste à piloter un processus en utilisant les connaissances et l'expérience d'un opérateur humain qualifié « EXPERT ». Les connaissances de l'expert sont schématisées à l'aide d'un ensemble de règles de conduite basées sur un vocabulaire symbolique. La figure 2 représente la structure d'un réglage par logique floue. Le régulateur par logique floue (noté RLF) fournit un signal de commande « U ». Il reçoit à son entrée la grandeur de consigne « X » et une ou plusieurs grandeurs mesurées, réunies dans le vecteur « Y_m ». Ce dernier contient en général la grandeur à régler « Y » et le cas échéant, d'autres grandeurs qui sont déterminantes pour saisir l'évolution dynamique du système à régler [Büh 94] [Kha 01].

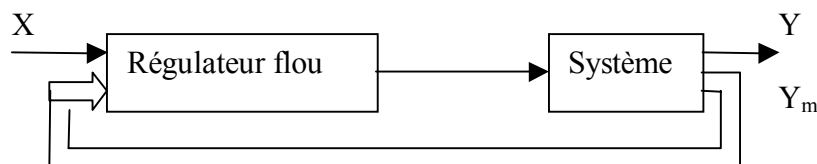


Figure 2. Structure d'un réglage par logique floue

Par opposition à un régulateur standard, le régulateur par logique floue ne traite pas une relation mathématique bien définie, mais utilise des inférences avec plusieurs règles, se basant sur des variables linguistiques [Ram 02]. Ces inférences sont alors traitées par des opérateurs de la logique floue.

1.2. Contrôleur flou

La commande par logique floue se base sur les connaissances et l'expérience d'un opérateur humain synthétisée sous forme de règles empruntées à son langage. Elle se résume en trois étapes la fuzzification, l'inférence et la défuzzification. La figure 3 montre la configuration interne d'un régulateur par logique floue [Lee 90a] [Bül 94]. On distingue trois blocs.

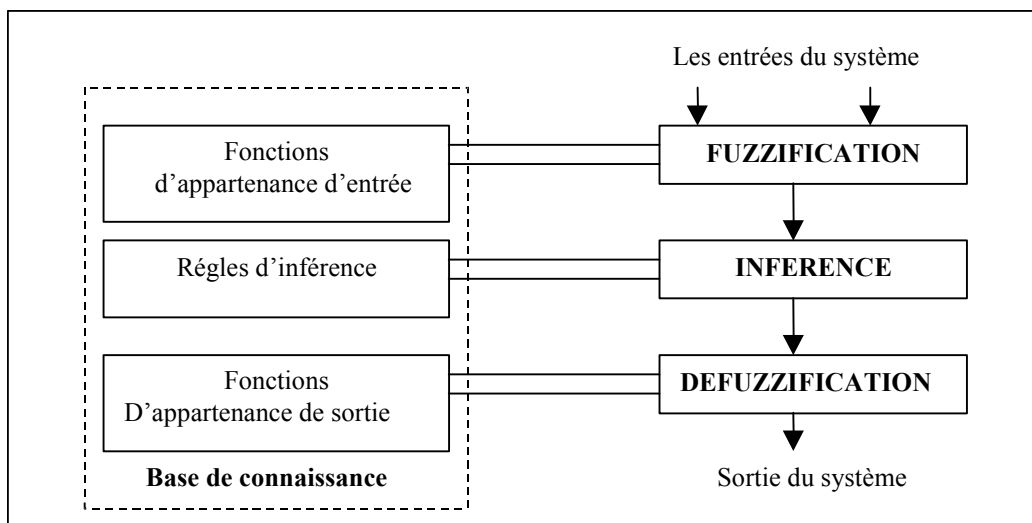


Figure 3. Configuration d'un régulateur flou .

- **La fuzzification:** La première étape du traitement d'un problème par la logique floue consiste donc à modéliser chacune des entrées du système par des courbes donnant les degrés d'appartenance à différents états identifiés pour ces entrées. La fuzzification des variables d'entrée est une phase délicate du processus mise en œuvre par la logique floue. Elle est souvent réalisée de manière itérative et requiert de l'expérience.
- **L'inférence:** Plusieurs valeurs de variables linguistiques, convenablement définies par des fonctions (fuzzification) sont liées entre elles par des règles formulées par l'opérateur afin de tirer des conclusions, on parle alors d'inférence.
- **La défuzzification:** C'est une interface « Linguistique-Numérique » qui permet de transformer les valeurs linguistiques (valeurs floues) en valeurs numériques directement exploitables par le processus. Il existe plusieurs stratégies de défuzzification on peut citer :
 - La technique du maximum.
 - La technique de la moyenne pondérée.
 - La technique du centre de gravité.

1.3. Types de régulateurs flous

Il existe plusieurs types de régulateurs, parmi eux on cite ceux, de Mamdani, de Sugeno....etc.

- a- Régulateur flou du type Mamdani: Ce type de régulateurs utilise des règles à prémisses et conclusions symboliques, l'inférence (max, min) et la défuzzification par centre de gravité.

Exemple:

Si « l'erreur est négative » et « sa variation est négative » ALORS « la commande est grande ».

- b- Régulateurs flous du type Sugeno: Ce type de régulateurs est basé sur des règles, dont les prémisses sont symboliques, tandis que les conclusions sont numériques.

Exemple:

Si « l'erreur est négative » et « sa variation est négative » ALORS « la sortie est : $u=1+3*e-2*de$ ».

Remarque: Dans notre présent travail, nous avons choisi d'utiliser un régulateur proportionnel dérivateur à logique floue de type Sugeno.

V.II.6 Stratégie de commande

L'application de la commande floue à un système complexe nécessite d'abord de choisir entre la structure de commande centralisée (figure 4) et la structure de commande décentralisée (figure 5) selon les critères suivants [Bou 02]:

- La réalisabilité de la commande.
- Le temps de calcul.

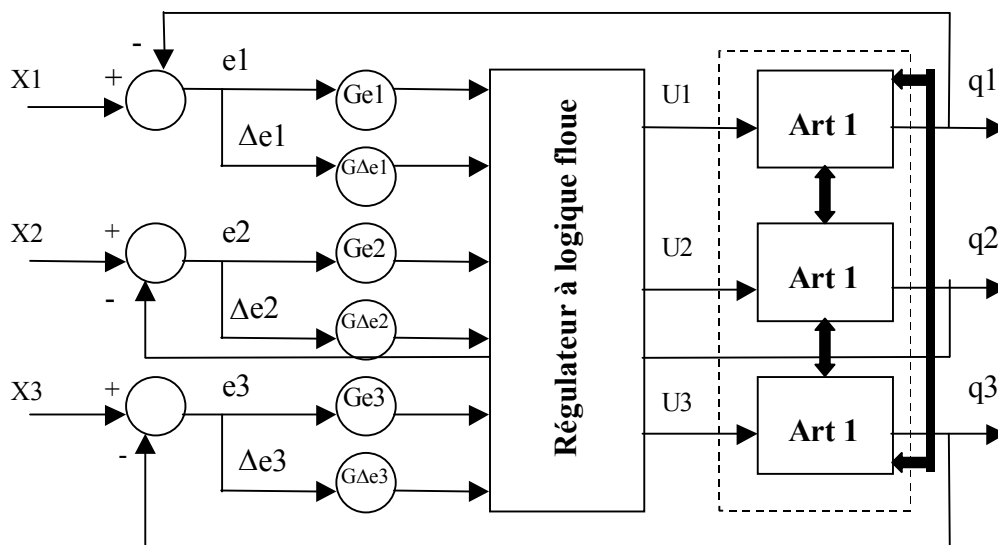


Figure 4. Structure de la commande Centralisée.

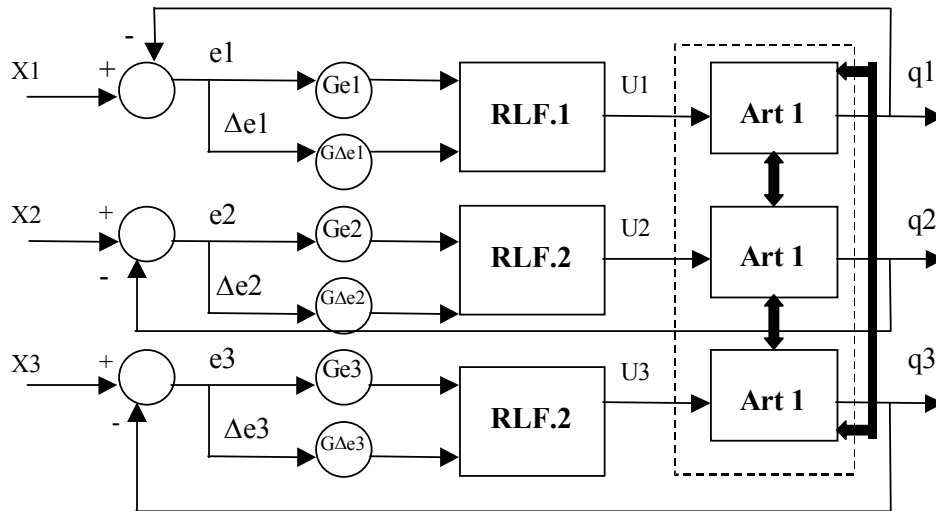


Figure 5. Structure de la commande Décentralisée.

Où RLF.1, RLF.2 et RLF.3 désignent respectivement le premier, second et troisième régulateur flou.

Afin de choisir entre ces deux structures, le tableau 1 donne l'évaluation du nombre de règles utilisées pour chaque structure en fonction de la variation du nombre de fonctions d'appartenance.

Nombre de classe d'appartenance	Nombre de règles floues	
	Cas centralisé	Cas décentralisé
3	81	9
5	625	25
7	2401	49
11	14641	121

Tableau 1. Nombre de règles, cas centralisé, cas décentralisé

L'analyse du tableau fait apparaître, dans le cas centralisé les difficultés suivantes :

- Le nombre élevé de règles à déterminer.
- La difficulté d'avoir les règles appropriées. Ce qui nécessite une très grande connaissance du système et cela n'est pas toujours évident.
- Le temps de calcul est grand.
- La grande taille du régulateur (d'où la difficulté de réalisabilité).

Le cas décentralisé présente quant à lui les facilités suivantes [Bou 02]:

- Un temps de calcul réduit.
- La commande du système est partagée entre les trois régulateurs.
- La taille des trois régulateurs est réduite.
- La possibilité de la réalisation pratique.

Ceci dit, nous adoptons dans notre présente étude la structure décentralisée. Ainsi, chaque articulation sera commandée indépendamment. Les interconnexions et l'effet de la

commande de l'autre articulation seront considérés par les régulateurs comme des perturbations externes.

V.II.7 Synthèse de la Commande par logique floue

La synthèse et la conception de la commande d'un système doit répondre à un certain nombre de critères de performances et d'objectifs à atteindre à savoir: rapidité, précision, robustesse, etc.. Ce paragraphe a pour objectif, la synthèse d'un régulateur à base de connaissances, cet objectif est atteint en suivant certaines étapes nous amenant pas à pas au régulateur désiré.

Le régulateur synthétisé doit répondre aux exigences et critères suivants :

- Taille du régulateur réduite. Elle se présente comme étant le nombre de règles, dans la base de connaissance.
- Rapidité du calcul de la commande.
- Bonne précision.
- Robustesse.

La conception d'un tel régulateur doit se faire par expérimentation en définissant [Lee 90b] [Bou 99] :

- Les entrées et les sorties du système à commander (bras manipulateur).
 - Les entrées et les sorties du régulateur flou (RLF).
 - Les fonctions d'appartenance et règles d'inférence.
 - Evaluation des règles et défuzzification.
- e- **Entrées et Sorties du système à commander:** La structure décentralisée choisie, permet de commander les actionneurs de chaque articulation du bras manipulateur séparément. En utilisant une tension « U ». La sortie du système à commander est la position articulaire.
- f- **Entrées et Sorties du régulateur par la logique floue:** Il s'agit de fixer le nombre et la nature des grandeurs d'entrée. Une de ces grandeurs d'entrée doit être l'erreur $e = X_{désiré} - X_{mesuré}$. La seconde grandeur d'entrée est la variation de l'erreur entre deux échantillons successifs (dérivée approximative de l'écart) $\Delta e = e(k) - e(k-1)$. Dans notre présent travail nous avons choisi la structure décentralisée avec des régulateurs flous de type Sugeno, leur organisation développée dans ce manuscrit correspond à celle de tout contrôleur flou (figure 6). Elle comporte ainsi les étapes de fuzzification, d'inférence et de défuzzification.

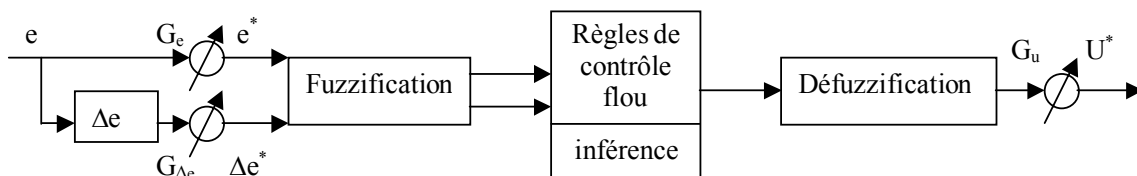


Figure 6. Structure d'un régulateur flou

- g- **Les fonctions d'appartenance:** Les variables d'entrée et de sortie étant définies, nous devons subdiviser leurs domaines de variation respectifs en sous-ensembles représentés par des fonctions d'appartenance.

- 1- **Normalisation** : pour faciliter la définition des fonctions d'appartenance, les grandeurs physiques d'entrée et de sortie sont réduites à des grandeurs normalisées variant entre $[-1,1]$. Un problème particulier est posé par le choix de ces valeurs de gains de normalisation ($G_e; G_{\Delta e}; G_u$) qui sont obtenues expérimentalement.
- 2- **Choix des fonctions et les règles d'inférence**: Les fonctions d'appartenance des variables d'entrées adoptées sont des fonctions gaussiennes définies par la figure 7. Le choix des règles est déterminé essentiellement par les inférences adoptées dont la formulation dépend du comportement statique et dynamique du système à régler, ainsi que du type d'asservissement envisagé (régulation en position dans notre cas).

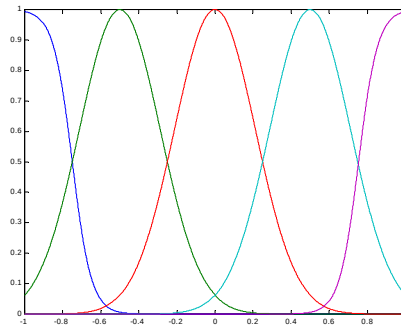


Figure 7. Sous-ensembles flous définis respectivement pour l'erreur et sa variation.

Les règles d'inférences choisies dans notre présente étude, sont représentées dans le tableau 2.

Δe e	NB	NS	ZE	PS	PB
NB	NB	NB	NB	NS	ZE
NS	NB	NB	NS	ZE	PS
ZE	NB	NS	ZE	PS	PB
PS	NS	ZE	PS	PB	PB
PB	ZE	PS	PB	PB	PB

Tableau 2. Base de règles du régulateur flou (5x5).

- h- **Défuzzification**: Elle est réalisée par la technique du centre de gravité. Dont la formule numérique est [Bab 98]:

$$\Delta U^* = \frac{\sum_{i=1}^m \mu(\Delta U_i) \cdot \Delta U_i}{\sum_{i=1}^m \mu(\Delta U_i)} \dots \dots \dots (V.11).$$

V.II.8 Simulation de la commande par la logique floue

Afin d'évaluer les performances de la commande proposée, nous l'appliquerons pour l'asservissement d'un bras manipulateur redondant à trois liaisons rotoïdes. La tâche à réaliser consiste en un suivi de trajectoire, nous avons choisi la même trajectoire que celle utilisée dans la première partie (commande dynamique), à savoir la trajectoire de LEAHVY. Dans la présente étude, nous avons considéré les mêmes configurations de mouvements que celles utilisées dans la commande par découplage non linéaire: Soit, des mouvements dans un espace articulaire sans contrainte et des mouvements dans un espace cartésien libre puit contraint. Les régulateurs utilisés lors des simulations sont de type Sugeno utilisant 5 fonctions d'appartenance pour chacune des entrées. Le réglage des gains de commande a été effectué expérimentalement suite à plusieurs tests, afin d'obtenir un bon suivi de la trajectoire désirée.

➤ Commande par la logique floue dans un espace articulaire

Le robot manipulateur redondant se déplace ici dans un espace articulaire libre. Il n'y a donc aucun contact avec l'environnement. La force de contact étant nulle. La structure de la commande en effort externe est équivalente à un schéma d'asservissement en position (figure 8).

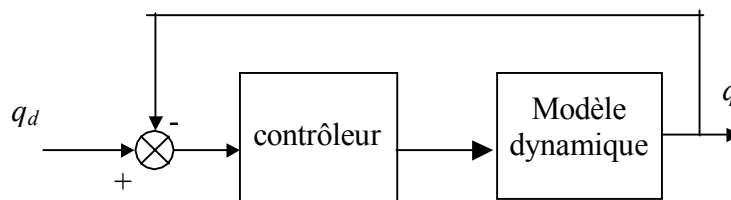


Figure 8. Asservissement de la position dans l'espace articulaire

De même que dans la commande dynamique, trois types de configurations sont considérés: le cas d'un mouvement libre sans charge, le cas d'un mouvement libre avec une charge et le cas d'une défaillance sur un des actionneurs du bras manipulateur.

• Mouvement dans un espace libre sans charge

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

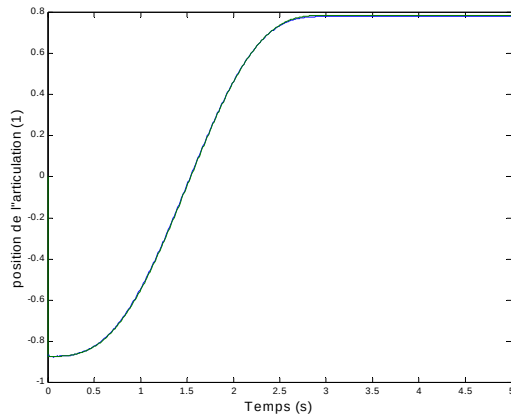
$$\begin{aligned} G_{u1} &= 490; & G_{e1} &= 2 \cdot \pi; & G_{de1} &= 4 \cdot \pi; \\ G_{u2} &= 400; & G_{e2} &= 1 \cdot \pi; & G_{de2} &= 0.7 \cdot \pi; \\ G_{u3} &= 420; & G_{e3} &= \pi; & G_{de3} &= 1 \cdot \pi; \end{aligned}$$

Les résultats obtenus sont consignés sur la figure 9, et font apparaître un bon suivi de trajectoire.

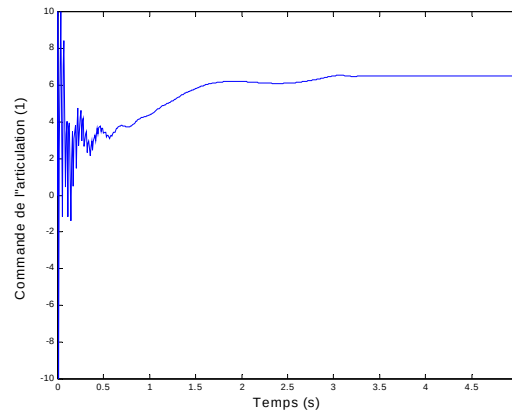
Les figures 9.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement des articulations 1, 2 et 3. Ces courbes montrent un bon suivi de trajectoires. En effet, les courbes des erreurs de suivis articulaires (figures 9.c1, c2 et c3) montrent que les erreurs initiales sont de $-8 \cdot 10^{-3}$ rad pour l'articulation 1, -0.04 rad pour la deuxième et 0.040 rad pour la troisième articulation. Ces erreurs sont dues essentiellement à l'inertie des bras et au fait qu'initialement les positions des bras ne coïncident pas avec les positions initiales désirées. En régime permanent les erreurs sont de 0.006 rad pour l'articulation 1, 0.007 rad pour la deuxième et 0.001 rad pour la troisième articulation.

Les figures 9.b1, b2 et b3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant considéré. Les courbes obtenues montrent qu'au démarrage les

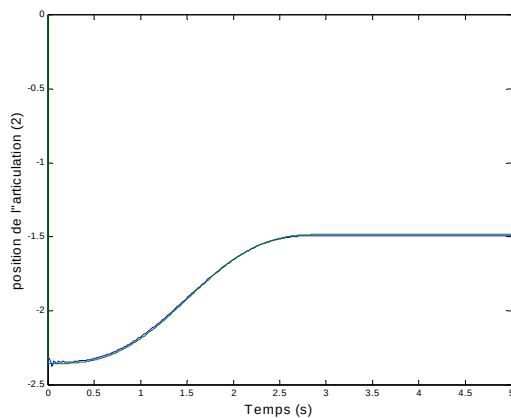
valeurs des couples sont élevées soit: 10 N.m pour la première articulation, 8,4 N.m pour la seconde et en fin de -10 N.m pour la troisième articulation. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.50 N.m pour la première articulation, 3.92 N.m pour la deuxième articulation et 2.47 N.m pour la troisième articulation. Nous pouvons aussi constater qu'il existe une relation quasi-linéaire entre les erreurs articulaires et les vecteurs de commandes.



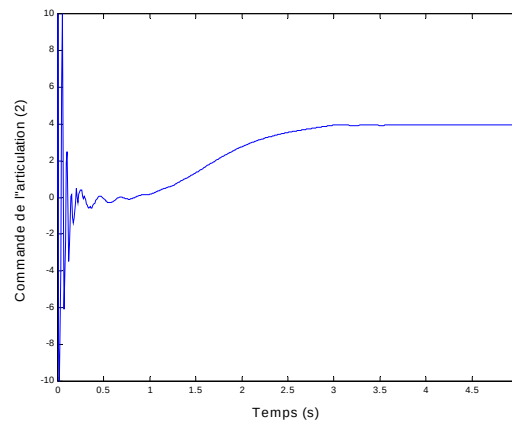
- a1 -



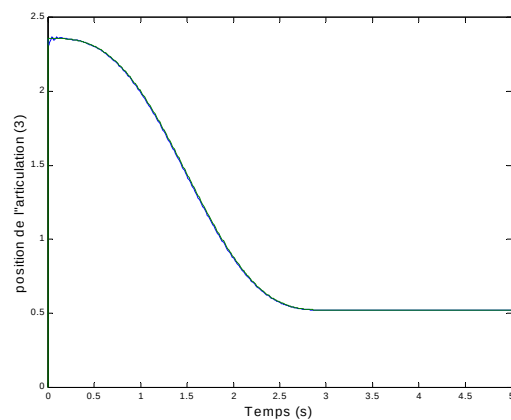
- b1 -



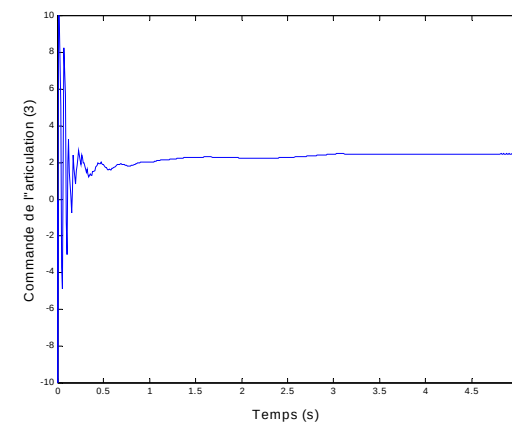
- a2 -



- b2 -



- a3 -



- b3 -

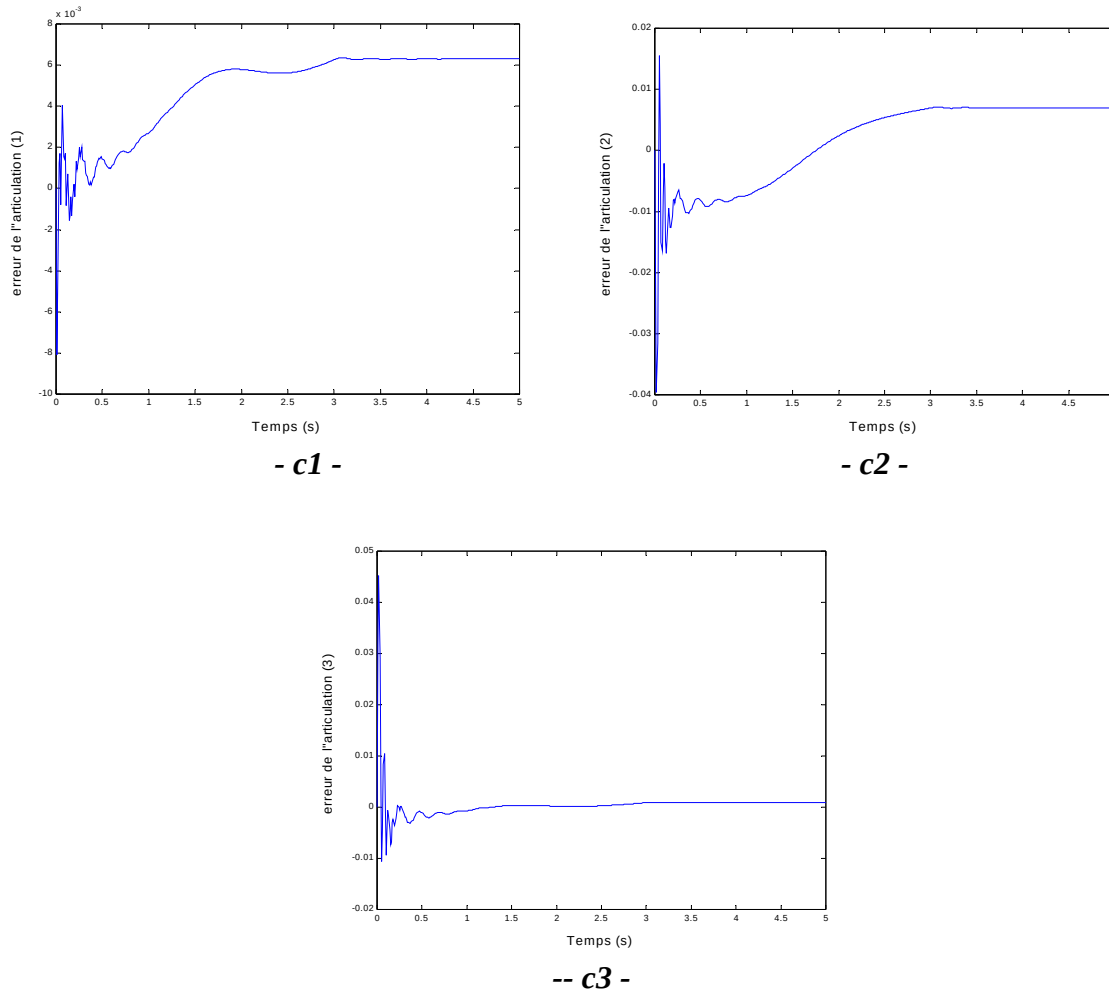


Figure 9. Mouvement dans un espace libre sans charge.

- a_1 , a_2 et a_3 Trajectoires respectivement de l'articulation 1,2 et 3.
- b_1 , b_2 et b_3 Couples appliqués respectivement à l'articulation 1,2 et 3.
- c_1 , c_2 et c_3 Erreurs de suivis respectivement de l'articulation 1,2 et 3

Ces premiers résultats nous permettent de valider l'approche de la commande proposée. Afin de tester la robustesse de la commande proposée nous avons effectué une seconde expérience qui consiste à étudier le mouvement du même bras manipulateur avec une charge.

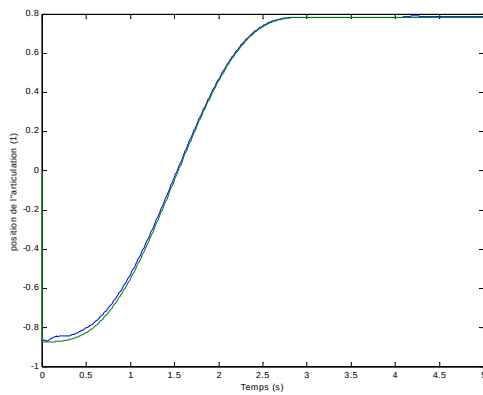
• Mouvement dans un espace libre avec charge

Nous avons utilisé la même expérience que celle appliquée pour la commande par découplage non linéaire, le robot démarre avec une charge de valeur 1.5 Kg qui sera lâchée à l'instant 2 secondes après le démarrage.

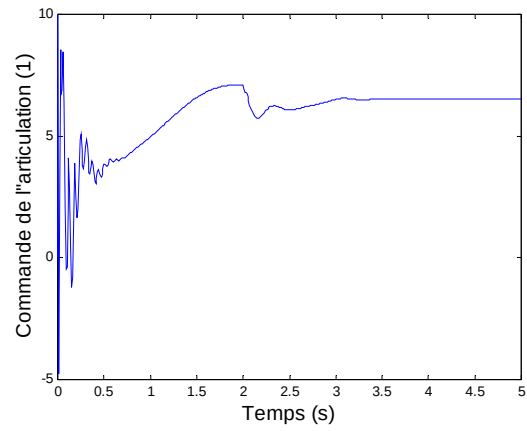
Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$$\begin{array}{lll}
 G_{u1}=300; & G_{e1}=0.8\pi/2; & G_{de1}=3\pi; \\
 G_{u2}=220; & G_{e2}=1.7\pi/2; & G_{de2}=1\pi; \\
 G_{u3}=100; & G_{e3}=0.9\pi; & G_{de3}=0.9\pi;
 \end{array}$$

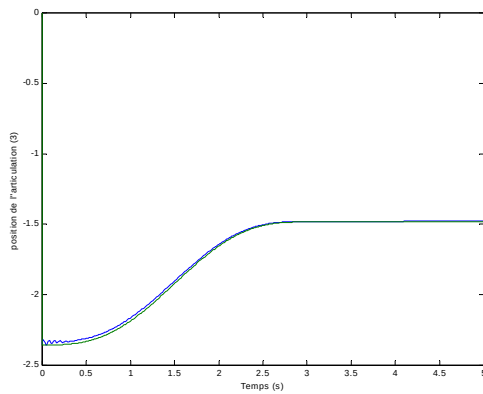
Dans ce test nous avons simulé le mouvement d'un bras manipulateur redondant portant initialement une charge de 1.5 Kg qu'il lâchera au cours du mouvement (à l'instant $t = 2s$) le bras manipulateur lâche cette charge, et poursuit son mouvement selon une trajectoire désirée.



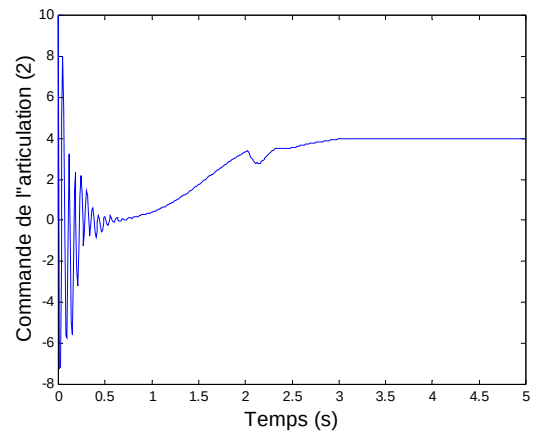
- a1 -



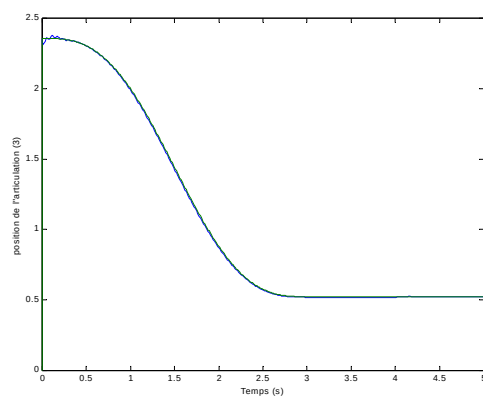
- b1 -



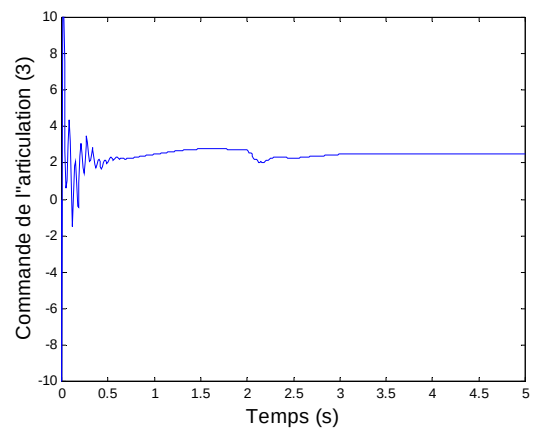
- a2 -



- b2 -



- a3 -



- b3 -

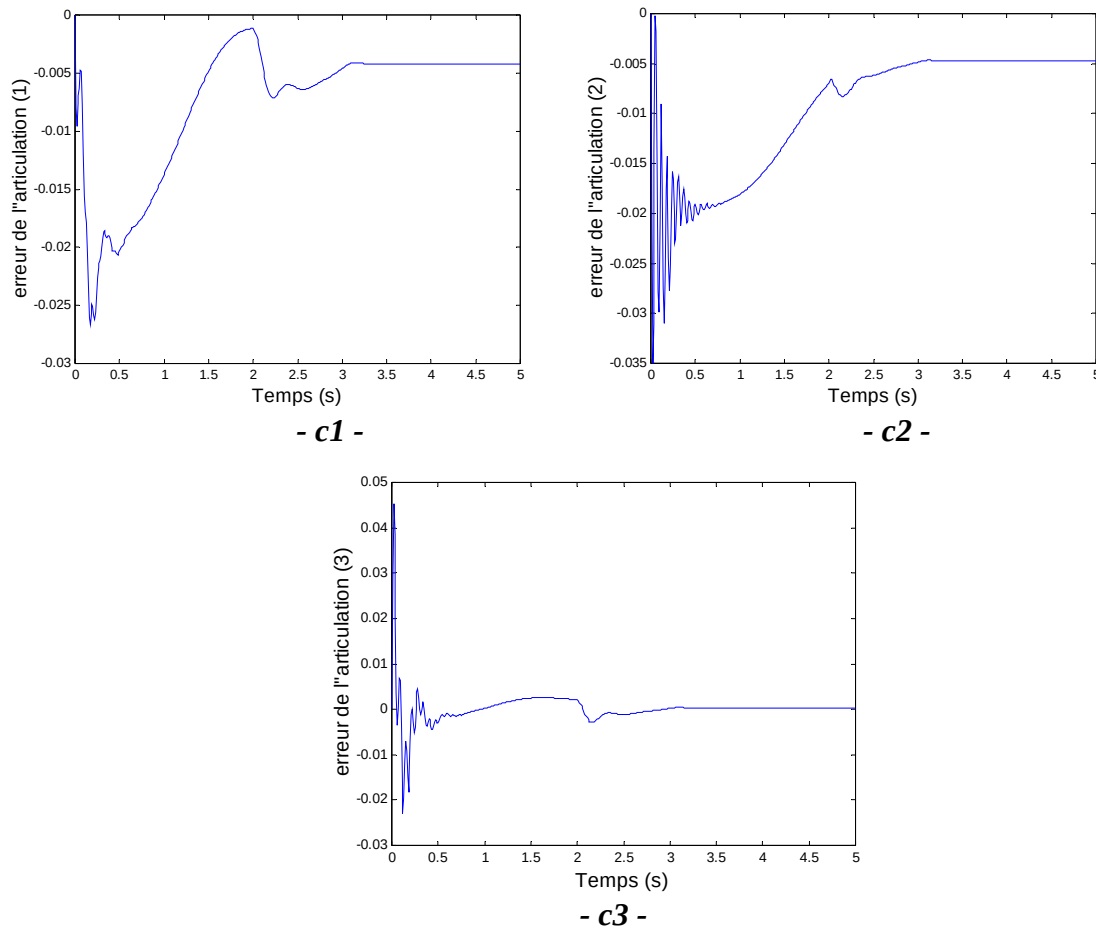


Figure 10. Mouvement dans l'espace libre du bras manipulateur redondant soumis à une charge initiale de 1.5 Kg variant instantanément ($t = 3s$) à 0 Kg

- a1, a2 et a3 Trajectoires respectivement de l'articulation 1, 2 et 3.
- b1, b2 et b3 Couples appliqués respectivement à l'articulation 1, 2 et 3.
- c1, c2 et c3 Erreurs de suivis respectivement de l'articulation 1, 2 et 3

Les figures 10.a1, a2 et a3 donnent les trajectoires articulaires réelles et désirées respectivement des trois articulations. Ces courbes montrent un suivi de trajectoires satisfaisant. En effet, Les courbes des erreurs de suivi articulaires (figures 10.c1, c2 et c3) montrent que les erreurs initiales sont 0.03 rad pour la première articulation, 0.025 rad pour la deuxième et -0.06 rad pour la troisième articulation. Ceci s'explique par le fait qu'initialement les positions des bras ne coïncident pas avec les positions initiales désirées ainsi l'effet de la charge supplémentaire. A l'instant « $t = 2s$ », les courbes des erreurs articulaires font apparaître de faibles oscillations. Les erreurs articulaires sont maximales à cet instant soit : $-2.42 \cdot 10^{-3}$ rad pour la première articulation, $-7 \cdot 10^{-3}$ rad pour la deuxième et $1.2 \cdot 10^{-3}$ rad pour la troisième. Ceci s'explique par la perte de la charge qui agit sur le système comme une perturbation. Ces erreurs articulaires se sont stabilisées à l'instant « $t = 3s$ » à des valeurs respectivement: $-4.2 \cdot 10^{-3}$ rad pour la première articulation, $-4.7 \cdot 10^{-3}$ rad pour la deuxième et $3 \cdot 10^{-4}$ rad pour la troisième articulation.

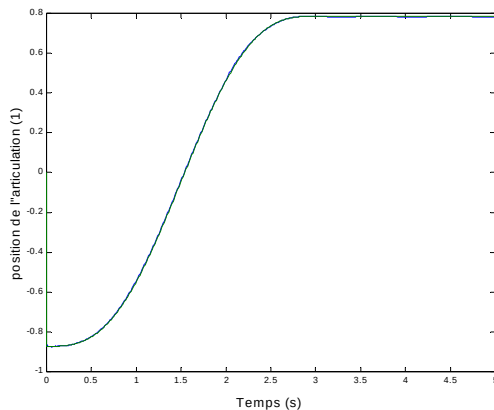
Les figures 10.b1, b2 et b3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant utilisé. Au démarrage les valeurs des couples sont élevées soit: 28.6 N.m pour la première articulation, 26.3 N.m pour la deuxième et de 27 N.m pour la troisième. Ces couples sont sensiblement plus élevés que ceux obtenus lors d'un mouvement libre sans charge. En effet, le bras manipulateur étant initialement au repos il est nécessaire

d'appliquer des couples élevés afin de vaincre l'inertie des bras et la charge supplémentaire (initialement la charge portée par l'effecteur est de 1.5 Kg). En régime permanent, à l'instant « $t = 2 \text{ s}$ », les courbes représentant les couples font apparaître de faibles oscillations. Les valeurs maximales des couples à cet instant sont respectivement: 7.5 N.m pour la première articulation, 4.6 N.m pour la deuxième et 2.95 N.m pour la troisième. Ceci s'explique par la perte de la charge qui agit sur le système comme une perturbation. Les vecteurs commandes finissent par se stabiliser à l'instant « $t = 3 \text{ s}$ » aux valeurs: 6.50 N.m pour la première articulation, 3.94 N.m pour la deuxième articulation et 2.48 N.m pour la troisième.

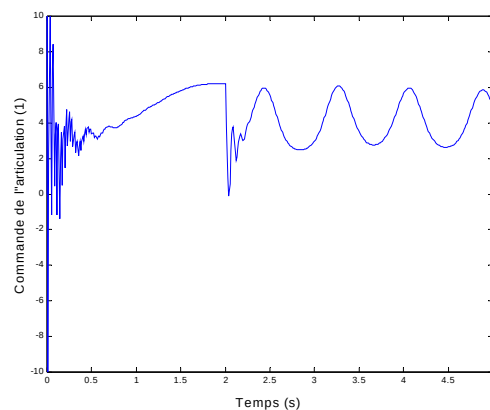
Les résultats obtenus montrent la robustesse de la commande vis à vis de la variation de la charge. En effet, la commande proposée permet de compenser les perturbations engendrées par la chute de la charge de 1.5 Kg pendant le mouvement.

- **Cas d'une défaillance de l'un des régulateurs (ou actionneur) des bras du robot manipulateur redondant**

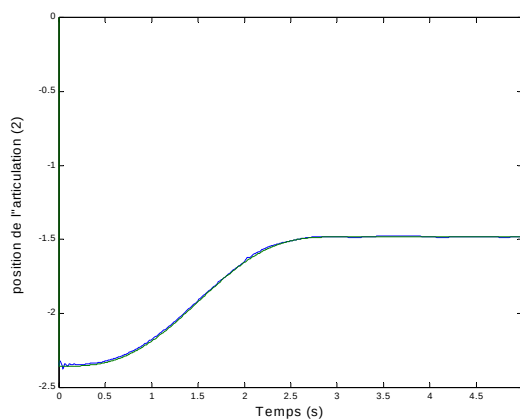
Dans cette partie, nous simulons à l'instant $t = 2 \text{ s}$ une défaillance sur le régulateur de la troisième articulation.



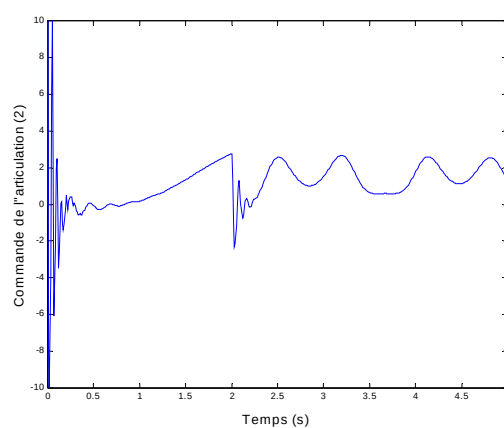
- a1 -



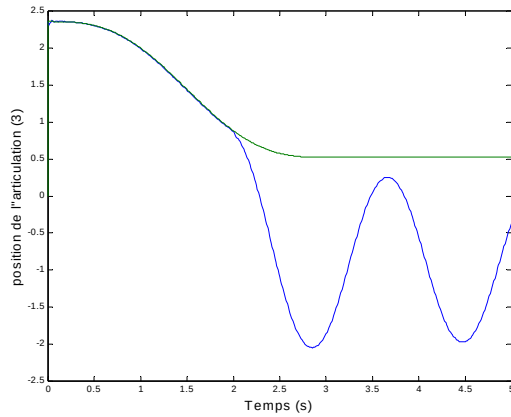
- b1 -



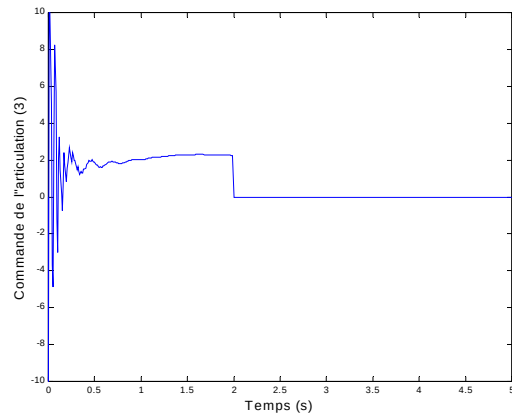
- a2 -



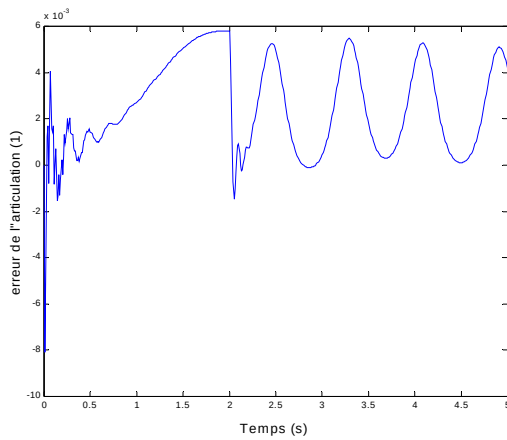
- b2 -



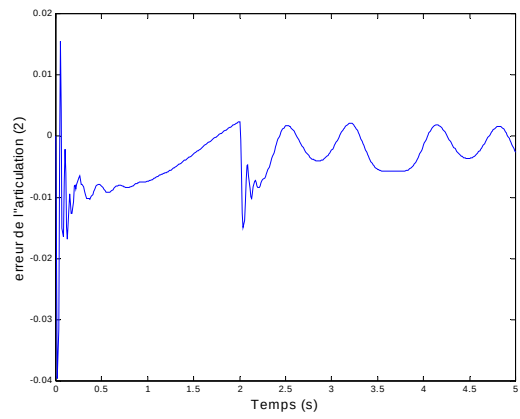
- a3 -



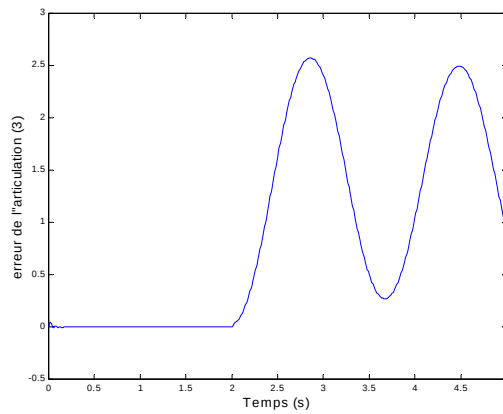
- b3 -



- c1 -



- c2 -



- c3 -

Figure 11. Mouvement dans l'espace libre du bras manipulateur redondant sans charge. Cas d'une rupture de la commande sur la troisième articulation

- a1, a2 et a3 Trajectoires respectivement de l'articulation 1,2 et 3.
- b1, b2 et b3 Couples appliqués respectivement à l'articulation 1,2 et 3.
- c1, c2 et c3 Erreurs de suivis respectivement de l'articulation 1,2 et 3.

Les figures 11.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement de la première, deuxième et troisième articulation. Ces courbes montrent un suivi de trajectoires satisfaisant pour la première et la deuxième articulation. Dans le cas de la troisième articulation

une divergence apparaît à partir de l'instant « $t = 2s$ ». En effet, à cet instant une défaillance de ce régulateur est introduite.

Les courbes des erreurs de suivis articulaires (figures 11.c1, c2 et c3) montrent que les erreurs initiales sont: $-8 \cdot 10^{-3}$ rad pour la première articulation, -0.04 rad pour la deuxième et 0.04 rad pour la troisième articulation. Ces résultats sont identiques à ceux obtenus lors du mouvement du bras manipulateur sans charge. Ces erreurs articulaires sont dues essentiellement à l'inertie du bras et aux erreurs initiales de positions des bras du manipulateur. En régime permanent, à l'instant « $t = 2s$ », les courbes des erreurs articulaires font apparaître un pic d'amplitude -0.0018 rad pour la première articulation, -0.015 rad pour la deuxième suivi de faibles oscillations. Ces dernières se stabilisent autour de $-2 \cdot 10^{-3}$ rad pour la première articulation et $-1 \cdot 10^{-3}$ rad pour la deuxième. Dans le cas de la troisième articulation nous observons des oscillations croissantes d'où la divergence du système. Ces résultats s'expliquent par l'occurrence d'une défaillance sur le régulateur de la troisième articulation. En effet, une défaillance sur une articulation entraîne des contraintes plus élevées sur les autres articulations en agissant comme une perturbation. De ce fait, les deux régulateurs seront plus sollicités à cet instant.

Les figures 11.b1, b2 et b3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant. Les courbes obtenues montrent qu'au démarrage les valeurs des couples sont élevées soit: 10 N.m pour la première articulation, 8.4 N.m pour la deuxième et -10 N.m pour la troisième. Ces couples sont identiques à ceux obtenus lors d'un mouvement libre sans charge. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des axes du robot manipulateur. En régime permanent, à l'instant « $t = 2s$ », les courbes obtenues font apparaître un pic d'amplitude -0.1 N.m et -2.5 N.m suivi d'oscillations respectivement sur la première et la deuxième articulation. Dans le cas de la troisième articulation le vecteur commande s'annule. Ceci est dû à l'occurrence d'une défaillance sur le régulateur (ou actionneur) lié à cette troisième articulation. Les vecteurs de commandes appliqués à la première et deuxième articulation finissent par se stabiliser autour des valeurs: 4 N.m pour la première articulation et 2 N.m pour la deuxième.

Les résultats de simulation obtenus montrent que malgré la divergence de la troisième articulation causée par la défaillance introduite sur le troisième régulateur, les autres articulations suivent leurs trajectoires de référence. Ceci illustre l'avantage de la structure de commande décentralisée choisie par rapport à la structure de commande centralisée.

➤ Commande par la logique floue dans un espace cartésien

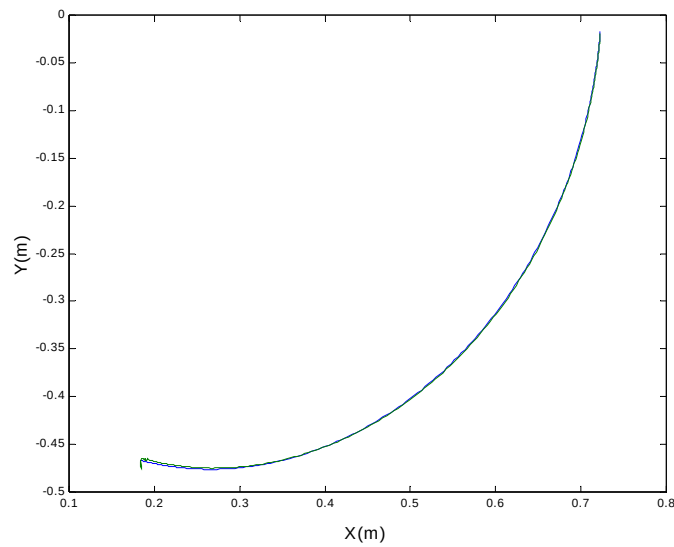
Nous avons également évalué les performances de notre structure de commande dans l'espace cartésien. En effet, nous avons considéré deux configurations: le cas d'un mouvement libre sans charge et le cas d'un mouvement libre puis contraint.

• Mouvement dans un espace cartésien libre sans charge

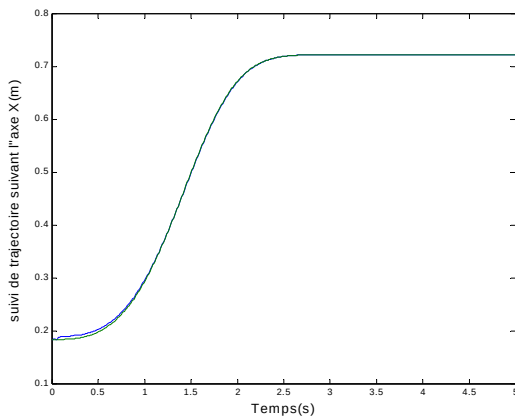
Le but de ces premières simulations, est d'évaluer les erreurs de suivis de trajectoires désirées lors d'un mouvement dans un espace libre sans contrainte d'effort, en utilisant la structure de commande proposée. Les gains assurant un bon suivi de trajectoire sont:

$$\begin{array}{lll}
 G_{u1}=180; & G_{e1}=5\pi/2; & G_{de1}=18\pi/2; \\
 G_{u2}=165; & G_{e2}=1\pi; & G_{de2}=1\pi; \\
 G_{u3}=96; & G_{e3}=2\pi; & G_{de3}=1\pi;
 \end{array}$$

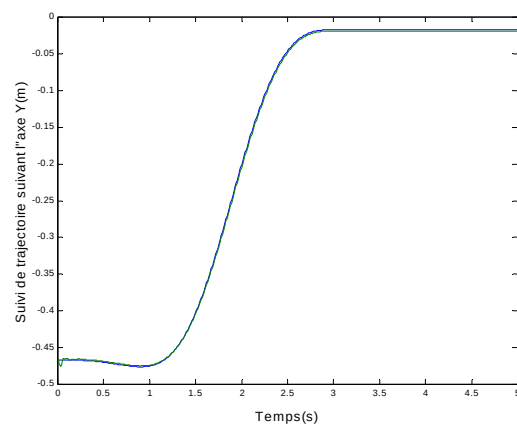
Les résultats obtenus sont consignés sur la figure 12. La figure 12.a donne la trajectoire réelle et désirée de l'organe terminal dans l'espace opérationnel. Les courbes obtenues font apparaître un bon suivi de trajectoire. Les figures 12.b1 et b2 donnent les trajectoires réelles et désirées de l'effecteur dans l'espace tâche respectivement suivant l'axe X et l'axe Y. Ces courbes montrent que le suivi de trajectoire est de bonne qualité. En effet, les erreurs de suivi (figures 12.c1 et c2) initiales sont: $-4 \cdot 10^{-4}$ m suivant l'axe X et $4.1 \cdot 10^{-3}$ m suivant l'axe Y. Ces erreurs sont dues essentiellement à l'inertie des bras et à l'erreur initiale de positionnement du robot. En régime permanent les erreurs sont faibles soit: $3 \cdot 10^{-5}$ m suivant l'axe X et $2.33 \cdot 10^{-3}$ m suivant l'axe Y.



- a -



- b1 -



- b2 -

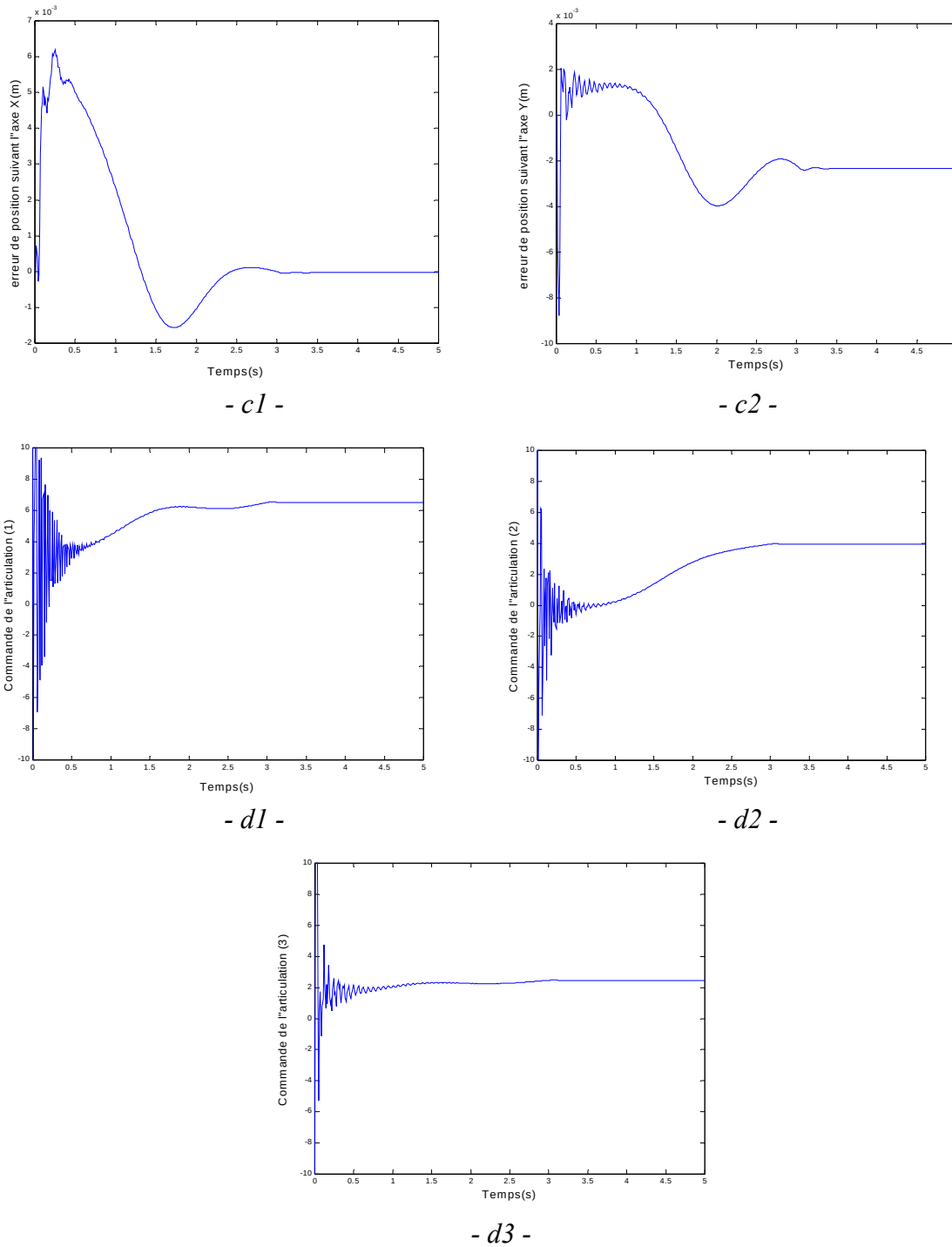


Figure 12. Mouvement dans l'espace libre cartésien du bras manipulateur redondant sans charge.

- a. Trajectoire de l'organe terminal dans le plan (X, Y)
- b1 et b2 Trajectoires suivant l'axe X et l'axe Y
- c1 et c2 Erreur de suivi selon les axes X et Y
- d1, d2, d3 Couple appliqué respectivement aux articulations 1, 2 et 3.

Les figures 12.d1, d2 et d3 représentent les couples appliqués aux différentes articulations du bras manipulateur considéré. Ces courbes montrent qu'au démarrage les valeurs des couples sont élevées soit: 17 N.m pour la première articulation, 16,4 N.m pour la deuxième et 15.5 N.m pour la troisième articulation. En effet, le bras manipulateur étant

initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.51 N.m pour la première articulation, 3.94 N.m pour la deuxième et 2.48 N.m pour la troisième.

- **Mouvement dans un espace cartésien libre puis contraint**

De même que pour la commande dynamique, nous avons étudié le cas de trois tâches différentes:

- a- Le suivi d'un contour rectiligne en maintenant un effort constant.
- b- Le suivi d'un contour curviligne en maintenant un effort constant.
- c- La prise et transport d'un objet d'un point « A » à un point « B ».

Ainsi, nous avons simulé le mouvement du robot manipulateur redondant dans l'espace libre puis dans l'espace contraint. Ce dernier est caractérisé par la présence d'un obstacle de raideur non nulle à une position X_e .

Les simulations ont pour but l'évaluation des erreurs de suivis des trajectoires désirées. Les gains de normalisation et les gains de commandes assurant une dynamique globale satisfaisante sont obtenus expérimentalement.

a. Suivi d'un contour rectiligne en maintenant un effort constant

Dans cette première partie, la tâche assignée au robot sera le suivi d'un mur vertical. Ceci dit, à l'instant $t = 1.75s$, on simule la présence d'un mur vertical. Avec lequel le bras manipulateur rentre en contact en maintenant sur sa surface un effort normal de 50N (figure 13).

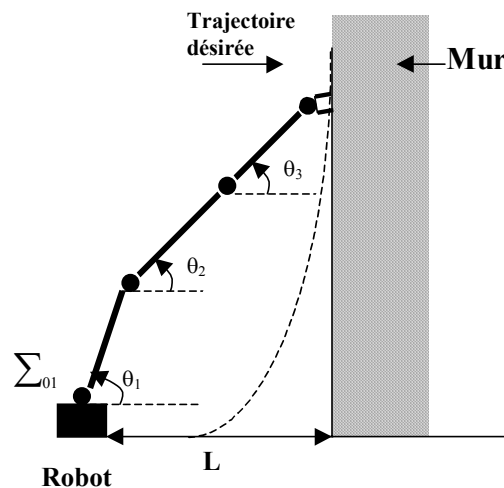


Figure 13. Suivi d'un contour rectiligne en maintenant un effort constant.

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$G_{u1}=240;$	$G_{e1}=2*\pi/2;$	$G_{de1}=8*\pi/2;$
$G_{u2}=125;$	$G_{e2}=1*\pi;$	$G_{de2}=2.6*\pi;$
$G_{u3}=100;$	$G_{e3}=1*\pi;$	$G_{de3}=1*\pi;$

La réponse en position du système en boucle fermée est représentée par la figure 14.

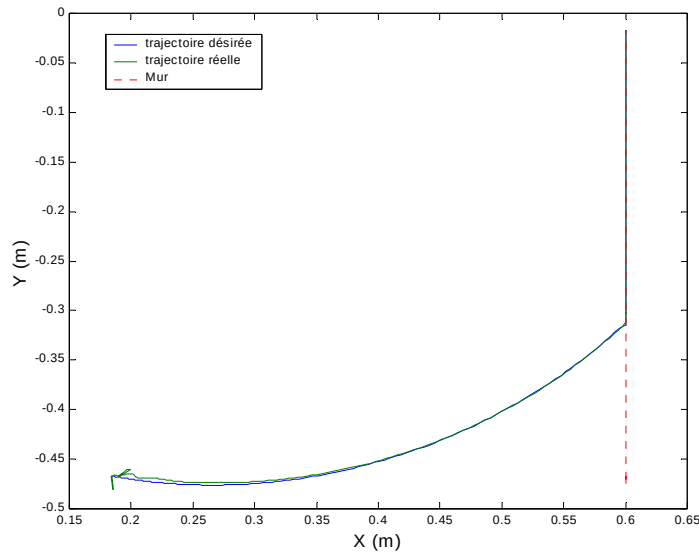
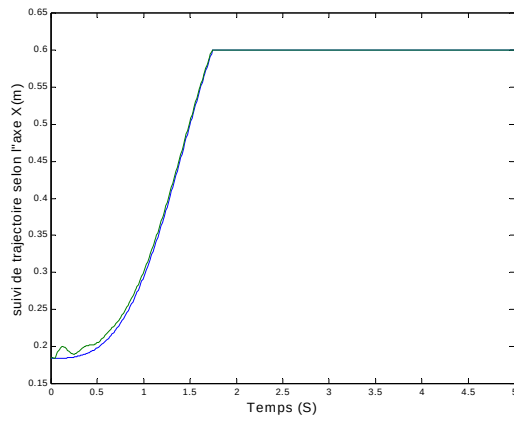


Figure 14. Suivi de contour rectiligne dans le plan (X, Y) .

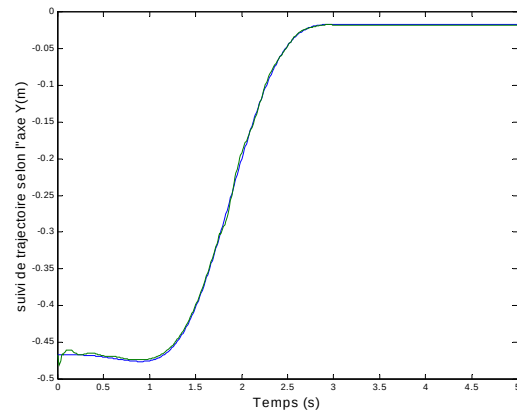
Nous constatons un bon suivi de trajectoire. La présence du mur vertical se traduit par un déplacement tangent à la surface de celui-ci, donc parallèle à l'axe Y . Les figures 15.a1 et a2 illustrent les trajectoires réelles et désirées respectivement suivant l'axe X et suivant l'axe Y . Le suivi selon l'axe X , est satisfaisant (figure 15.a1). En effet, la figure 15.b1 montre que l'erreur initiale est de $-1.5 \cdot 10^{-3}$ m. Cette dernière est due à la position initiale de l'effecteur. Une fois que l'effecteur est en contact avec l'environnement ($t=1.75$ s), l'erreur se stabilise à $-4.87 \cdot 10^{-8}$ m. En effet, à cet instant l'effecteur est contraint par la présence du mur selon cette direction. Suivant l'axe Y (figure 15.b2), l'erreur se stabilise à $4 \cdot 10^{-3}$ m après 1.25s, mais elle présente un pic d'amplitude 0.0109m au moment de mise en contact.

Les figures 15.d1, d2 et d3 présentent les trajectoires articulaires. Ces courbes montrent un bon suivi au cours de tout le mouvement. Toutefois, l'impact du choc fait apparaître, une petite déformation de la trajectoire réelle sur toutes les courbes à l'instant $t = 1.75$ s. Après le contact, la loi de commande en effort permet de compenser cette perturbation, ce qui a permis d'améliorer le suivi.

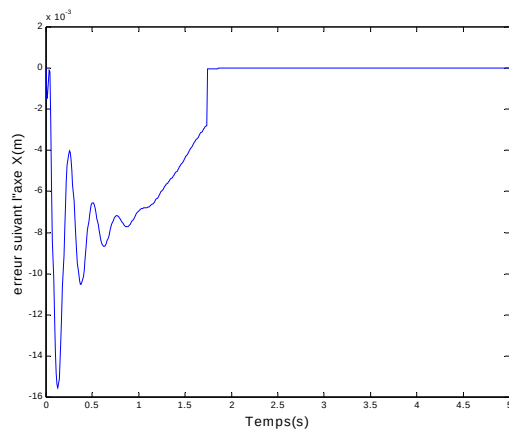
Les figures 15.e1, e2 et e3 présentent les couples appliqués aux articulations du bras manipulateur redondant considéré. Au démarrage les valeurs des couples sont élevées soit 17 N.m, 16.4 N.m et 15.5 N.m respectivement pour la première, deuxième et troisième articulation. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. Lors du contact avec l'environnement, quelques oscillations apparaissent au niveau des couples de commande. Les valeurs maximales des oscillations sont 16.2 N.m, 15.4 N.m et 14.8 N.m respectivement pour la première, deuxième et troisième articulation.



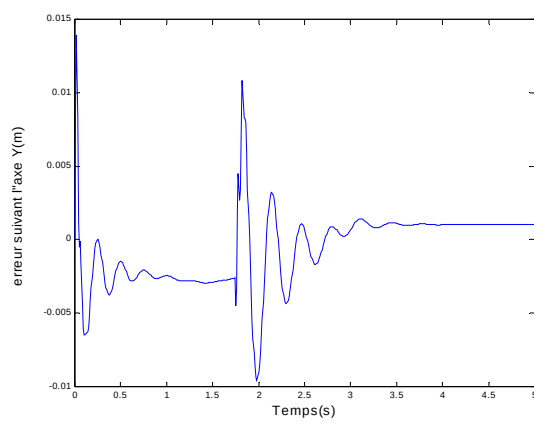
- a1 -



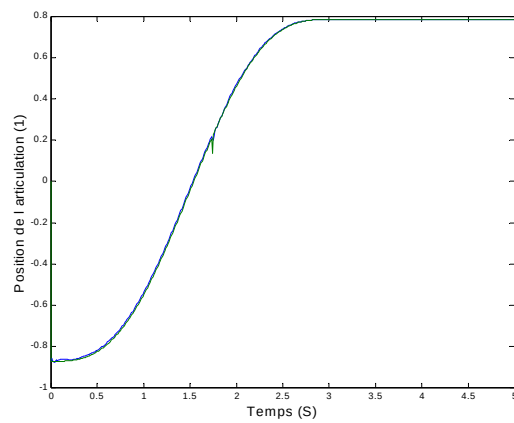
- a2 -



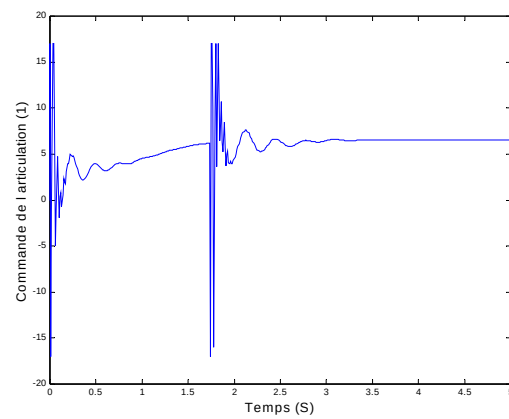
- b1 -



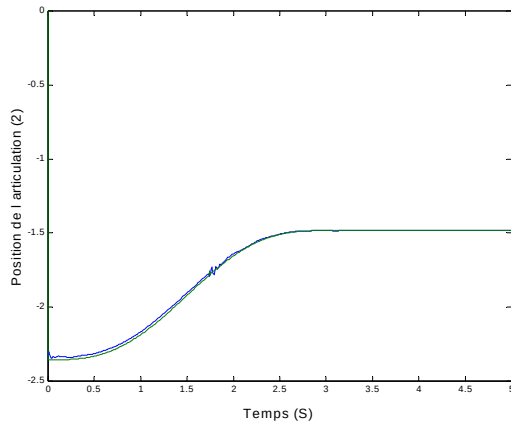
- b2 -



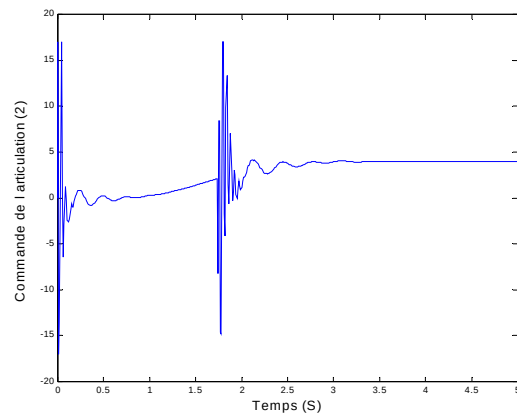
- d1 -



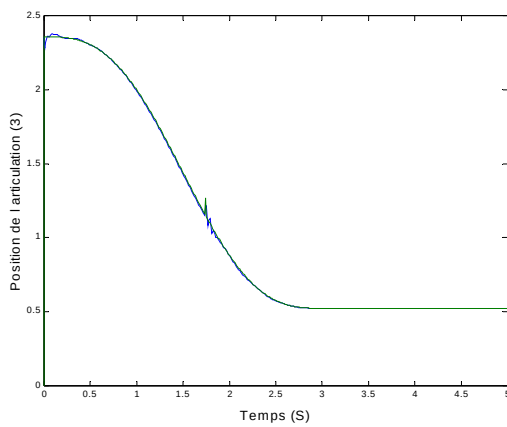
- e1 -



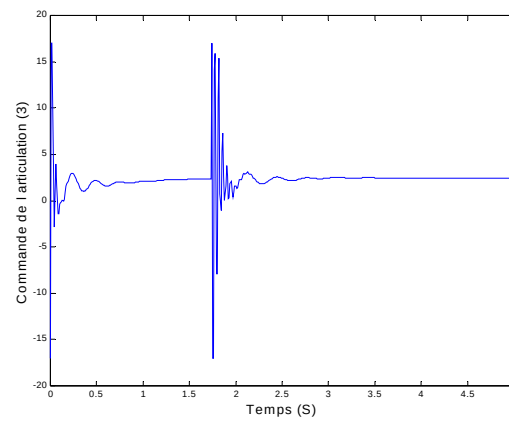
- d2 -



- e2 -



- d3 -

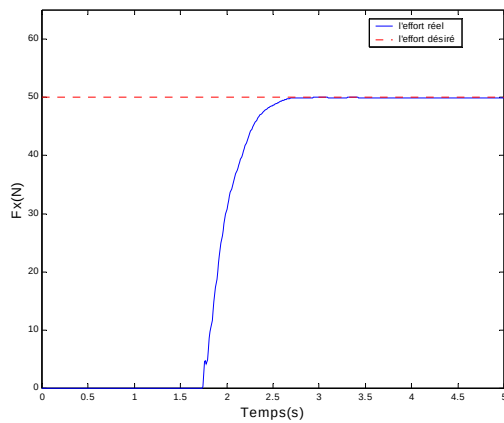


- e3 -

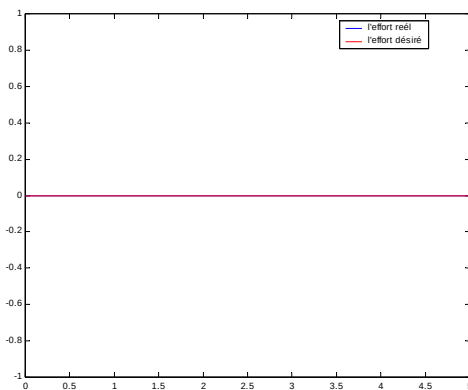
Figure 15. Suivi d'un contour rectiligne en maintenant un effort constant

- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- d1, d2 et d3 Trajectoires respectivement de l'articulation 1, 2 et 3.
- e1, e2 et e3 Couple appliqué respectivement aux articulations 1, 2 et 3.

Les courbes de la figure 16 représentent les réponses en effort obtenues durant l'exécution de cette tâche.



- a1 -



- a2 -

Figure 16. Réponse en effort.

a1 et a2 Effort selon respectivement l'axe X et l'axe Y

Lors du déplacement de l'effecteur dans l'espace libre, aucun effort n'est mesuré ($F_x = F_y = 0$). Cependant, la transition entre le mouvement libre et le mouvement contraint se traduit par l'apparition d'un effort de contact. Dans le cas de notre présent travail les paramètres des régulateurs sont choisis de sorte à minimiser le pic dû au contact. Ceci dans le but de minimiser au maximum les phénomènes d'instabilité engendrés par le contact. Une fois le contact établi, le robot parvient à maintenir la consigne imposée d'amplitude 50N, normale à la surface de l'environnement. A ce moment, les efforts selon l'axe Y (figure 16.a2) sont nuls compte tenu de la tâche considérée. En effet, dans cette première tâche le déplacement de l'effecteur s'effectue suivant cet axe (l'axe « Y »).

Dans le but de mieux évaluer les performances de notre structure de commande, nous proposons de considérer une tâche de suivi d'un mur de forme curviligne. En effet, ce type de trajectoire est complexe et son suivi excite toute la dynamique du robot. De plus, lors du mouvement contraint les efforts de contact peuvent apparaître selon les deux directions X et Y.

b. Suivi d'un contour curviligne en maintenant un effort constant.

Dans cette partie, le robot manipulateur doit se déplacer initialement dans un espace libre puis suivre un mur curviligne. Ceci dit, on simule à l'instant $t = 1.75s$, la présence d'un mur curviligne situé à la position (0.6m, 0.3m), avec lequel il rentre en contact (figure 17).

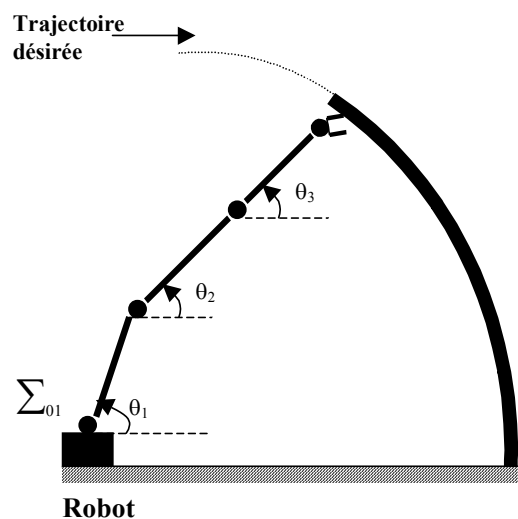
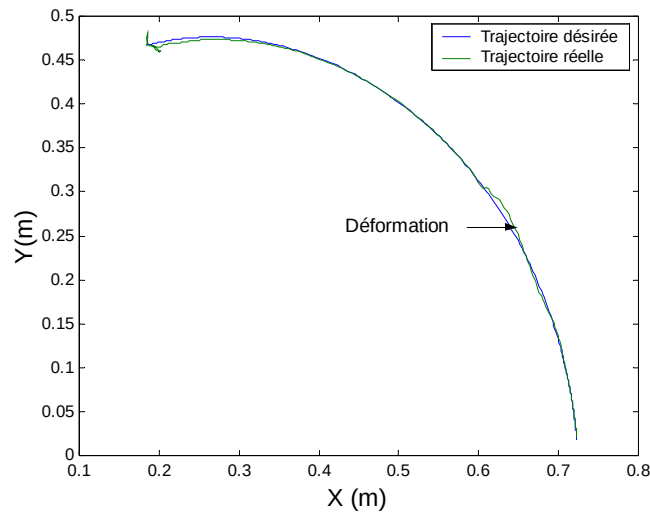


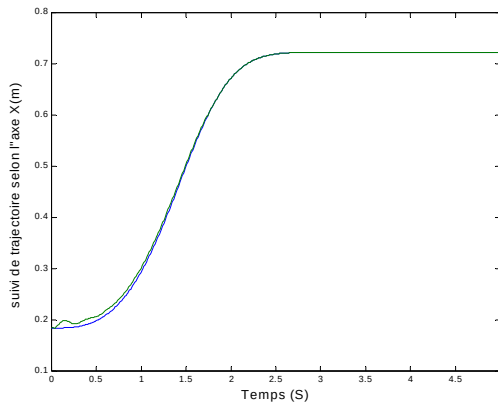
Figure 17. Suivi d'un contour curviligne en maintenant un effort.

Les résultats de simulations obtenus sont consignés sur la figure 18. Dans l'intervalle de temps $[0s, 1.75s]$, le robot manipulateur se déplace en espace libre. Remarquons que dans cet intervalle la poursuite est de bonne qualité. Dès que l'effecteur entre en contact avec le mur curviligne à l'instant « $t=1.75s$ », il suit sa surface en se déplaçant suivant les deux axes X et Y (figure 18.a). L'impact du choc correspond à la petite déformation de la trajectoire réelle qui est visible sur la figure 18.a. ensuite la loi de commande en effort tend à compenser ce dépassement.

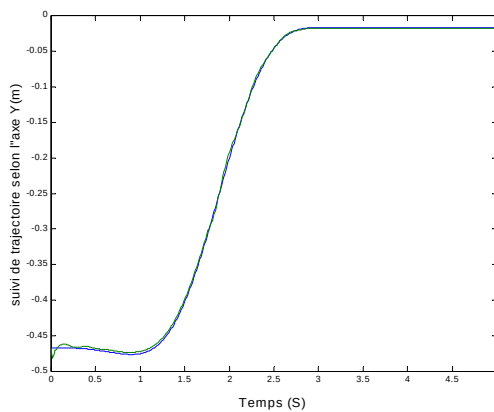
Les figures 18.a1 et a2 illustrent les trajectoires réelles et désirées respectivement suivant l'axe X et suivant l'axe Y. Le suivi selon l'axe X, est satisfaisant. En effet, la figure 18.b1 fait apparaître que l'erreur initiale est $-1.5 \cdot 10^{-3}$ m. Cette erreur est due à la position initiale de l'effecteur. Une fois que l'effecteur est en contact avec l'environnement ($t=1.75$ s), cette erreur décroît jusqu'à atteindre 10^{-8} m. Ceci s'explique par le fait que pour assurer un bon suivi de consignes, l'effecteur prend une configuration lui permettant de maintenir un effort désiré normal à la surface de l'obstacle, tout en garantissant un déplacement suivant les directions non contraintes. La figure 18.b2 montre que le suivi suivant l'axe Y est de bonne qualité. Toute fois, un pic d'amplitude $1.4 \cdot 10^{-2}$ m à $t=1.75$ s apparaît puis l'erreur se stabilise à la valeur $9 \cdot 10^{-4}$ m à partir de l'instant $t = 3$ s. Ceci peut s'expliquer par l'occurrence du contact du bras manipulateur avec l'environnement à cet instant.



- a -



- a1 -



- a2 -

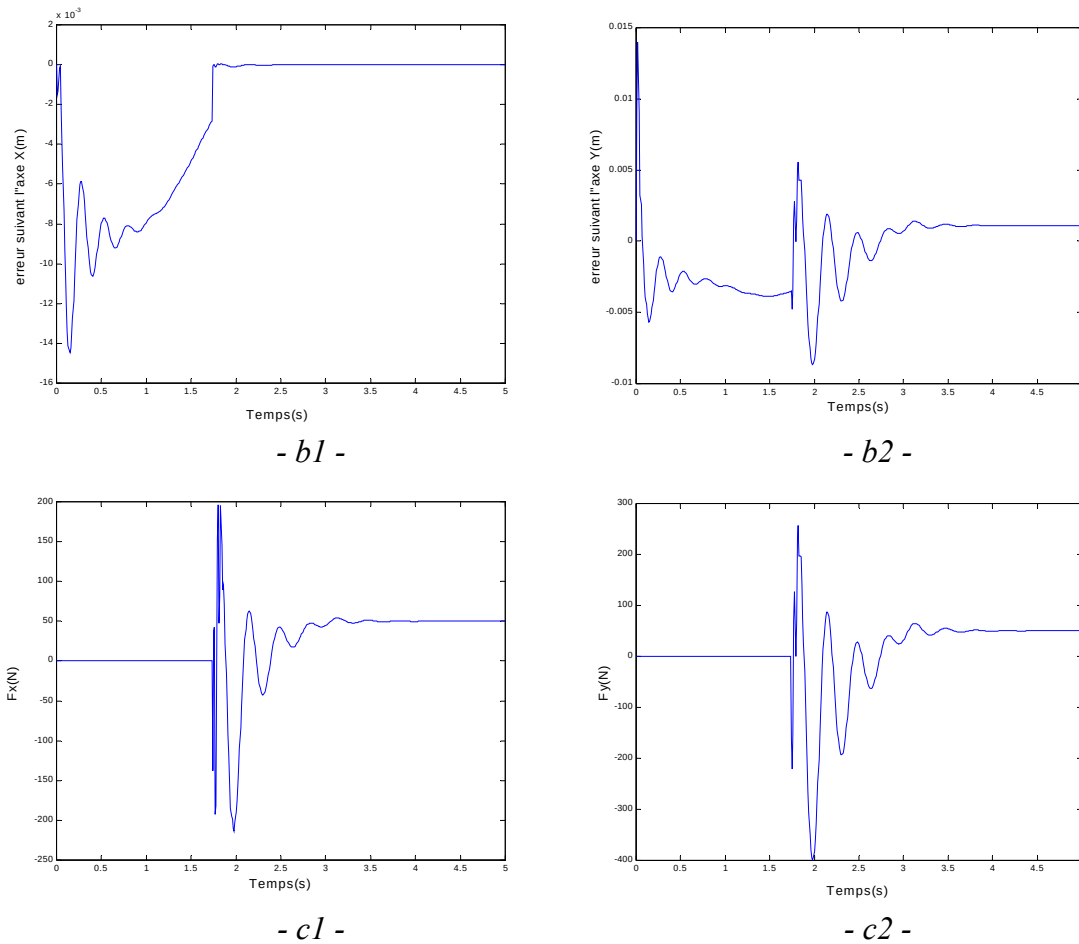


Figure 18. Suivi d'un contour curviligne en maintenant un effort sur sa surface.

- a. Trajectoire de l'organe terminal dans le plan (X, Y)
- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- c1 et c2 Effort selon l'axe X et l'axe Y

La figure 18.d1 et d2 montrent les réponses en effort obtenus durant la poursuite de trajectoire. Avant la collision, le robot se déplace dans l'espace libre, aucun effort n'est mesuré par le capteur de force. En effet, dans l'intervalle de temps $[0 \text{ s}, 1.75 \text{ s}]$, les forces d'interactions suivant l'axe X et l'axe Y sont nulles. A partir de l'instant « $t = 1.75 \text{ s}$ » le bras manipulateur rentre en contact avec l'environnement. En effet, les courbes d'effort obtenues suivant les deux axes présentent un pic suivi de quelques oscillations à cet instant. La modélisation de l'interaction robot-environnement par un ressort explique ce phénomène d'élasticité. Ces oscillations se stabilisent à la valeur 50N à l'instant $t = 3 \text{ s}$. Ceci s'explique par le fait que le contact a eu lieu suivant les deux directions. En effet, l'effort d'interaction appartient à une direction oblique à cette surface curviligne. Elle aura donc deux composantes, chacune suivant un axe comme le montre la figure 19:

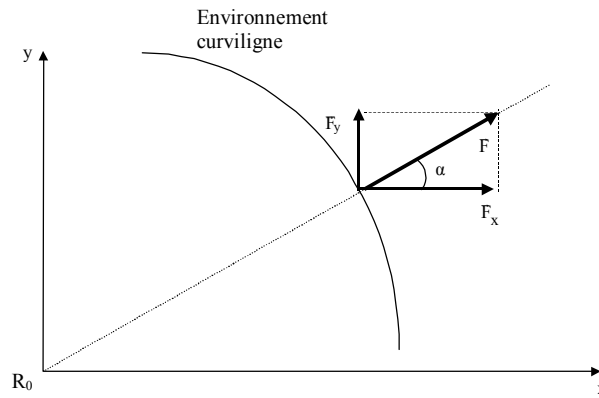


Figure 19. Orientation de l'effort de contact.

c. Saisie et transport d'un objet d'un point « A » à un point « B »

Dans cette tâche, le bras manipulateur doit soulever une pièce. Après une évolution dans l'espace libre, l'effecteur entre en contact avec l'objet et le saisit, en exerçant un effort constant sur sa surface, puis il le déplace du point « A » vers le point « B » (figure 20).

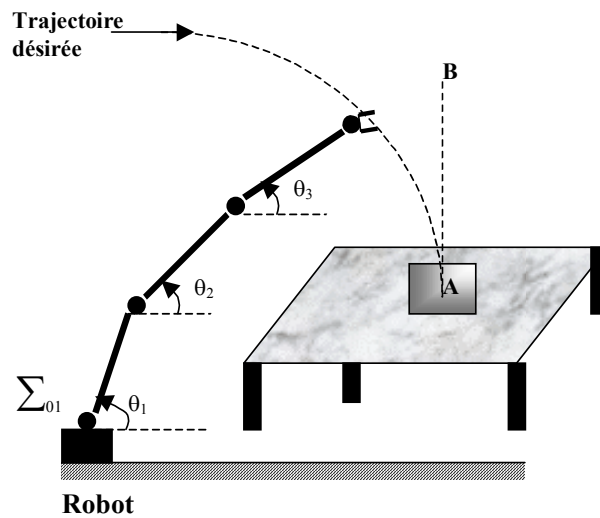
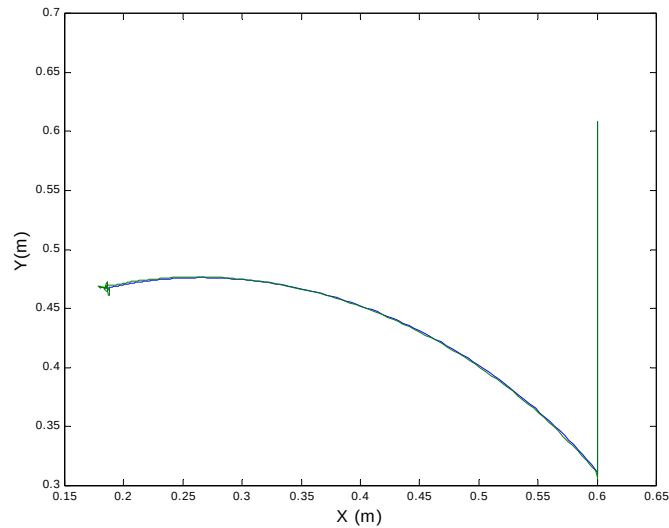


Figure 20. Tâche de prise et transport d'un objet.

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

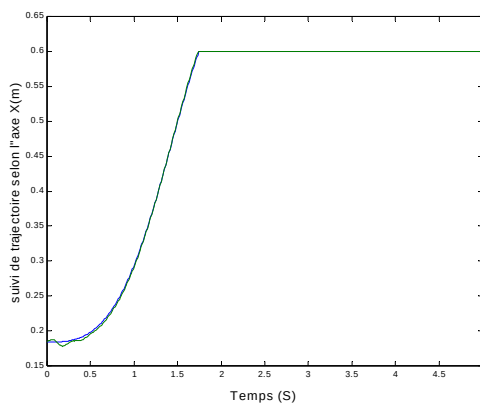
$$\begin{array}{lll} G_{u1}=340; & G_{e1}=2\pi/2; & G_{de1}=8\pi/2; \\ G_{u2}=125; & G_{e2}=1\pi; & G_{de2}=4.6\pi; \\ G_{u3}=90; & G_{e3}=1\pi; & G_{de3}=2\pi; \end{array}$$

Les résultats obtenus sont consignés sur la figure 21.

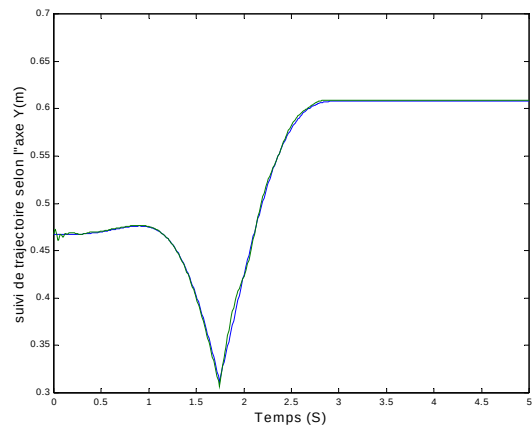


- a -

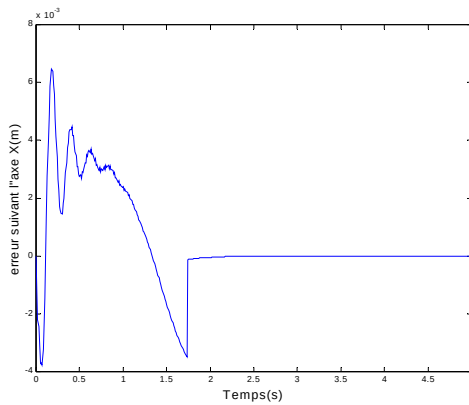
La figure 21.a illustre la trajectoire réelle et désirée. La poursuite est de bonne qualité. Les courbes de suivi selon les deux axes du repère tâche sont présentées sur les figures 21.b1 et figure 21.b2. Les erreurs initiales sont respectivement -2.110^{-3} m et $-8.1 \cdot 10^{-3}$ m selon X et selon Y. Ces erreurs sont dues essentiellement à l'inertie des bras et au fait qu'initialement la position de l'effecteur ne coïncide pas avec la position initiale désirée. Au moment de la mise en contact, l'erreur suivant l'axe X croît brusquement à la valeur $-1.3 \cdot 10^{-3}$ m. Suivant la direction Y a fait apparaître des oscillations. L'erreur cartésienne est maximale à cet instant soit $5.6 \cdot 10^{-3}$ m. ceci s'explique par la mise en contact du bras manipulateur avec l'objet. Ces erreurs se stabilisent en régime permanent à la valeur $8 \cdot 10^{-7}$ m selon l'axe X et à la valeur $-1.7 \cdot 10^{-3}$ m selon l'axe Y.



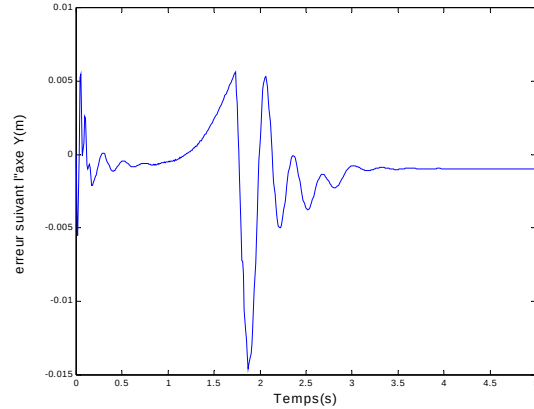
- b1 -



- b2 -



- c1 -

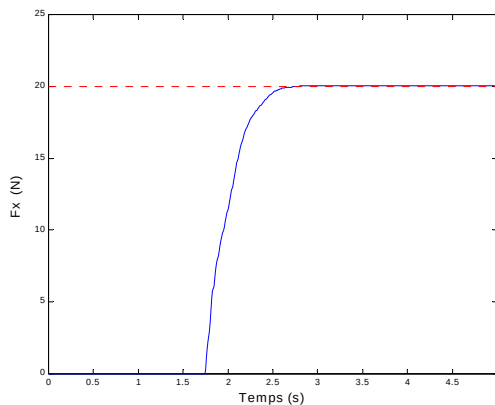


- c2 -

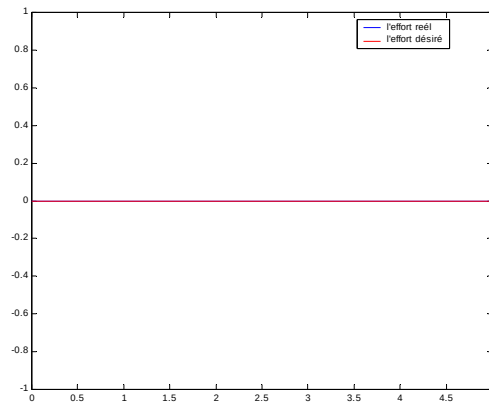
Figure 21. Transport d'un objet par un bras manipulateur redondant.

- a. Trajectoire de l'organe terminal dans le plan (X,Y)
- b1. Trajectoire suivant l'axe X
- b2. Trajectoire suivant l'axe Y
- c1 et c2 Erreur de suivi respectivement suivant l'axe X et l'axe y

La figure 22 illustre les courbes des efforts obtenus.



- a1 -



- a2 -

Figure 22. La réponse en effort.

a1 et a2 Effort respectivement selon l'axe X et l'axe Y

Le mouvement du robot dans l'espace libre annule tout effort de contact. Entre l'instant 0s et 1.75s, les forces d'interactions sont nulles suivant les deux axes. A la mise en contact avec l'environnement, le robot maintient un effort normal de 20N à la surface de l'objet. En effet, à l'instant $t = 1.75s$, les efforts suivant l'axe « X » tendent vers l'effort désiré (20N), elles l'atteignent au bout de 0.75s. . Tandis que les efforts selon l'axe Y restent nuls, compte tenu du déplacement désiré qui s'effectue suivant cet direction, le contact donc a eu lieu suivant la perpendiculaire à la surface de l'environnement.

V.II.6 Conclusion

Dans cette deuxième partie, la théorie de la logique floue et les différentes démarches de conception d'un régulateur flou ont été présentées. Une nouvelle structure de commande hybride force position basée sur la structure de commande en effort externe a été aussi proposée. La particularité de la structure proposée est d'utiliser les concepts de la logique floue lors de la mise en oeuvre des régulateurs. Une telle approche permet de contourner le problème de la connaissance du modèle du système et de son interaction avec l'environnement.

Afin de tester les performances de l'approche de commande proposée nous l'avons implémentée sur un robot redondant à trois liaisons rotoïdes. Différentes tâches sont considérées au cours des simulations réalisées. Ces tâches ont été choisies en fonction de leur complexité et de leur représentabilité dans le domaine de la robotique. Les régulateurs utilisés lors des simulations sont de type Sugeno (5 x 5). Les résultats obtenus pour des différentes tâches considérées sont satisfaisants ceci permet de valider l'approche proposée. En effet, les suivis de trajectoire sont satisfaisant que ce soit en espace libre ou en espace contraint. De plus, le test de la variation brusque de la charge a permis de montrer la robustesse de la structure proposée vis à vis de cette variation.

L'intérêt majeur d'une telle approche réside dans sa simplicité de mise en œuvre. Ceci s'explique par le fait qu'elle ne nécessite pas de modèle dynamique du système. Toutefois, son inconvénient majeur réside dans la détermination des gains de normalisations. En effet, les performances de la commande dépendent fortement de ces gains.

Nous proposons dans la troisième partie suivante d'utiliser les concepts neuro-flous pour la conception des régulateurs de la structure de commande proposée. L'utilisation de l'approche neuro-floue nous permettra de combiner les avantages des réseaux de neurones et des concepts de la logique floue.

Troisième partie

Commande hybride force-position utilisant les concepts des réseaux Neuro-Flous.

Dans cette dernière partie, nous allons fusionner les concepts neuronaux et les concepts flous, en effet, il s'agit d'une représentation neuronale du flou dans laquelle le neuronal joue le rôle du « HARDWARE » pour implanter la logique floue c'est à dire la mise d'un système d'inférence floue sous la forme d'un réseau neuronal adaptatif non linéaire. A partir de ces considérations et des données brutes d'un système (entrée - sortie), le réseau neuronal peut gérer automatiquement les règles floues et ajuster les paramètres du régulateur [Kig 95].

V.III.1 Représentation neuronale des règles floues

Cette méthode d'association est la plus couramment étudiée. Elle représente actuellement plus de trois quarts des travaux existants dans la littérature. L'idée principale est de décomposer l'inférence floue en opérations simples. Chaque opération est calculée par un neurone. L'architecture est analogue à celle d'une perceptron. A chaque couche est associé un type de neurone particulier capable d'effectuer une opération plus ou moins complexe: Calcul du degré d'appartenance, calcul d'une opération logique,...etc. Certains poids sont fixes et d'autres réglables [Chi 94].

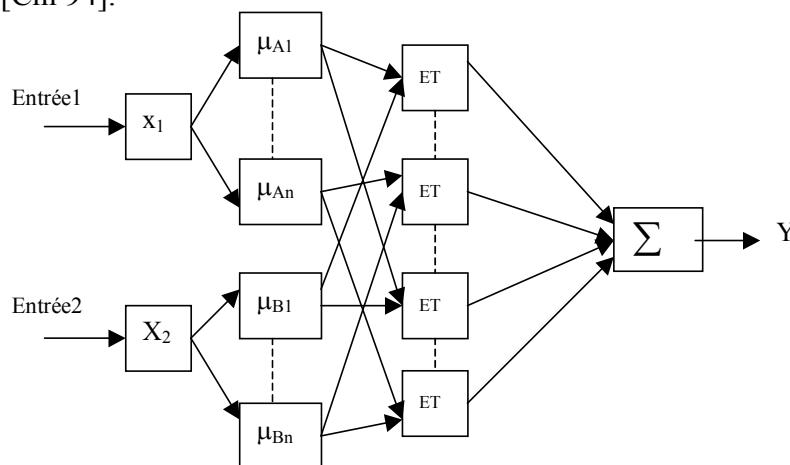


Figure 1. Représentation neuronale des règles floues.

Cette structure figure 1, dite «réseau neuro-flou universel» laisse clairement apparaître l'analogie avec un système de commande flou classique [Chu 95]:

- La couche d'entrée (x_1, x_2) diffuse les entrées vers les neurones contenant les définitions des termes linguistiques.
- La première couche cachée (μ_{A_i}, μ_{B_j}) réalise le calcul des degrés d'appartenance des entrées à chaque sous - ensemble flou.
- La seconde couche cachée «ET» calcule les degrés d'activation a_k de chaque règle.

- Les poids synaptiques w_k représentent les valeurs des conclusions des règles. La sortie Y est calculée en faisant une moyenne pondérée de ces poids par le degré d'activation a_k des règles :

$$Y = \frac{\sum_{k=1}^p a_k \cdot w_k}{\sum_{k=1}^p a_k} \dots\dots\dots (V.12).$$

Avantage et limitations des réseaux neuro-flous

La représentation neuronale des règles floues est avantageuse dans le sens où la structure classique des systèmes flous est conservée, et permet le réglage automatique des poids synaptiques w_i et des paramètres des fonctions d'appartenance (μ_{Ai} , μ_{Bj}) par apprentissage. De plus, la plupart des chercheurs constatent que la convergence de l'apprentissage est nettement plus rapide car le nombre de poids à déterminer est très diminué. L'interprétabilité de ces poids permet leur initialisation non plus d'une manière aléatoire mais à des valeurs proches de l'optimum. Néanmoins, la répartition de la connaissance dans le réseau est perdue, et par conséquent la robustesse aux pannes. Quant à l'apprentissage, il garde ses handicaps : convergence, incertitudes, minima locaux...etc [Chu 94].

V.III.2 Structure de commande

La structure de commande est la même que celle utilisée pour la commande Floue. En remplaçant juste le régulateur flou par un régulateur neuro-flou (figure 2).

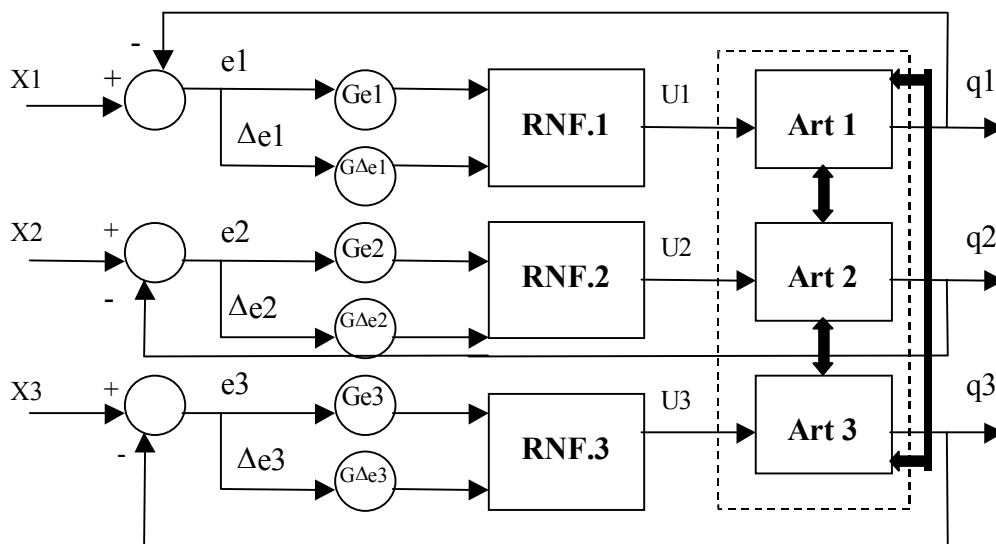


Figure 2. Structure de la commande Neuro-floue décentralisée.

Où RNF.1, RNF.2 et RNF.3 sont des régulateurs neuro-flous correspondant respectivement aux articulations 1, 2 et 3.

Il existe plusieurs approches pour réaliser un système d'inférence flou. Chaque approche a son image dans la représentation par un réseau adaptatif non linéaire (RAN). On peut citer: l'approche de Mamdani et l'approche de Sugeno.

Dans notre étude, nous avons choisi l'approche de Sugeno. Ainsi, nous allons synthétiser trois régulateurs neuro-flous de type Sugeno à trois classes d'appartenance. Ce type de régulateurs présente deux avantages majeurs par rapport à celui de type Mamdani:

- Temps de calcul de la commande réduit.
- La mise sous forme de réseau adaptatif non linéaire est facile:

V.III.3 Synthèse des régulateurs neuro-flous

La structure du régulateur sous forme d'un réseau adaptatif est illustrée par la figure 3 où les nœuds à paramètres variables sont ceux de l'avant dernière couche.

Le régulateur est soumis à un algorithme d'apprentissage pour déterminer les paramètres des polynômes qui donnent les meilleures performances [Wei 97].

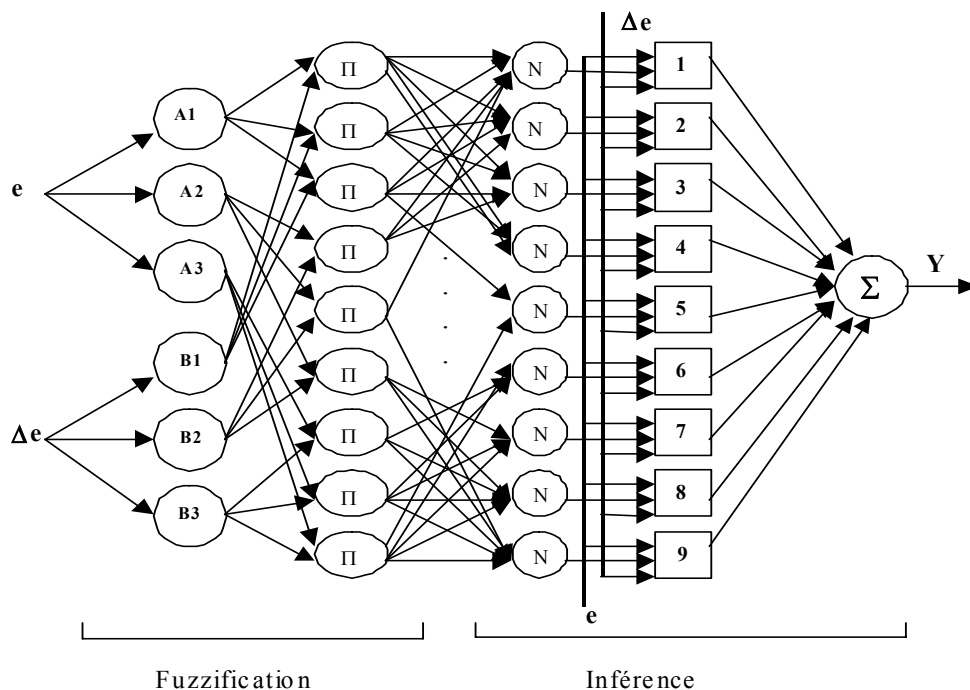


Figure 3. Le régulateur Neuro-flou de type Sugeno (3x3)

V.III.4 Algorithme de commande

Couche1 : Chaque nœud i de cette couche est un nœud ajustable ayant une sortie définie par :

$$O_{1,i} = \mu_{A_i}(e), \text{ pour } i = 1,3 \text{ et } O_{1,i} = \mu_{B_{i-2}}(\Delta e), \text{ pour } i = 4,6$$

Tel que μ_{A_i} et $\mu_{B_{i-2}}$ sont les fonctions d'appartenances de forme triangulaire à paramètres fixes associées aux prémisses.

Couche 2: Les sorties des nœuds de cette couche sont exprimées par:

$$O_{2,1} = O_{1,1} \cdot O_{1,4} = w_1, \quad O_{2,2} = O_{1,3} \cdot O_{1,5} = w_2, \quad O_{2,3} = O_{1,1} \cdot O_{1,6} = w_3$$

$$O_{2,4} = O_{1,2} \cdot O_{1,4} = w_4, \quad O_{2,5} = O_{1,2} \cdot O_{1,5} = w_5, \quad O_{2,6} = O_{1,3} \cdot O_{1,6} = w_6$$

$$O_{2,7} = O_{1,3} \cdot O_{1,4} = w_7, \quad O_{2,8} = O_{1,3} \cdot O_{1,5} = w_8, \quad O_{2,9} = O_{1,3} \cdot O_{1,6} = w_9$$

Couche 3: La sortie du nœud est exprimée par :

$$O_{3,i} = \bar{\omega}_i = \frac{w_i}{\sum_{k=1}^9 w_k}.$$

Couche 4: les nœuds de cette couche sont à paramètres ajustables, leurs fonctions sont de la forme :

$$O_{4,i} = \bar{\omega}_i f_i = \bar{\omega}_i (p_i x + q_i y + r_i) \quad i=1,9$$

où : $\bar{\omega}_i$ est la sortie de la couche 3 et (p_i, q_i, r_i) est l'ensemble des paramètres à ajuster du nœud (4i) .

Couche 5: Cette couche contient un seul nœud qui réalise la somme des sorties des nœuds de la couche précédente :

$$O_5 = \sum_{i=1}^9 O_{4,i}$$

les paramètres des prémisses sont supposés fixes, donc O_5 peut être écrite sous la forme :

$$O_5 = \sum_{i=1}^9 (\bar{w}_i \cdot e) \cdot p_i + (\bar{w}_i \cdot \Delta e) \cdot q_i + (\bar{w}_i) \cdot r_i$$

V.III.5 Algorithme d'apprentissage

Pour notre cas nous avons utilisé un algorithme d'apprentissage basé sur le filtre de Kalman [gouass] étendu. Ce dernier est couramment utilisé pour estimer les paramètres des réseaux neuronaux. Le régulateur est caractérisé par un vecteur de paramètres θ (poids du réseau). Notre objectif est de trouver les valeurs du vecteur θ en minimisant l'erreur suivante :

$$e(k) = y_d(k) - y(k) \dots \dots \dots (V.13).$$

Où $y_d(k)$ et $y(k)$ sont respectivement les états désirés et réels du système à l'itération k.

L'approche du filtre de Kalman étendu consiste à linéariser à tout instant la sortie « y » autour du vecteur estimé $\hat{\theta}$. Ceci revient à écrire :

$$y_d(k) = y(x(k); \hat{\theta}(k-1)) + \Psi^T(k) \cdot (\theta - \hat{\theta}(k-1)). \text{ Avec: } \Psi = \left. \frac{\partial y(x(k); \theta)}{\partial \theta} \right|_{\hat{\theta}(k-1)}$$

La forme, bien connue, de la solution est [gouass] :

$$\begin{aligned} \hat{\theta}(k) &= \hat{\theta}(k-1) + p(k) \cdot \Psi(k) \cdot e(k) \\ e(k) &= y_d(k) - y(x(k); \hat{\theta}(k-1)) \end{aligned} \dots \dots \dots (V.14).$$

Où $p(k)$ est le gain de l'algorithme d'estimation. Dans la méthode du gradient modifiée, le gain est choisi variable. Il est donné par la relation suivante :

$$p(k) = \frac{\alpha_1}{\alpha_2 + \Psi^T(k) \cdot \Psi(k)} ; \alpha_1 > 0 ; \alpha_2 > 0. \dots\dots\dots (V.15).$$

Nous remarquons que cette méthode nécessite le calcul du gradient $\Psi = \frac{\partial y}{\partial \theta}$. En effet ce gradient n'est autre que la méthode de la rétropropagation utilisée dans les réseaux neuronaux. Pour le cas étudié, le vecteur des paramètres est: $\theta_i = [r_i \ p_i \ q_i]^T$. Par conséquent nous avons:

$$\Psi_i = \frac{\partial y}{\partial \theta_i} = \left[\frac{\partial y}{\partial r_i} \quad \frac{\partial y}{\partial p_i} \quad \frac{\partial y}{\partial q_i} \right]^T \dots\dots\dots (V.16).$$

Où: $\frac{\partial y}{\partial r_i} = \frac{\alpha_1}{\sum_{i=1}^M \alpha_i}$, $\frac{\partial y}{\partial p_i} = \frac{\alpha_1 \cdot e_i}{\sum_{i=1}^M \alpha_i}$ et $\frac{\partial y}{\partial q_i} = \frac{\alpha_1 \cdot \Delta e_i}{\sum_{i=1}^M \alpha_i}$ Avec M le nombre de règles floues.

L'application de cet algorithme dans ce cas est simple.

$$e_i = y_d - y$$

$$\Psi_i = [\theta_{3,i} \quad \theta_{3,i} \cdot e_i \quad \theta_{3,i} \cdot \Delta e_i]^T$$

Les paramètres sont ajustés par la relation suivante :

$$\theta_i = \theta_{i-1} + p_i \cdot \Psi_i \cdot e_i \dots\dots\dots (V.17).$$

Avec: $p_i = \frac{\alpha_1}{\alpha_2 + \Psi_i^T \cdot \Psi_i}$; pour $i = 1, 9$

α_1 , α_2 sont les gains d'adaptation pour varier la vitesse de convergence ; sont fixes et faibles.

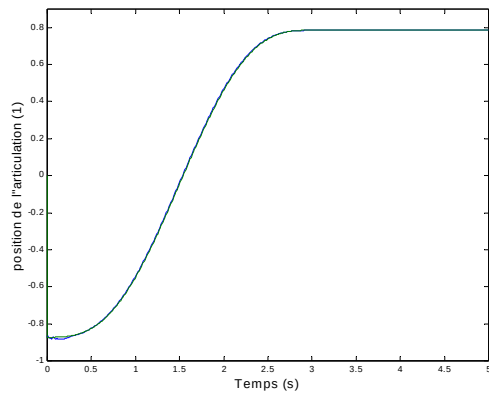
V.III.6 Simulation de la commande par réseaux neuro-flous

Afin de valider l'approche proposée, différentes configurations de mouvements articulaires et cartésiens sont considérées. Lors des simulations, les trajectoires désirées sont ceux de LEAHVY déjà défini dans les deux premières parties de ce chapitre.

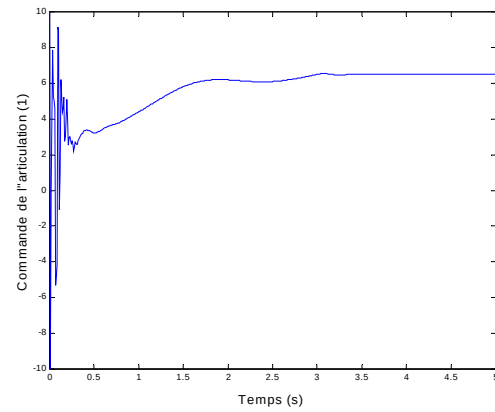
➤ Simulation de la commande par réseau neuro-flou dans l'espace articulaire

• Mouvement dans un espace libre sans charge

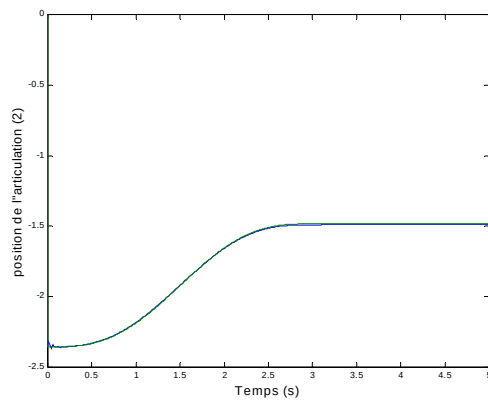
Les réponses articulaires du système sont illustrées sur la figure 4. En examinant les figures nous remarquons que les suivis articulaires réalisés sont satisfaisants (figure 4.a1, a2 et a3). Les erreurs initiales sont maximales au démarrage pour les trois articulations (figure 4.c1, c2 et c3) soit -8.10^{-3} rad pour la première articulation, -0.02 rad pour la deuxième et 0.031 rad pour la troisième. Ceci est dû à l'erreur initiale de position des articulations. En régime permanent les erreurs sont: $2.25 \cdot 10^{-6}$ rad pour la première articulation, 0.0022 rad pour la deuxième et $9.62 \cdot 10^{-4}$ rad pour la troisième.



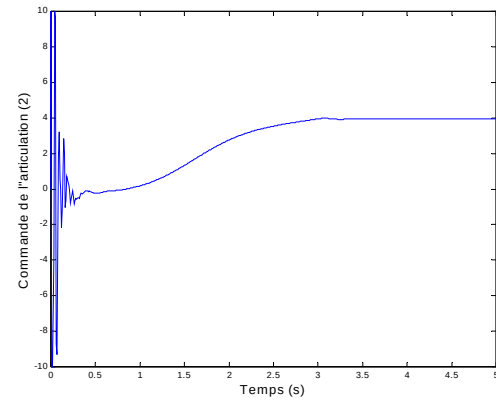
- a1 -



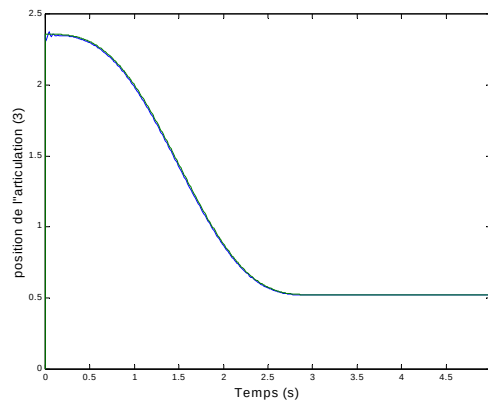
- c1-



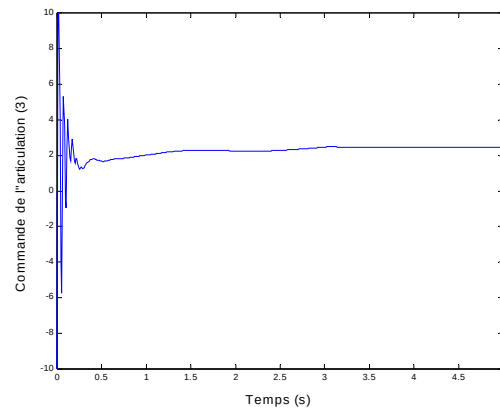
- a2 -



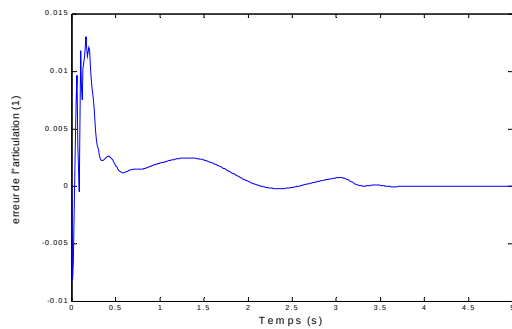
- c2-



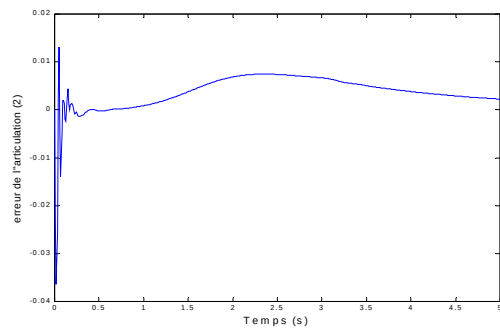
- a3 -



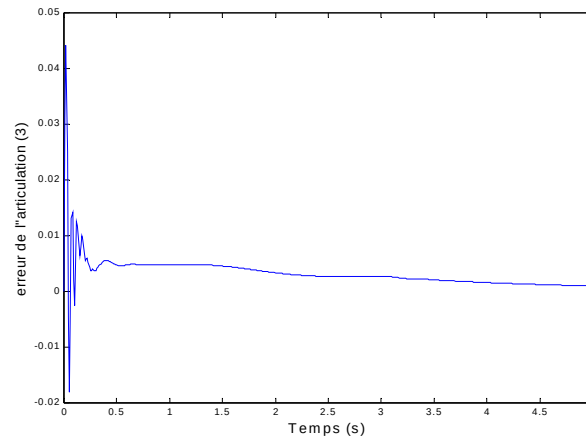
- c3-



- b1 -



- b2-



- b3 -

Figure 4. Mouvement dans un espace libre sans charge.

- a1, a2 et a3 Trajectoires des articulations 1, 2 et 3
- b1, b2 et b3 Erreur de suivi des articulations 1, 2 et 3
- c1, c2 et c3 Couple appliqué respectivement aux articulations 1, 2 et 3

Les figures 4.b1, b2 et b3 illustrent les couples appliqués aux articulations du bras manipulateur redondant considéré. Ces courbes montrent au démarrage les couples élevés soit : 10 N.m pour la première articulation, 8,4 N.m pour la deuxième et en fin de 8.5 N.m pour la troisième. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.50 N.m pour la première articulation, 3.95 N.m pour la deuxième et 2.46 N.m pour la troisième. Nous pouvons aussi constater qu'il existe une relation quasi-linéaire entre les erreurs articulaires et les vecteurs de commandes. Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$$\begin{aligned} G_{u1} &= 190; & G_{e1} &= 0.8 \cdot \pi; & G_{de1} &= 4 \cdot \pi; \\ G_{u2} &= 140; & G_{e2} &= 1.6 \cdot \pi; & G_{de2} &= 2 \cdot \pi; \\ G_{u3} &= 160; & G_{e3} &= 1.1 \cdot \pi; & G_{de3} &= 4 \cdot \pi; \end{aligned}$$

$$k_{11}=0.1; \quad k_{12}=1.7; \quad k_{21}=0.1 \cdot k_{11}; \quad k_{22}=k_{12}; \quad k_{31}=0.1 \cdot k_{11} \text{ et } k_{32}=k_{12};$$

Les résultats obtenus sont satisfaisants, Ces premiers résultats nous permettent de valider l'approche de la commande proposée. Afin de tester la robustesse de la commande proposée nous avons effectué une seconde expérience qui consiste à étudier le mouvement du même bras manipulateur avec une charge à son extrémité.

• Mouvement dans un espace libre avec charge

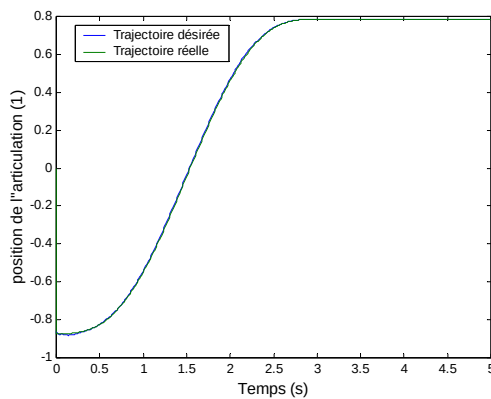
Dans cette partie, la charge portée par le bras manipulateur n'est plus négligeable, elle est égale à 1.5Kg. Nous avons simulé le démarrage du mouvement du robot manipulateur considéré, avec cette charge à son extrémité. Cette charge sera lâchée au bout de trois secondes. La figure 5 présente les résultats obtenus. Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$$\begin{aligned} G_{u1} &= 260; & G_{e1} &= 0.8 \cdot \pi; & G_{de1} &= 4 \cdot \pi; \\ G_{u2} &= 140; & G_{e2} &= 1.6 \cdot \pi; & G_{de2} &= 2 \cdot \pi; \\ G_{u3} &= 160; & G_{e3} &= 1.1 \cdot \pi; & G_{de3} &= 4 \cdot \pi; \end{aligned}$$

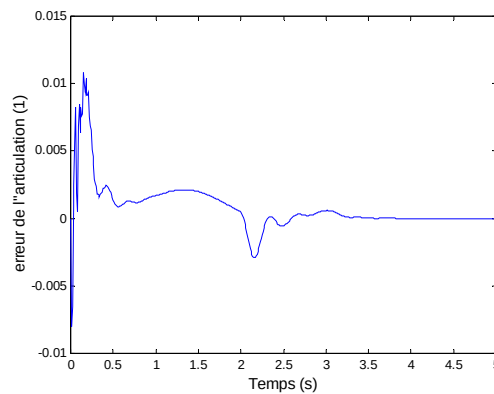
$$k_{11}=0.1; \quad k_{12}=1.7; \quad k_{21}=0.1*k_{11}; \quad k_{22}=k_{12}; \quad k_{31}=0.1*k_{11} \quad \text{et} \quad k_{32}=k_{12};$$

Les figures 5.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement de la première, deuxième et troisième articulation. Ces courbes montrent que le suivi de trajectoire est satisfaisant. En effet, Les courbes des erreurs de suivi articulaires (figures 5.b1, b2 et b3) montrent que les erreurs initiales sont: -8.10^{-3} rad pour la première articulation, -0.019 rad pour la deuxième et 0.028 rad pour la troisième. Les erreurs articulaires sont essentiellement dues à l'inertie du bras et à l'erreur de positionnement initial. A l'instant « $t = 2s$ », les courbes des erreurs articulaires font apparaître de faibles oscillations. Les erreurs articulaires sont alors: $3 \cdot 10^{-3}$ rad pour la première, $5.5 \cdot 10^{-3}$ rad pour la deuxième et 10^{-3} rad pour la troisième articulation. Ceci s'explique par la perte de la charge. Ce phénomène agit sur le système comme une perturbation. Les erreurs articulaires se stabilisent de nouveau à $t = 2.65s$ aux valeurs: $-1.65 \cdot 10^{-6}$ rad pour la première articulation, $2.1 \cdot 10^{-3}$ rad pour la deuxième et $7.5 \cdot 10^{-4}$ rad pour la troisième.

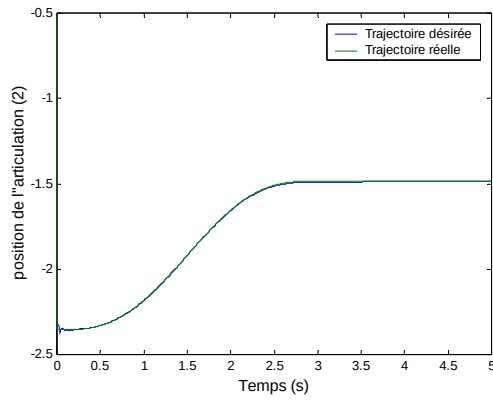
Les figures 5.c1, c2 et c3 donnent les couples appliqués aux différentes articulations du bras manipulateur redondant utilisé. Au démarrage les valeurs des couples sont élevées soit : 20 N.m pour la première articulation, 18.3 N.m pour la deuxième et de 17 N.m pour la troisième. Ces couples sont sensiblement plus élevés que ceux obtenus lors d'un mouvement libre sans charge. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras avec la charge. En régime permanent, à « $t = 2s$ » de faibles oscillations apparaissent. Les valeurs maximales des couples à cet instant sont respectivement de: 7 N.m pour la première articulation, 4.2 N.m pour la deuxième et de 2.75 N.m pour la troisième. Ceci s'explique par la perte de la charge qui agit sur le système comme une perturbation. Les vecteurs commandes finissent par se stabiliser à l'instant « $t = 2.65s$ » à la valeur de 6.50 N.m pour la première articulation, 3.96 N.m pour la deuxième et 2.49 N.m pour la troisième.



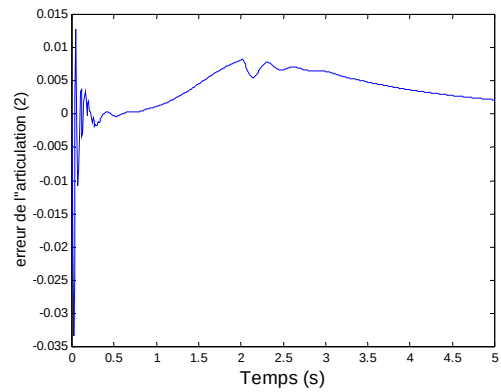
- a1 -



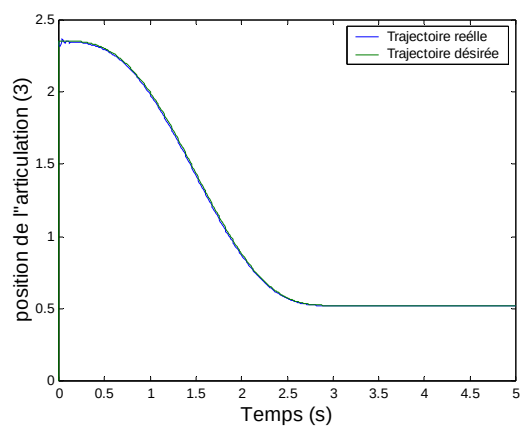
- b1 -



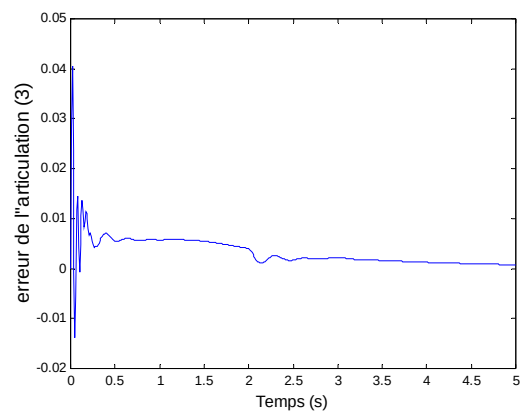
- a2 -



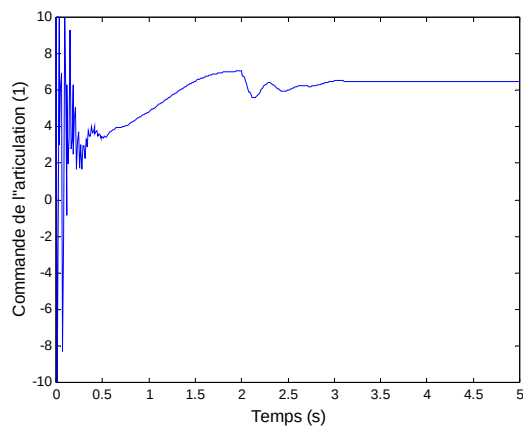
- b2-



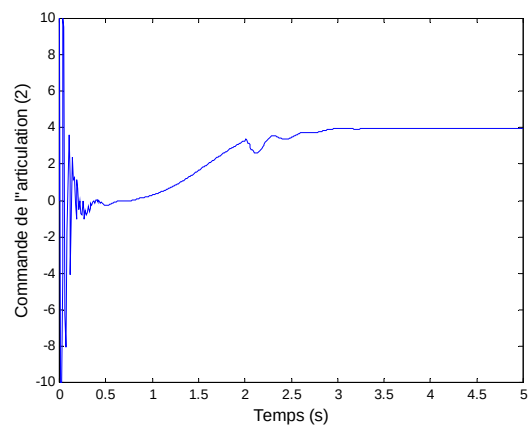
- a3 -



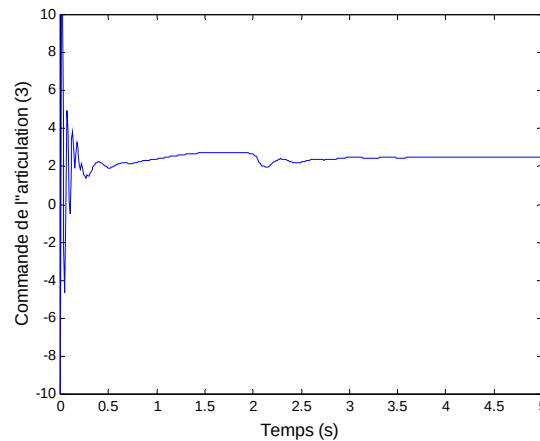
- b3-



- c1 -



- c2-



- c3 -

Figure 5. Mouvement dans l'espace libre du bras manipulateur redondant soumis à une charge initiale de 1.5 Kg variant instantanément ($t=2s$) à 0 Kg

- $a1, a2$ et $a3$ Trajectoires des articulations 1, 2 et 3
- $b1, b2$ et $b3$ Erreur de suivi des articulations 1, 2 et 3
- $c1, c2$ et $c3$ Couple appliqué respectivement aux articulations 1, 2 et 3

Les résultats obtenus montrent la robustesse de la commande vis à vis de la variation de la charge. En effet, la commande proposée permet de compenser les perturbations engendrées par la chute de la charge pendant le mouvement.

Les deux expériences précédentes, permettent de valider la structure de commande à base des concepts neuro-flous proposée lors d'un mouvement non contraint en effort d'un bras manipulateur redondant avec ou sans charge. Nous nous intéresserons dans la suite de notre travail au comportement de la structure de commande dans l'espace cartésien.

• Cas d'une défaillance de l'un des régulateurs (ou actionneur) des bras du robot manipulateur redondant

Dans cette partie, nous simulons à l'instant $t = 2s$ une défaillance sur le régulateur de la troisième articulation. Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

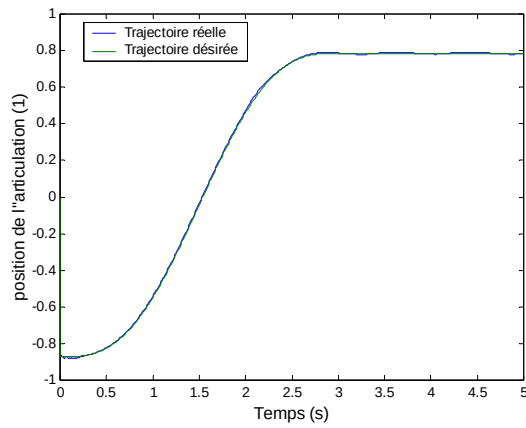
$$\begin{aligned} G_{u1} &= 260; & G_{e1} &= 0.8 \cdot \pi; & G_{de1} &= 4 \cdot \pi; \\ G_{u2} &= 140; & G_{e2} &= 1.6 \cdot \pi; & G_{de2} &= 2 \cdot \pi; \\ G_{u3} &= 160; & G_{e3} &= 1.1 \cdot \pi; & G_{de3} &= 0.4 \cdot \pi; \end{aligned}$$

$$k_{11} = 0.1; \quad k_{12} = 1.7; \quad k_{21} = 0.1 \cdot k_{11}; \quad k_{22} = k_{12}; \quad k_{31} = 0.1 \cdot k_{11} \quad \text{et} \quad k_{32} = k_{12};$$

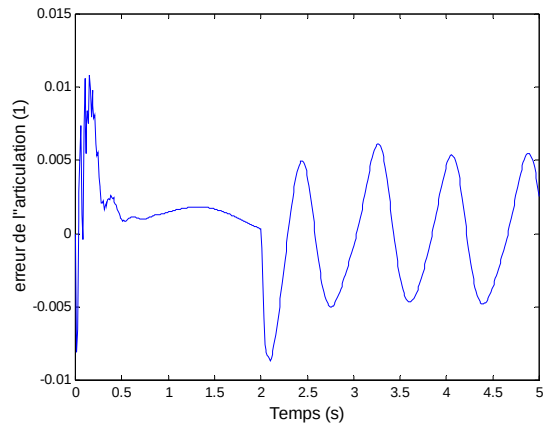
Les figures 6.a1, a2 et a3 donnent les trajectoires réelles et désirées respectivement de la première, deuxième et troisième articulation. Ces courbes montrent un suivi de trajectoires satisfaisant pour la première et la deuxième articulation. Dans le cas de la troisième articulation une divergence apparaît à partir de l'instant « $t = 2s$ ». En effet, à cet instant une défaillance de ce régulateur est introduite.

Les résultats de simulation obtenus montrent que malgré la divergence de la troisième articulation causée par la défaillance introduite sur le troisième régulateur, les autres

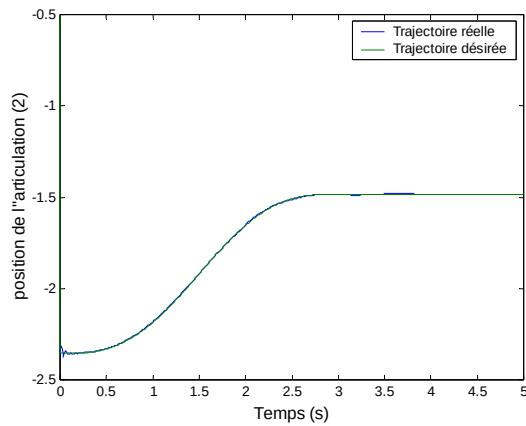
articulations suivent leurs trajectoires de référence. Ceci illustre l'avantage de la structure de commande décentralisée choisie par rapport à la structure de commande centralisée.



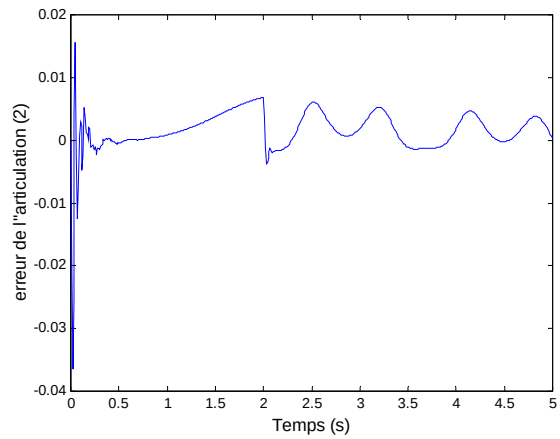
- a1 -



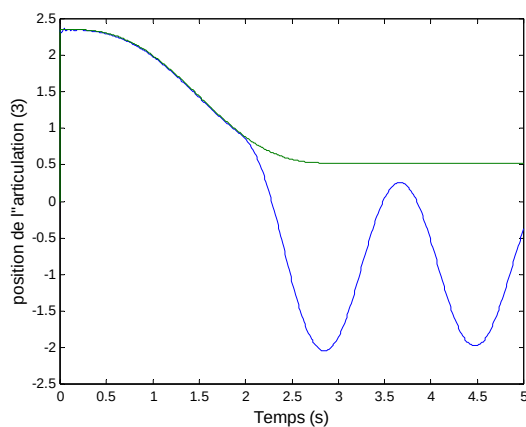
- b1 -



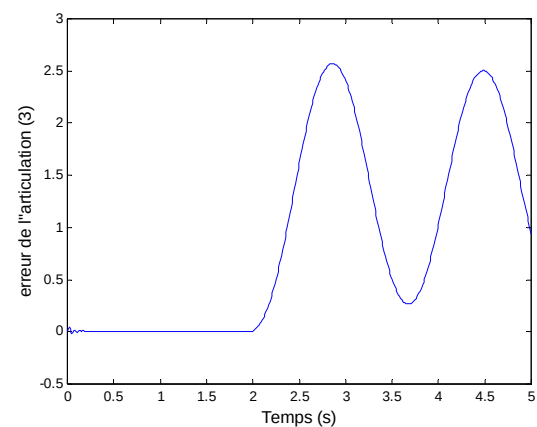
- a2 -



- b2 -



- a3 -



- b3 -

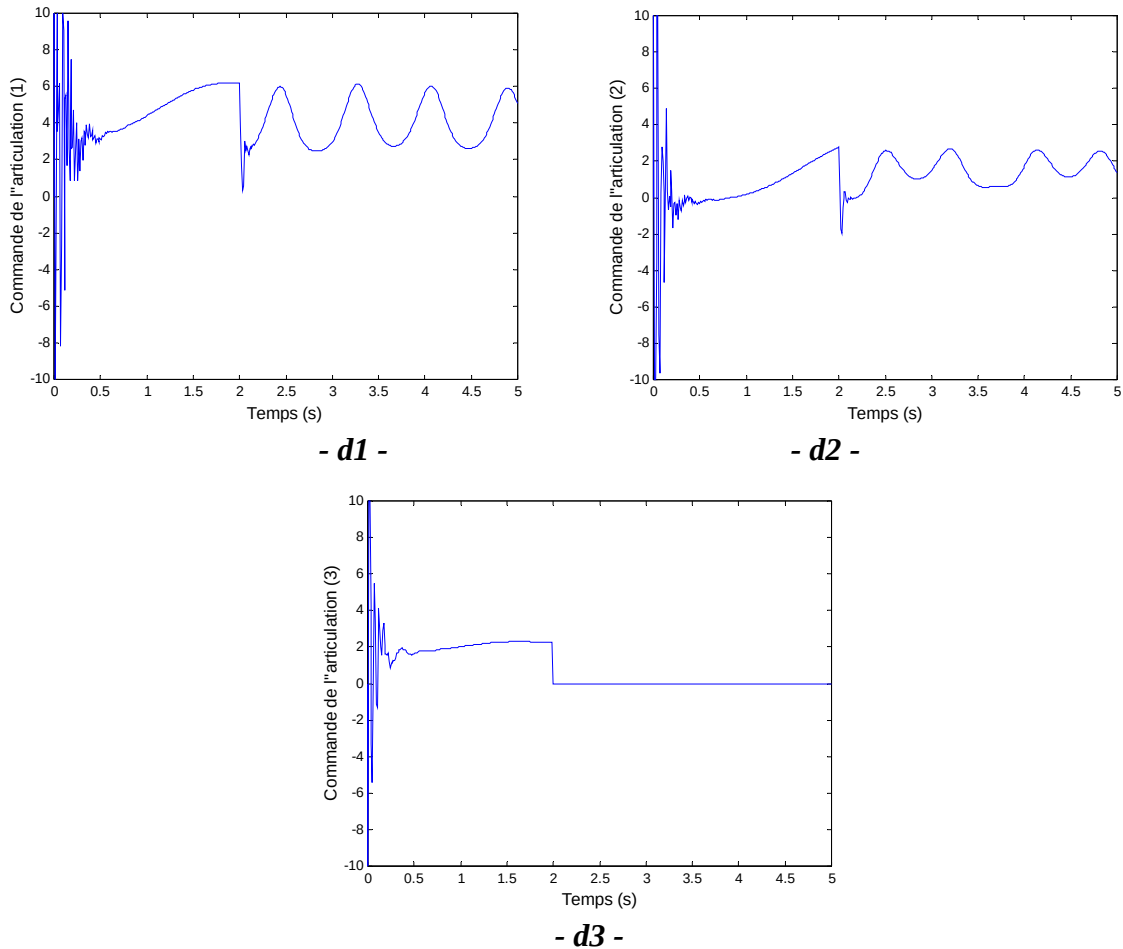


Figure 6. Mouvement dans l'espace libre du bras manipulateur redondant sans charge cas d'une rupture de la commande sur la troisième articulation

- a_1, a_2 et a_3 Trajectoires des articulations 1, 2 et 3
- b_1, b_2 et b_3 Erreur de suivi des articulations 1, 2 et 3
- c_1, c_2 et c_3 Couple appliqué respectivement aux articulations 1, 2 et 3

➤ Simulation de la commande par réseau neuro-flou dans l'espace cartésien

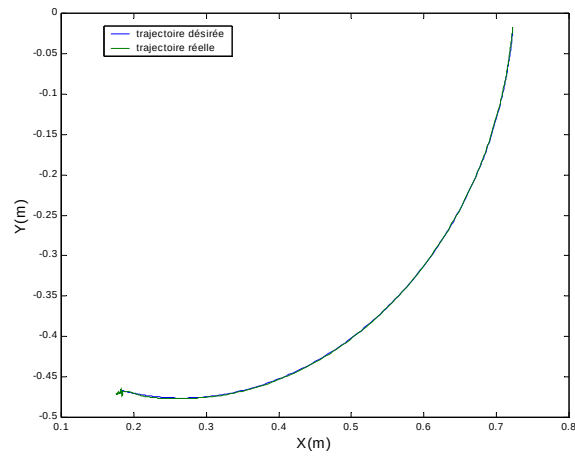
Dans cette partie, nous avons évalué les performances de la commande proposée pour des tâches spécifiées dans un espace cartésien sans contraintes et avec contraintes.

• Mouvement dans un espace cartésien libre sans charge

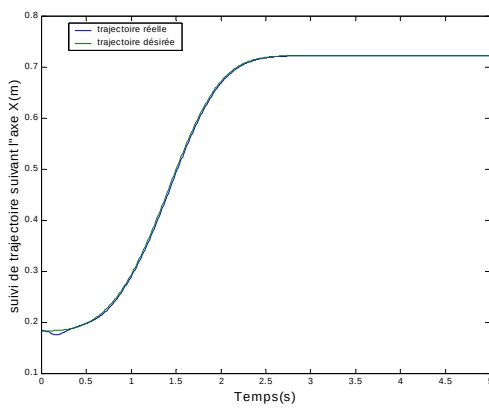
Le but de ces simulations est d'évaluer les erreurs de poursuite de la trajectoire par le robot manipulateur considéré lors d'un mouvement dans un espace libre sans contrainte d'effort, en utilisant la structure de commande neuro-floue. Le réglage des gains de normalisations assurant un bon suivi de trajectoire est effectué expérimentalement. Les résultats obtenus pour cet essai à vide sont consignés sur la figure 7.

La figure 7.a. fait apparaître un bon suivi de trajectoire de l'organe terminal dans l'espace opérationnel. En effet, les suivis selon X et Y sont de bonne qualité (figure 7.b1 et b2). Sur les courbes des figures 7.c1 et c2 l'erreur initiale est $-4 \cdot 10^{-4}$ m suivant l'axe X et $-4.1 \cdot 10^{-3}$

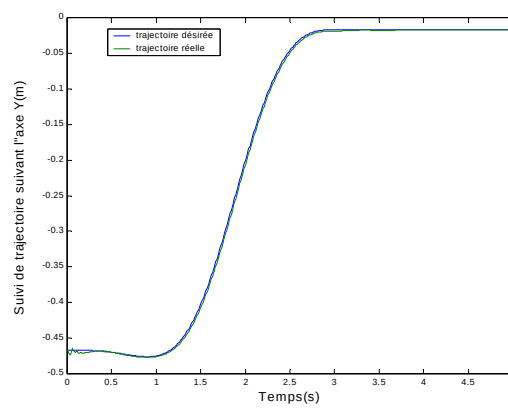
m suivant l'axe Y. Ces erreurs sont dues essentiellement à l'inertie des bras et à l'erreur de positionnement initial du robot. En régime permanent les erreurs sont faibles ($1.45 \cdot 10^{-5}$ m suivant l'axe X, $3.28 \cdot 10^{-5}$ m suivant l'axe Y).



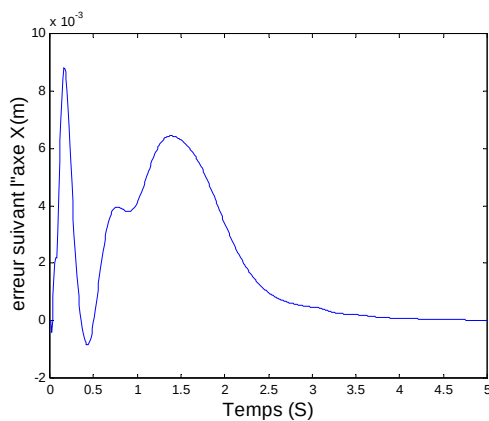
- a -



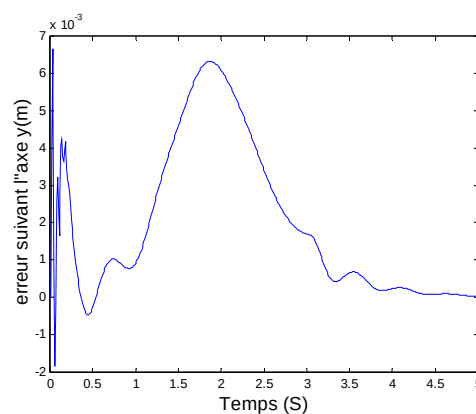
- b1 -



- b2 -



- c1 -



- c2 -

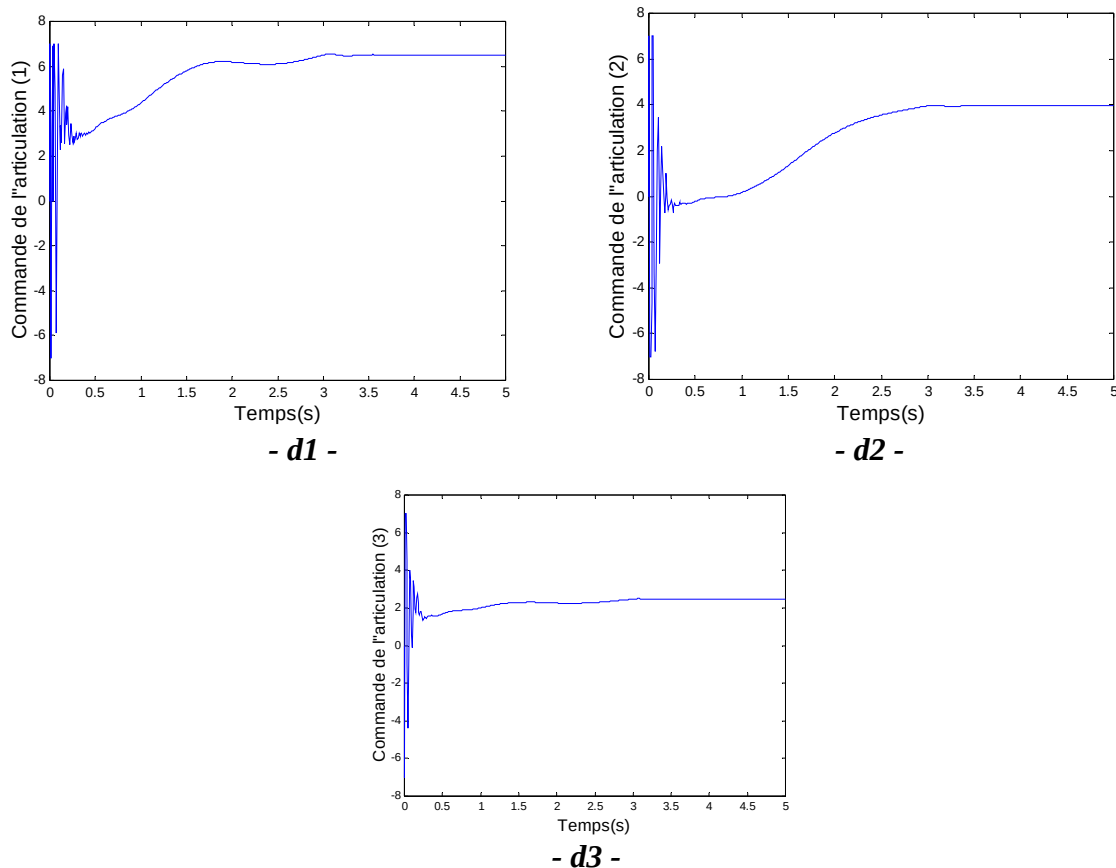


Figure 7. Mouvement dans l'espace libre cartésien du bras manipulateur redondant sans charge.

- a. Trajectoire de l'organe terminal dans le plan (X, Y)
- b1 et b2 Trajectoires suivant l'axe X et l'axe Y
- c1 et c2 Erreur de suivi respectivement selon les axes X et Y
- d1, d2 et d3 Couple appliqué respectivement aux articulations 1, 2 et 3

Les figures 6.d1, d2 et d3 représentent les couples articulaires. Au démarrage les valeurs des couples sont élevées soit: 17 N.m pour la première articulation, 16,4 N.m pour la deuxième et 15.5 N.m pour la troisième. En effet, le bras manipulateur étant initialement au repos il est nécessaire d'appliquer des couples élevés afin de vaincre l'inertie des bras. En régime permanent, aucune oscillation n'apparaît. Les vecteurs commandes atteignent respectivement la valeur de 6.51 N.m pour la première articulation, 3.94 N.m pour la deuxième et 2.48 N.m pour la troisième.

• Mouvement dans un espace cartésien libre puis contraint

De même que pour la commande dynamique et la commande par la logique floue, nous avons étudié le mouvement du robot manipulateur redondant durant l'exécution de trois tâches soit: la saisie et le transport d'un objet, le suivi d'un contour rectiligne et le suivi d'un contour curviligne en maintenant un effort constant.

a. Suivi d'un contour rectiligne en maintenant un effort constant

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$$G_{u1}=180; \quad G_{e1}=6\pi/2; \quad G_{de1}=6.2\pi/2;$$

$$\begin{aligned} G_{u2} &= 200; & G_{e2} &= 0.8 \cdot \pi; & G_{de2} &= 1.4 \cdot \pi; \\ G_{u3} &= 70; & G_{e3} &= 1 \cdot \pi; & G_{de3} &= 3.9 \cdot \pi; \end{aligned}$$

$$k_{11} = .18; \quad k_{12} = 0.81; \quad k_{21} = 0.18 \cdot k_{11}; \quad k_{22} = k_{12}; \quad k_{31} = 0.18 \cdot k_{11} \text{ et } k_{32} = k_{12};$$

La réponse en position du système en boucle fermée est représentée par les figures 8. Le suivi selon l'axe X est satisfaisant (figure 9.a1). L'erreur est maximale au début du mouvement soit $-5.9 \cdot 10^{-3}$. Une fois en contact avec l'environnement ($t = 1.75s$), cette erreur décroît jusqu'à la valeur $1.30 \cdot 10^{-7}m$. Ceci s'explique par le fait que pour assurer un bon suivi de consignes, l'effecteur se place dans une position qui lui permet de maintenir un effort normal à la surface de l'obstacle, tout en garantissant un déplacement suivant l'axe Y. Le suivi selon l'axe Y est également satisfaisant au cours de tout le mouvement de l'effecteur (figure 9.a2). L'erreur atteint une valeur maximale au moment de l'impact soit $5.5 \cdot 10^{-3}m$, puis elle se stabilise à $8.03 \cdot 10^{-6}m$.

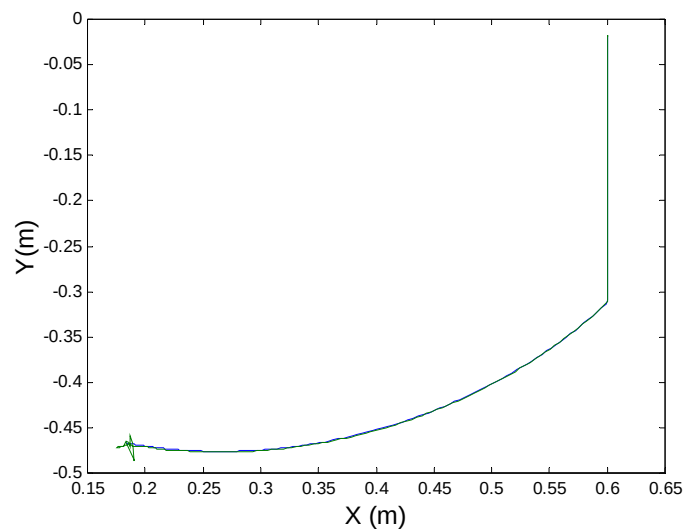
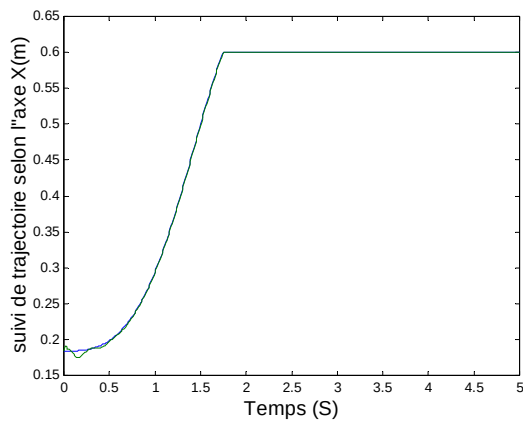
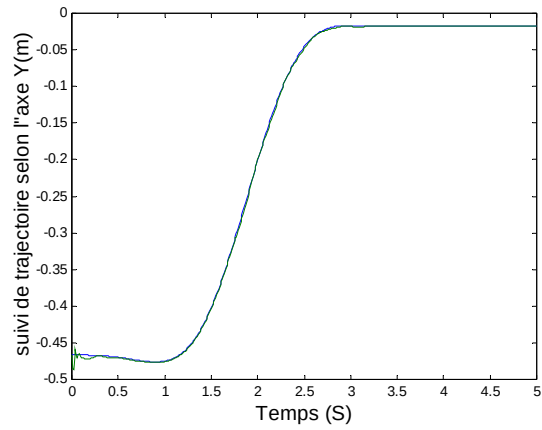
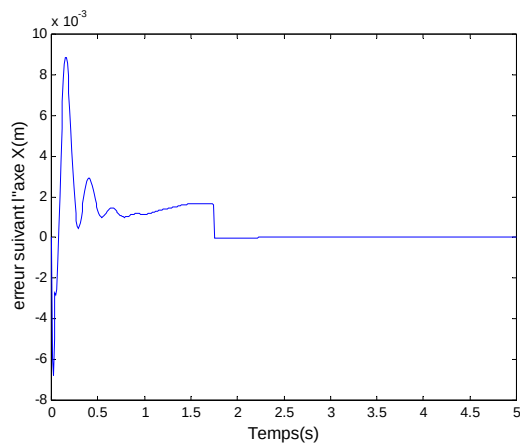
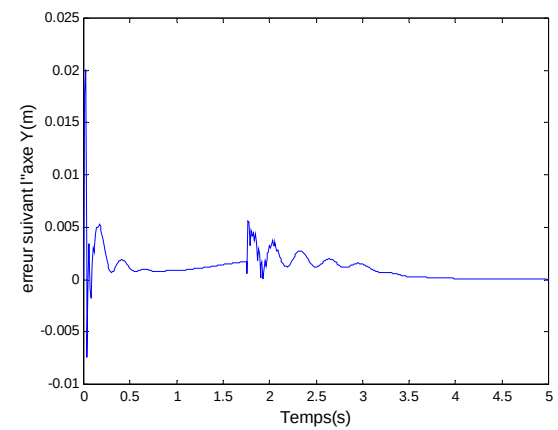
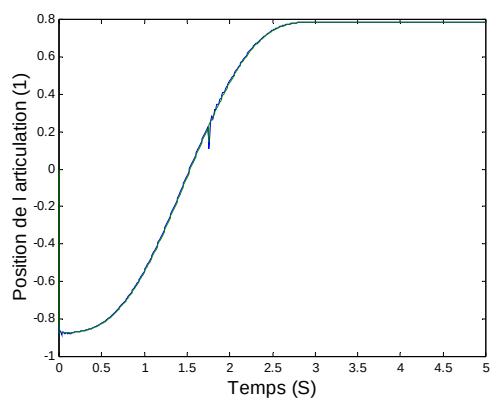
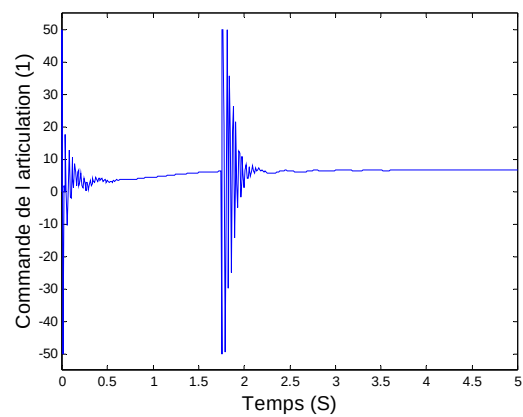


Figure 8. Trajectoire de l'effecteur dans le plan.

Les figures 9.c1, c2 et c3 illustrent les trajectoires articulaires. Ces courbes montrent un bon suivi de trajectoires au cours de tout le mouvement. La mise en contact avec l'environnement apparaît sous forme de petites déformations des trajectoires réelles à l'instant $t = 1.75s$. Après le contact, la loi de commande en effort tend à compenser cette perturbation, ce qui a permis de corriger le suivi.

Les figures 9.d1, d2 et d3 présentent les couples appliqués aux articulations du bras manipulateurs. Afin de vaincre l'inertie des bras, les couples sont très élevés soit 49.5 N.m, 48 N.m et 47 N.m respectivement pour la première, deuxième et la troisième articulation. Lors du contact avec l'environnement, des pics apparaissent au niveau des couples de commande suivis de quelques oscillations. La commande de la première articulation fournit une valeur maximale à cet instant (49 N.m) pour compenser l'effet du choc. Les valeurs maximales des oscillations pour les deux autres articulations sont 20 N.m et 14.8 N.m respectivement pour la deuxième et troisième articulation.

**- a1 -****- a2 -****- b1 -****- b2 -****- c1 -****- d1 -**

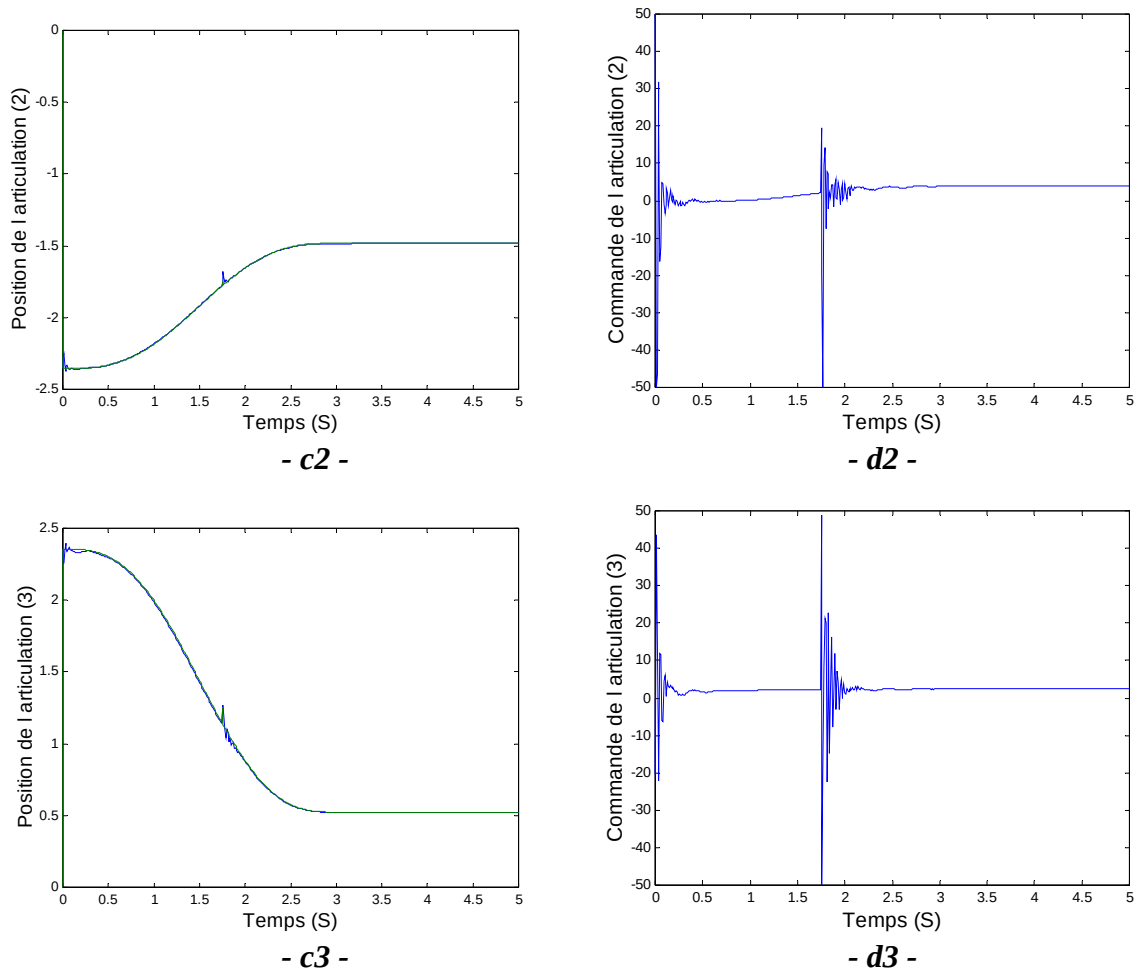


Figure 9. Suivi d'un contour rectiligne en maintenant un effort constant

- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- c1, c2 et c3 Positions respectivement des articulations 1, 2 et 3
- d1, d2 et d3 Couple appliqué respectivement aux articulations 1, 2 et 3

La figure 10 illustre les réponses en effort obtenues durant l'exécution de cette tâche.

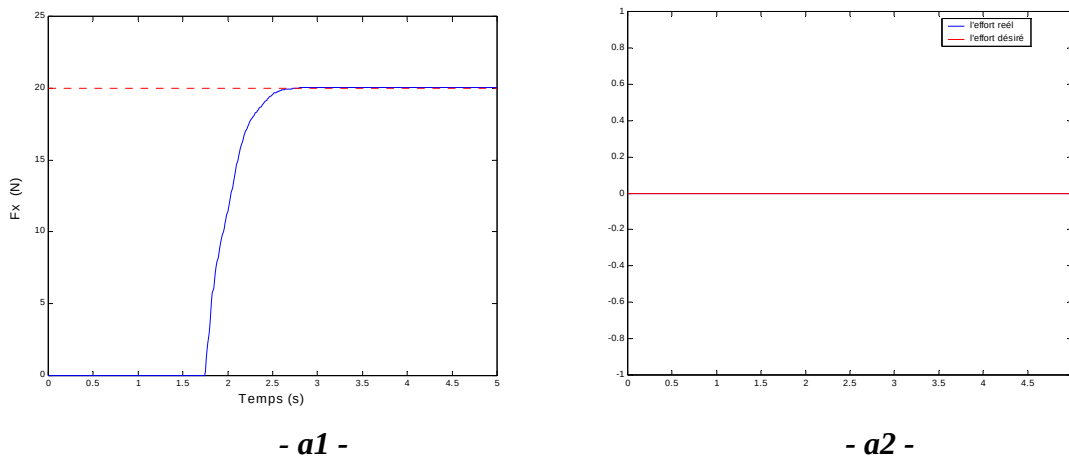


Figure 10. La réponse en effort.
a1, a2 Effort respectivement selon l'axe X et l'axe Y

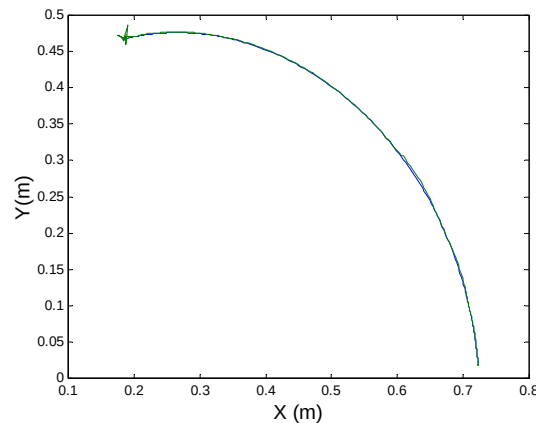
On constate que les forces avant le contact sont nulles suivant les deux directions. En effet, le robot se déplace dans cet intervalle de temps, dans un espace libre. Après l'occurrence du contact, l'effort suivant l'axe Y (figure 10.a2), reste nul car le déplacement s'effectue suivant cette direction. Tandis que l'effort suivant l'axe X (figure 10.a1), tend vers l'effort désiré soit 50N dès la mise en contact. La consigne est atteinte au bout de 0.75s.

b. Suivi d'un contour curviligne en maintenant un effort constant.

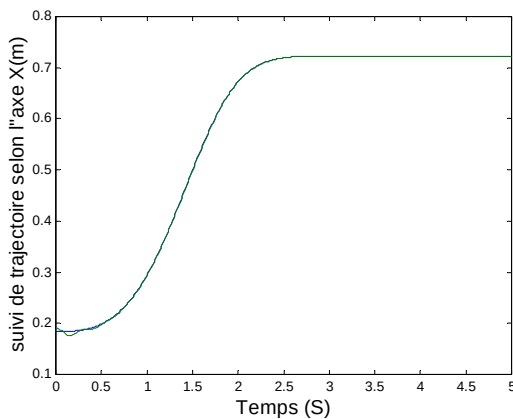
Dans cette partie, nous illustrons le suivi de trajectoire dans le cas du suivi d'un mur curviligne. La réponse en position est donnée par la figure 11 nous pouvons remarquer une bonne poursuite de trajectoire. La présence d'un environnement curviligne se traduit par un déplacement suivant les deux axes (X, Y).

Le suivi selon l'axe X est satisfaisant (figures 11.a1 et b1). Dans le repère tâche, l'erreur initiale est $-6 \cdot 10^{-3}$ m, elle se stabilise à $1.48 \cdot 10^{-3}$ m. Cette erreur est due à l'erreur de position initiale de l'effecteur. Une fois en contact avec l'environnement, cette erreur décroît jusqu'à atteindre la valeur $1.27 \cdot 10^{-7}$ m.

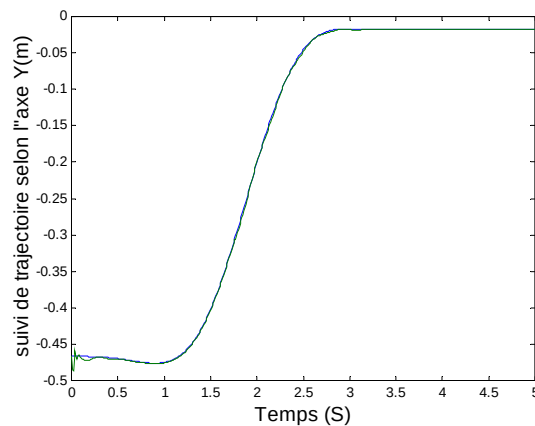
Le suivi selon l'axe Y est aussi satisfaisant durant le mouvement de l'effecteur (figures 11.a2 et b2). Ainsi l'erreur suivant cet axe se stabilise à 10^{-3} m après un dépassement dû à la tendance de l'effecteur à rejoindre la trajectoire désirée. Cependant, nous notons une erreur de $6 \cdot 10^{-3}$ m au moment de l'impact qui se stabilise enfin à la valeur $3.44 \cdot 10^{-5}$ m.



- a -



- a1 -



- a2 -

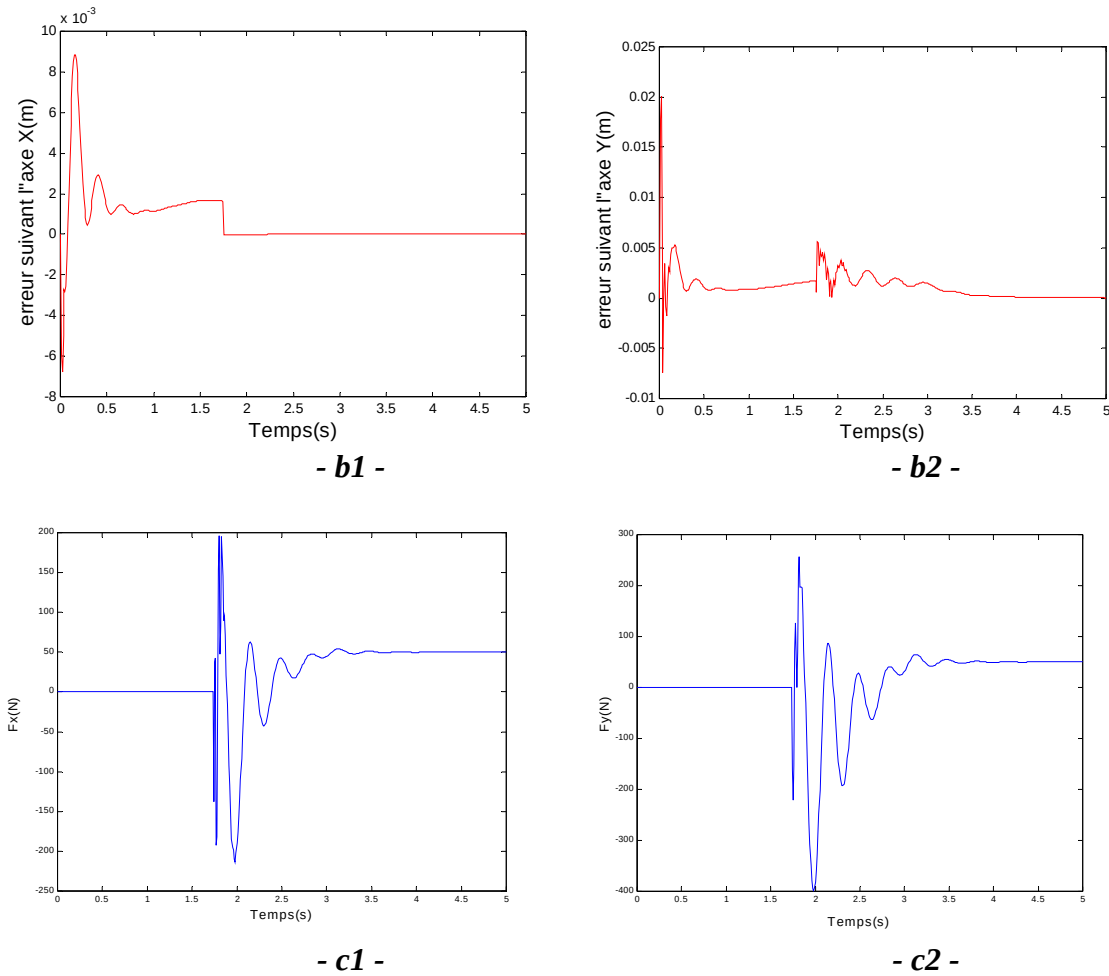


Figure 11. Suivi d'un contour curviligne en maintenant un effort.

- a. Trajectoire de l'effecteur dans le plan (X, Y)
- a1 et a2 Trajectoires suivant l'axe X et l'axe Y
- b1 et b2 Erreur de suivi selon les axes X et Y
- c1 et c2 Effort selon l'axe X et L'axe Y

Compte tenu de la forme curviligne de l'environnement, l'effort exercé sur ce dernier est à chaque instant selon les deux directions X et Y. En effet, il est à chaque instant normal à la tangente de la surface du contour. Ainsi, on obtient une variation d'effort suivant les deux axes. Les courbes 11.c1 et 11.c2 montrent que les efforts avant l'instant de l'impact ($t = 1.75s$) sont nuls. Une fois que le contact est établi, des pics apparaissent suivis de quelques oscillations traduisant ainsi l'effet de l'interaction robot-environnement. En régime permanent, les efforts mesurés se stabilisent à la valeur désirée soit 50N suivant l'axe X et 50N suivant l'axe Y.

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

$$\begin{aligned} G_{u1} &= 180; & G_{e1} &= 6\pi/2; & G_{de1} &= 6.2\pi/2; \\ G_{u2} &= 200; & G_{e2} &= 0.8\pi; & G_{de2} &= 1.4\pi; \\ G_{u3} &= 70; & G_{e3} &= 1\pi; & G_{de3} &= 3.2\pi; \end{aligned}$$

$$k_{11} = 0.195; \quad k_{12} = 0.81; \quad k_{21} = 0.195 \cdot k_{11}; \quad k_{22} = k_{12}; \quad k_{31} = 0.195 \cdot k_{11} \quad \text{et} \quad k_{32} = k_{12};$$

c. Saisie et transport d'un objet d'un point « A » à un point « B »

Les gains permettant d'obtenir un bon suivi de la trajectoire désirée sont:

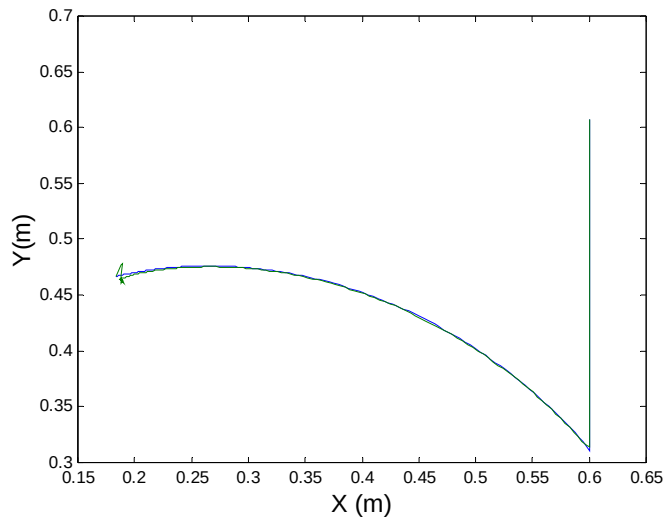
$$Gu_1=180; \quad Ge_1=6\pi/2; \quad Gde_1=6.2\pi/2;$$

$$Gu_2=200; \quad Ge_2=0.8\pi; \quad Gde_2=1.4\pi;$$

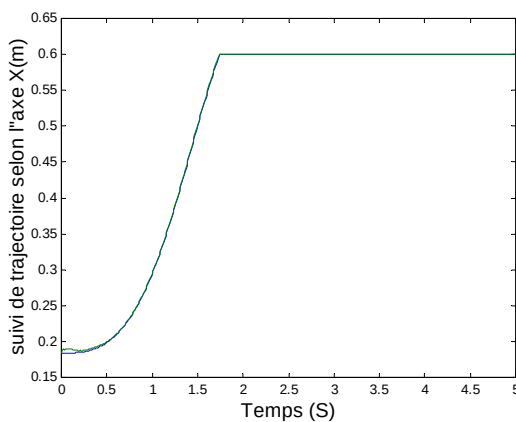
$$Gu_3=55; \quad Ge_3=1\pi; \quad Gde_3=5\pi;$$

$$k_{11}=0.18; \quad k_{12}=0.81; \quad k_{21}=0.18k_{11}; \quad k_{22}=k_{12}; \quad k_{31}=0.18k_{11} \text{ et } k_{32}=k_{12};$$

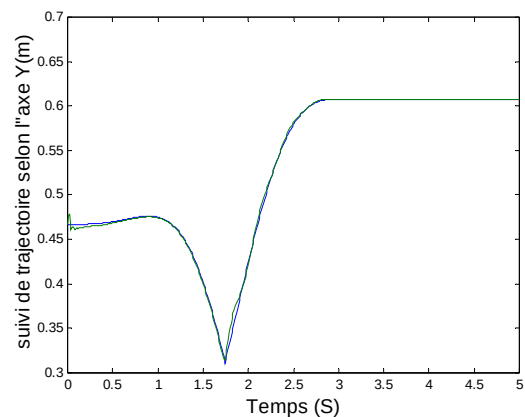
Les résultats obtenus sont consignés sur la figure 12. La trajectoire réelle et la trajectoire désirée montrent que le suivi est de bonne qualité. Les trajectoires selon les axes X et Y sont représentées sur les figures 12.b1 et b2. Les erreurs initiales sont $-3.7 \cdot 10^{-3}$ m suivant X et -0.01 m suivant Y. Elles sont dues essentiellement à l'inertie des bras et à l'erreur initiale de positionnement. Au moment de la mise en contact, l'erreur suivant l'axe X croît brusquement à la valeur $2.06 \cdot 10^{-6}$ m. Suivant la direction Y des oscillations apparaissent. L'erreur cartésienne est maximale à cet instant soit $3.15 \cdot 10^{-3}$ m. ceci s'explique par l'occurrence du contact avec l'objet. Cette erreur se stabilise en régime permanent à la valeur $8.74 \cdot 10^{-7}$ m.



- a -



- b1 -



- b2 -

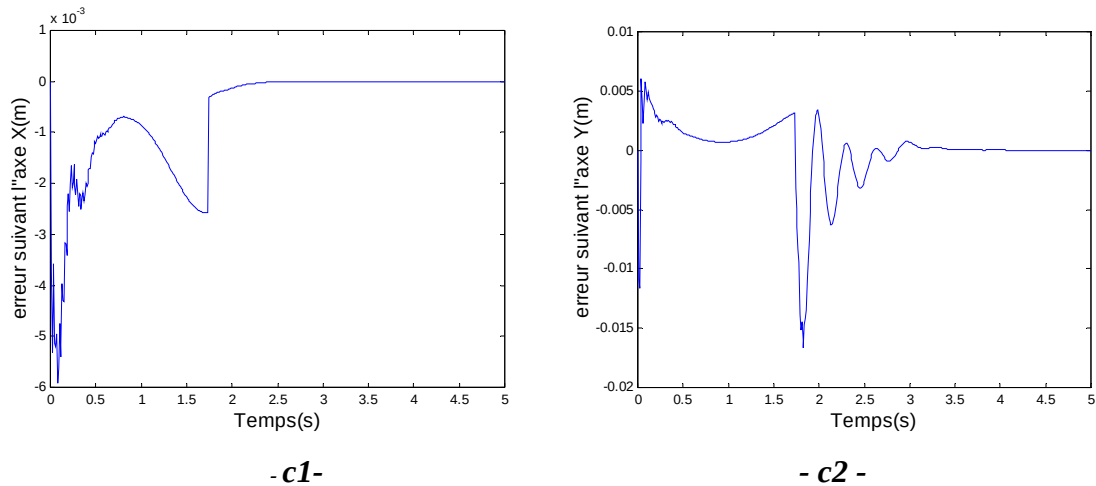


Figure 12. Transport d'un objet par un bras manipulateur redondant.

- a. Trajectoire de l'organe terminal dans le plan (X,Y)
- b1, b2 Trajectoire suivant l'axe X et l'axe Y
- c1, c2 Erreur de suivi suivant l'axe X et l'axe Y

Les courbes des efforts sont données dans la figure 13.

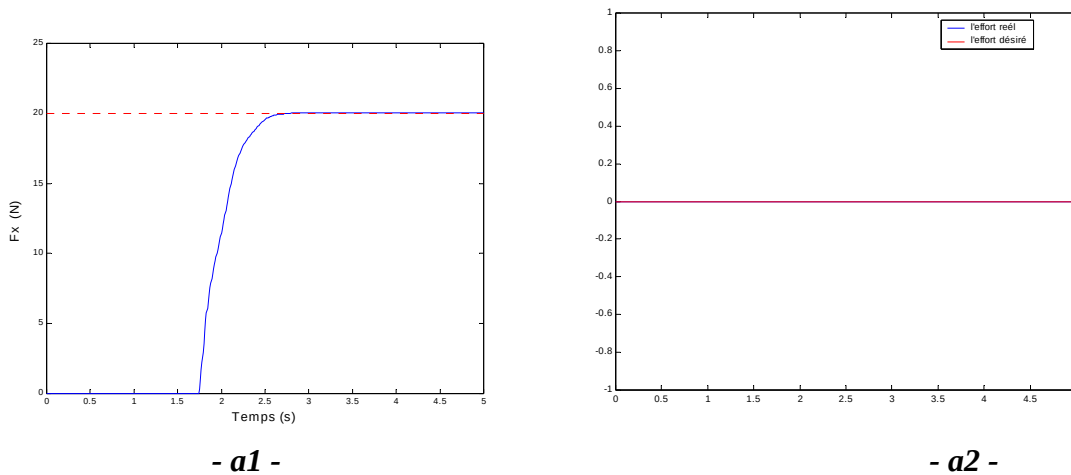


Figure 13. La réponse en effort.

a1 et a2 Effort respectivement selon l'axe X et l'axe Y

Le mouvement du robot dans l'espace libre annule tout effort de contact. Entre l'instant 0s et 1.75s, les forces d'interactions sont nulles suivant les deux axes. A la mise en contact avec l'environnement, le robot maintient un effort normal de 20N à la surface de l'objet.

En effet, à l'instant $t = 1.75s$, les efforts suivant l'axe « X » tendent vers l'effort désiré (20N), elle l'atteint au bout de 0.75s. . Tandis que les efforts selon l'axe Y restent nuls, compte tenu du déplacement désiré qui s'effectue suivant cette direction, le contact a donc eu lieu suivant la perpendiculaire à la surface de l'environnement.

V.III.8 Conclusion

Dans cette dernière partie, nous avons présenté la théorie des réseaux de neurones, l'algorithme d'apprentissage utilisé et la mise en œuvre d'un système d'inférence floue sous forme d'un réseau neuronal. Une architecture de commande hybride force- position utilisant les concepts neuro-flous est proposée. L'avantage d'une telle architecture permet de combiner les capacités d'apprentissage des réseaux de neurones et les avantages de la logique floue. En effet, les connaissances fournies par l'expert permettent d'initialiser les paramètres du réseau neuronal non plus d'une manière aléatoire mais à des valeurs proches de l'optimum. Ainsi, la convergence de l'apprentissage est nettement plus rapide.

Dans le but d'évaluer les performances de la commande proposée, nous avons considéré les mêmes configurations de tâches que l'approche de commande dynamique et l'approche de commande floue. Les régulateurs neuro-flous utilisés lors des simulations sont à trois classes. Les résultats obtenus sont satisfaisants. En effet, les suivis de trajectoires que ce soit en espace libre ou en espace contraint sont de bonne qualité. Ces résultats montrent l'efficacité de l'approche de commande proposée.

Afin de compléter notre étude, nous proposons dans le paragraphe suivant de comparer les résultats obtenus pour les différentes approches proposées.

Quatrième partie

Etude comparative des structures de commandes proposées

Trois approches de commande en effort externe sont proposées dans ce chapitre. Elles diffèrent par l'approche utilisée pour la détermination des lois de commandes. Les figures 1, 2 et 3 illustrent ces structures de commandes.

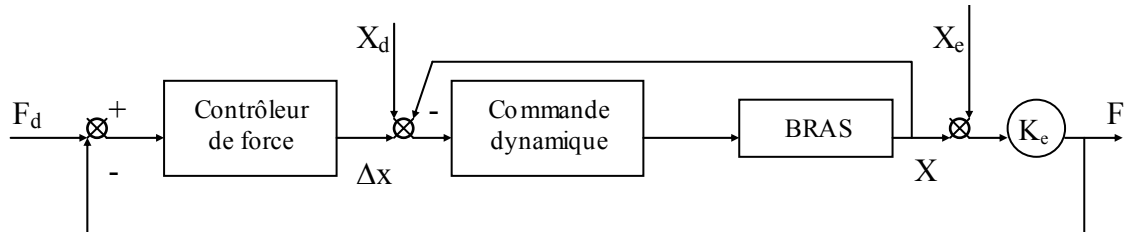


Figure 1. Structure de commande en effort externe basée sur la commande dynamique

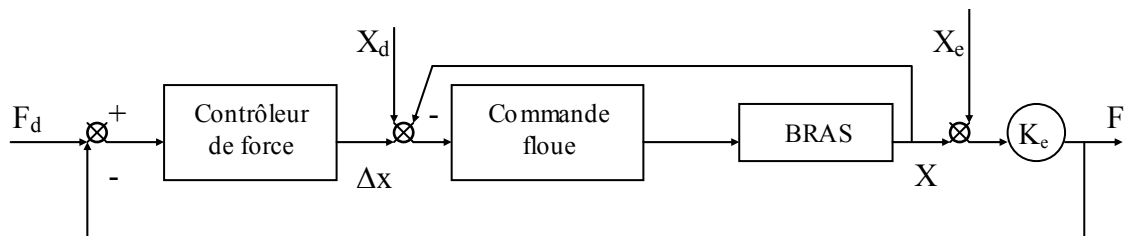


Figure 2. Structure de commande en effort externe basée sur la logique floue

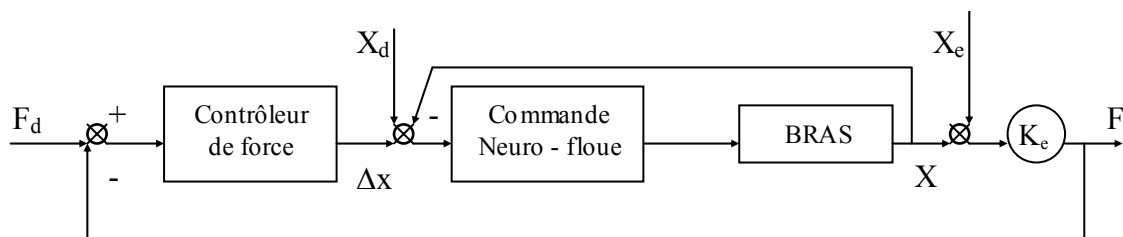


Figure 3. Structure de commande en effort externe basée sur la logique neuro-floue

L'évaluation de ces trois structures a été réalisée dans le cas de la commande d'un même robot manipulateur redondant pour différents types tâches. Ces tâches ont été choisies pour leur représentabilité dans le domaine de la robotique. Les résultats obtenus au cours des différentes simulations sont satisfaisants.

Le but de ce chapitre est de comparer les performances des trois structures de commandes proposées. Compte tenu que généralement les tâches robotisées sont spécifiées dans l'espace opérationnel, nous limiterons notre étude aux tâches spécifiées dans ce repère.

Lors des simulations trois types de configurations sont considérés:

- Mouvement dans un espace libre sans charge
- Suivi d'un mur rectiligne
- Suivi d'un mur curviligne
- Saisie et transport d'un objet.

Les courbes représentant les erreurs de suivis obtenus selon les axes X et Y pour les différentes tâches considérées sont représentées sur les figure (4, 5, 6 et 7). Ces courbes font apparaître un bon suivi de trajectoires pour les trois commandes. Toutefois, la commande basée sur les concepts des réseaux neuro-flou présentes de meilleurs résultats que les autres approches. En effet, l'erreur en régime permanent est plus faible.

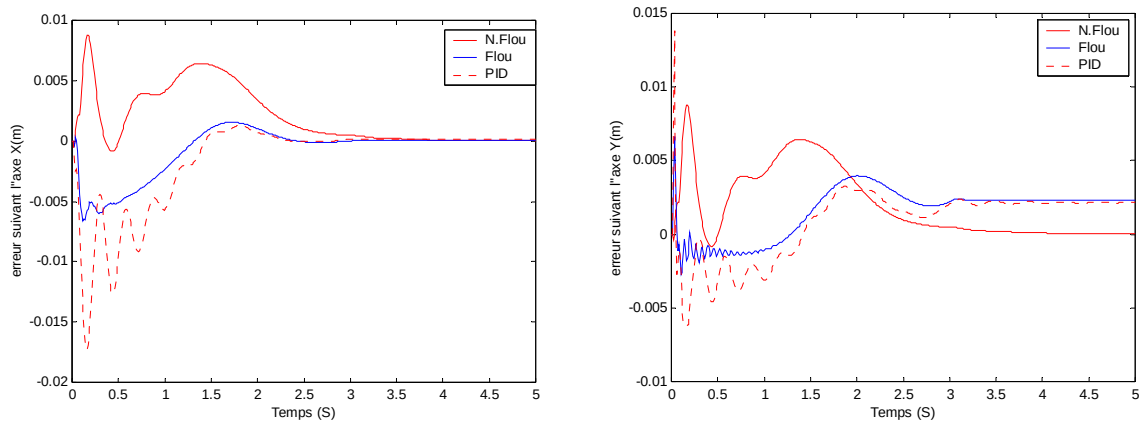


Figure 4. Mouvement dans un espace libre.

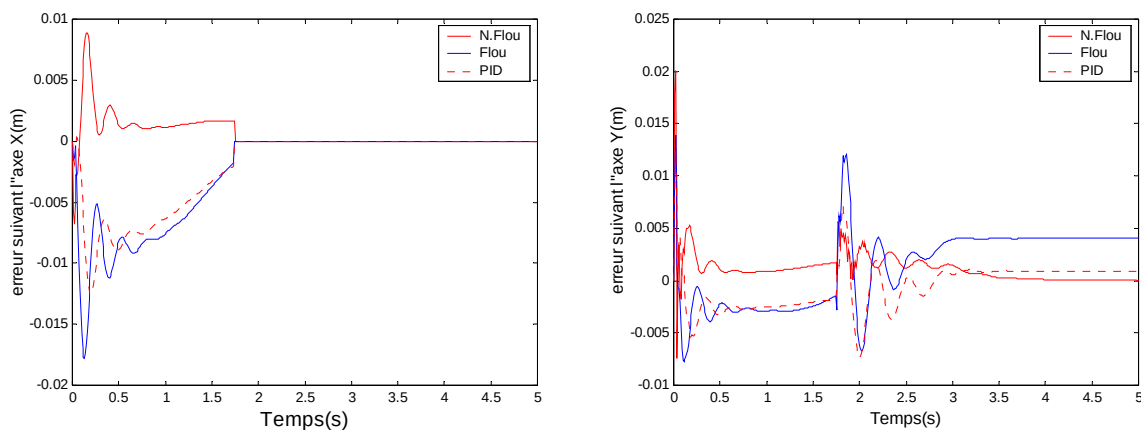


Figure 5. Erreur de suivi d'un mur rectiligne

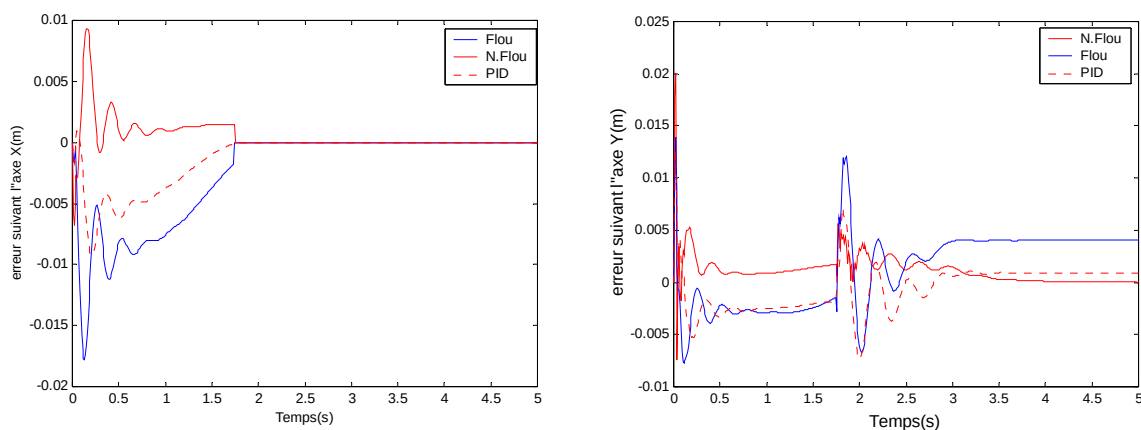


Figure 6. Erreur de suivi d'un mur curviligne.

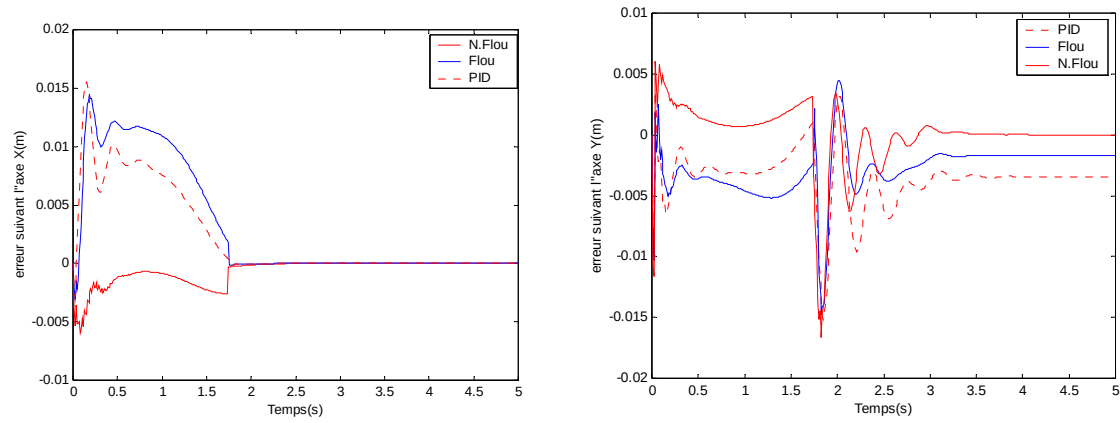


Figure 7. Erreur de suivi lors d'un transport d'un objet.

Le tableau 1 donne un récapitulatif des résultats obtenus aux points critiques.

Structure de commande	Mouvement non contraint sans charge	Suivi d'un mur rectiligne	Suivi d'un mur Curviligne	Saisie et transport d'un objet
Commande dynamique	$E_i = (-1.5, -9.2) \times 10^{-3} \text{ m}$ $E_f = (0.14, 2.3) \times 10^{-3} \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$	$E_i = (0.4, 7) \times 10^{-3} \text{ m}$ $E_f = (-1.8 \times 10^{-6}, 8 \times 10^{-4}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$	$E_i = (0.04, 0.07) \times 10^{-2} \text{ m}$ $E_f = (-1.8 \times 10^{-6}, 8 \times 10^{-4}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$	$E_i = (-2.95, -7.4) \times 10^{-3} \text{ m}$ $E_f = (-2.4 \times 10^{-7}, -3.5 \times 10^{-3}) \text{ m}$ $C_i = (15, 14.5, 14.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$
Commande floue	$E_i = (-0.4, 4.1) \times 10^{-3} \text{ m}$ $E_f = (0.03, 2.33) \times 10^{-3} \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$	$E_i = (-1.5, 9.2) \times 10^{-3} \text{ m}$ $E_f = (4.87 \times 10^{-8}, 4 \times 10^{-3}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$	$E_i = (-1.5, 0.14) \times 10^{-3} \text{ m}$ $E_f = (10^{-8}, 9 \times 10^{-4}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$	$E_i = (-2.1, -8.1) \times 10^{-3} \text{ m}$ $E_f = (8 \times 10^{-7}, 1.7 \times 10^{-3}) \text{ m}$ $C_i = (15, 14.5, 14.5) \text{ N.m}$ $C_f = (16.2, 15.4, 14.8) \text{ N.m}$
Commande neuro-floue	$E_i = (-0.6, -4.1) \times 10^{-3} \text{ m}$ $E_f = (1.45, 3.28) \times 10^{-5} \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$	$E_i = (-0.59, 1.7) \times 10^{-2} \text{ m}$ $E_f = (1.3 \times 10^{-7}, 8.03 \times 10^{-6}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$	$E_i = (-6, 2) \times 10^{-3} \text{ m}$ $E_f = (1.27 \times 10^{-7}, 3.44 \times 10^{-5}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$	$E_i = (-0.37, -1) \times 10^{-2} \text{ m}$ $E_f = (2.06 \times 10^{-6}, 8.74 \times 10^{-7}) \text{ m}$ $C_i = (17, 16.4, 15.5) \text{ N.m}$ $C_f = (6.5, 3.94, 2.48) \text{ N.m}$

Tableau V.I: Résultats obtenus dans le cas des tâches spécifiées dans l'espace cartésien

Où

- E_i et E_f désignent respectivement les erreurs initiales et finales selon e deux direction du plan (X,Y)
- C_i et C_f désignent respectivement les couples initiaux et finaux appliqués sur la première deuxième et troisième articulation

Conclusion

Ce chapitre porte sur la commande hybride force / position basée sur la structure de commande en effort externe en utilisant différentes approches. En effet, trois approches de commande sont proposées: l'approche utilisant la commande dynamique, l'approche utilisant les concepts de la logique floue et enfin approche utilisant les concepts neuro-flou.

Les trois structures de commandes proposées sont implémentées en simulation sur un système non-linéaire couplé (bras manipulateur redondant à trois liaisons rotoïdes) pour la réalisation de différentes tâches robotisées.

Les résultats obtenus ont permis de valider ces trois approches. Ils sont sensiblement identiques compte tenu du bon réglage des différents gains. Pour la suite de notre travail nous choisissons la structure de commande en effort externe basée sur les concepts neuro-flou. La tâche considérée nécessite la coopération de deux robots redondants. Ceci fera l'objet du chapitre suivant.

COOPERATION DE DEUX ROBOTS MANIPULATEURS REDONDANTS

Introduction

Afin d'augmenter le nombre d'applications industrielles, il devient de plus en plus nécessaire de s'intéresser à la notion de coopération de plusieurs systèmes mécaniques articulés. En effet, certaines tâches ne peuvent être effectuées par un seul bras. Certaines opérations nécessitent la manipulation d'objets lourds, encombrants, flexibles ou ne possédant pas de point de saisie, ou plus simplement l'action coordonnée de plusieurs bras permet de diminuer le temps d'exécution d'une tâche, donc d'augmenter la productivité. Il peut s'agir de tâches répétitives et bien définies en milieu industriel sur une chaîne de fabrication, comme de tâches moins visibles en environnement non structuré comme l'espace ou les fonds sous-marins, telles que des tâches de maintenances, d'inspection. Dans ces environnements, le système ne dispose en principe pas de matériel d'assistance péri-robotique et le second bras peut jouer ce rôle.

Au vu de ces tâches particulièrement intéressantes, on comprend que le problème de coordination et de commande de robots travaillant en coopération ait attiré l'attention des chercheurs depuis quelques années. En effet, on peut répartir ces problèmes en deux grands champs d'investigation. Le premier concerne la génération de la trajectoire coordonnée pour deux ou plusieurs bras et l'exécution synchronisée de ces trajectoires. Un environnement de programmation est nécessaire pour exécuter une telle tâche et assurer la cohérence des trajectoires. Le deuxième champ de recherche, qui fait l'objet de notre étude, traite la commande des robots. L'étude de la commande de plusieurs bras articulés est un problème très difficile, examiné seulement depuis quelques années.

Dans notre présente étude, nous nous limiterons au cas de deux robots manipulateurs redondants à trois degrés de liberté, et nous n'examinerons pas le problème d'évitement de collisions entre robots (figure 1).

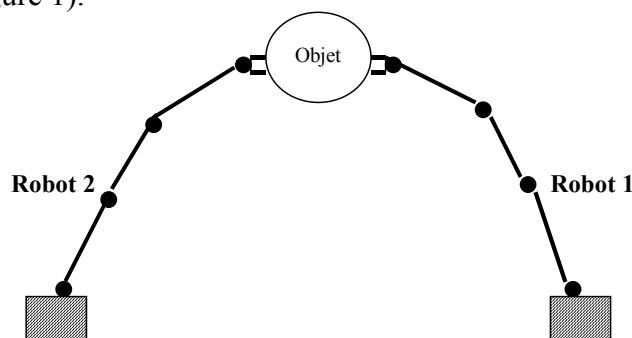


Figure 1. Système à deux bras manipulateurs.

Dans le cas des robots multi-bras impliquant la présence de contraintes internes à travers les objets manipulés, il est impératif de choisir une approche qui a pour principe général d'introduire une compliance active dans le système de commande. De nombreuses applications nécessitent de commander la position de l'effecteur et les efforts d'interactions avec l'environnement. En effet, généralement une commande en position seule, n'est pas suffisante pour les tâches complexes. Il est donc nécessaire de commander aussi les efforts de contact sur

les objets transportés. Dans ce cas, le contrôle d'effort dépend explicitement de la tâche, au cours du mouvement les objets flexibles doivent être maintenus en extension ou en pression constante. Les objets rigides peuvent être tenus uniquement par les effecteurs des robots, en exerçant des efforts sur sa surface, mais ces efforts ne doivent pas être trop élevés pour ne pas l'endommager [Deg 93].

L'objectif de ce chapitre, est de proposer une structure de commande permettant la coordination des mouvements de deux bras manipulateurs redondants.

Description de la tâche

Dans notre étude, La tâche à réaliser consiste à déplacer un objet (dans notre cas une barre métallique rigide), d'une position initiale « A » vers une position finale « B » (figure 2). L'analyse de l'exécution de cette tâche fait apparaître trois phases:

- Une phase d'approche où les deux bras manipulateurs se déplacent dans un espace libre
- Une phase de mise en contact où est effectuée la saisie de l'objet.
- Une phase de transport de l'objet. Durant cette dernière étape, les bras manipulateurs se déplacent simultanément dans un espace libre tout en assurant le maintien de l'objet.

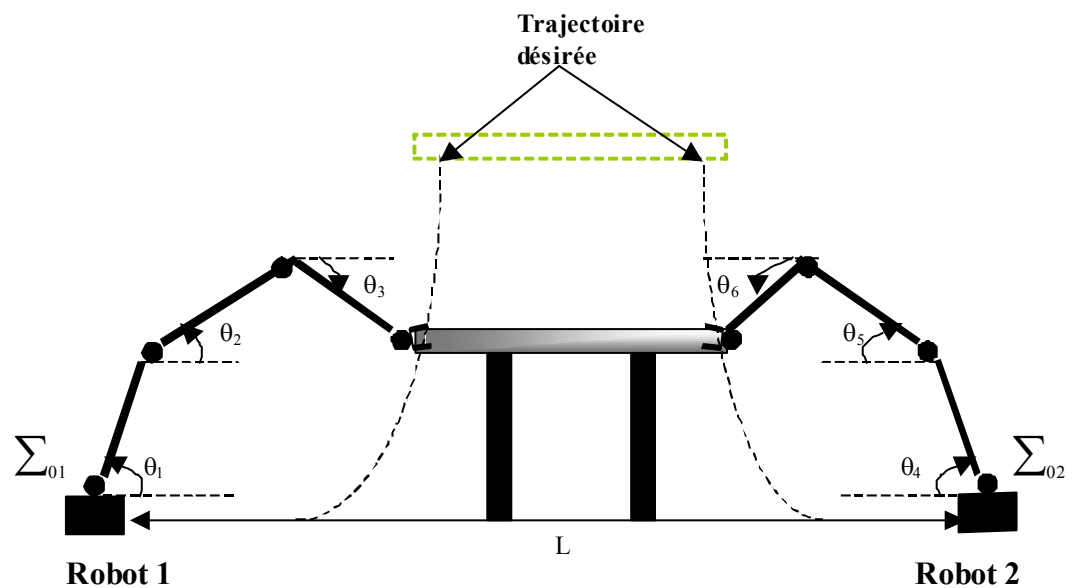


Figure 2. Tâche à réaliser

Il est important de noter que si les mouvements des robots ne sont pas simultanés il y a risque de perte (ou de déformation ou de détérioration) de l'objet ou encore de détérioration de l'un des effecteurs. Afin de mener à bien cette tâche, les variables à asservir sont la position des effecteurs des deux bras manipulateurs, ainsi que les efforts appliqués sur l'objet.

Architecture de commandes coopérante

Lors du contact de l'objet avec les deux robots manipulateurs une chaîne cinématique fermée est formée par cet ensemble. Il est nécessaire donc de prendre en compte les contraintes que cette situation implique. Ceci dit, il est indispensable de coordonner les mouvements des

deux bras manipulateurs. En effet, dans les cas extrêmes, un dysfonctionnement pourrait engendrer la détérioration des manipulateurs ou encore de l'objet manipulé [Der 03].

Pour la commande de chacun des robots coopérants, il est nécessaire de choisir une approche qui permet d'introduire une compliance active dans le système de commande. Compte tenu, de la tâche spécifiée, seules les méthodes de commande hybride Force/Position peuvent répondre à ce problème. De nombreux chercheurs se sont intéressés à la commande multi-robots et ont proposé des solutions en respectant les contraintes précédemment citées [Zhe 86] [Uch 88]. Parmi les travaux les plus utilisés dans la littérature On peut citer :

- **La commande non symétrique (Meneur/suiveur):** Cette méthode, proposée par Y.F.Zheng et J.Y.S.Luh est basée sur une utilisation non symétrique des deux bras. En effet, un des deux bras, le «Meneur», voit sa trajectoire définie à partir de celle de l'objet. La trajectoire du second robot, le «Suiveur», est directement dérivée de celle du «Meneur» à partir de la connaissance des contraintes de la chaîne cinématique fermée. Son inconvénient majeur est qu'elle suppose que la commande des positions est parfaite. En effet, les erreurs de modélisation géométrique et les erreurs d'asservissement introduites dans la chaîne cinématique fermée, composée des deux bras et de l'objet, engendrent des contraintes importantes.
- **La commande Maître/Esclave:** Afin de résoudre le problème de la commande Meneur/suiveur, certains chercheurs ont proposé d'utiliser une commande de type Maître/Esclave. Cette commande consiste à contrôler séparément et indépendamment les deux robots. Son principe est qu'elle déduit la trajectoire du bras maître à partir de celle de l'objet et qu'elle impose au second bras (esclave), d'exercer certains efforts sur cet objet. Cette méthode présente les inconvénients suivants: le moindre mouvement du bras maître sera retransmis sous la forme d'une contrainte au niveau du capteur d'effort du robot esclave. Ainsi, elle nécessite une réponse rapide de l'esclave, sachant que la trajectoire de ce dernier est déduite de celle du Maître.
- **La commande symétrique (Hybride Force/Position):** Uchiyama, Iwasawa et Hakomori utilisent une approche différente pour appliquer la méthode de Reboulet et Robert au contrôle coopératif de robots [Uch 87]. Cette méthode présente la particularité de travailler sur des données relatives à l'objet à manipuler, à la différence de bon nombre de méthodes qui s'attachent à considérer les bras manipulateurs plutôt que l'objet lui-même. La solution consiste à ne plus considérer les deux bras manipulateurs comme étant indépendants mais comme faisant partie d'un seul et même robot à deux bras, possédant un unique système de commande. Comme tout se passe au niveau de l'objet, les bras jouent des rôles semblables, il est donc possible de répartir symétriquement les déplacements et les efforts sur chaque bras. Ils définissent un nouveau système de coordonnées commun aux deux manipulateurs. Il est possible alors, d'obtenir à partir de ce nouveau système de coordonnées, de nouvelles relations statiques et cinématiques pour l'ensemble de la chaîne fermée, de la même façon que pour un seul manipulateur [Uch 88] [Hay 88]. La figure 3 présente le schéma de principe.

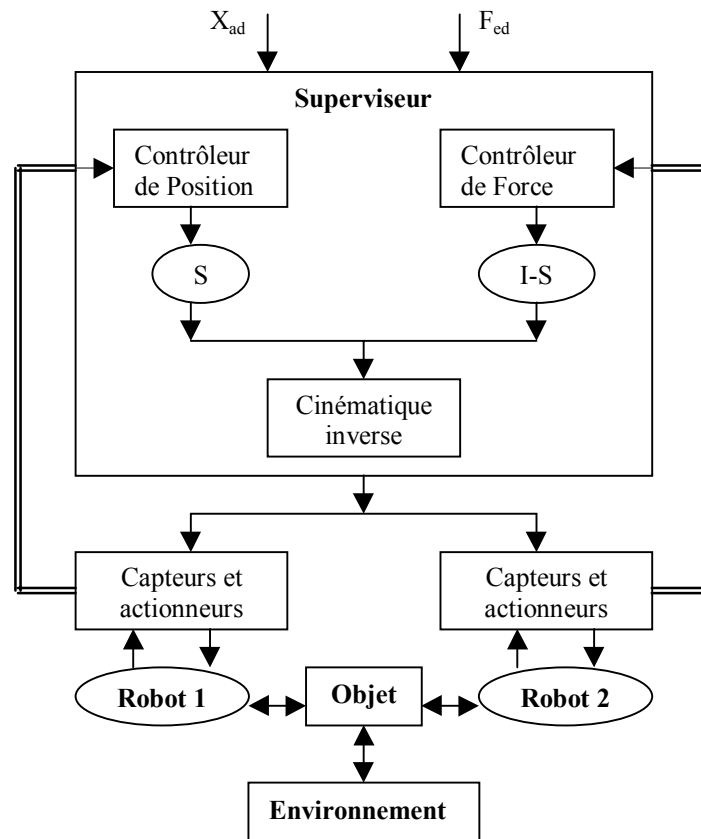


Figure 3. Schéma de principe de la solution hybride symétrique.

Où X_{ad} et F_{ed} sont respectivement les consignes de position et d'effort désirées. S est la matrice de sélection.

Compte tenu des avantages que présente la méthode proposée par Uchwiya [Uch 87] nous l'adopterons pour la suite de notre travail. Toute fois, chaque bras manipulateur sera commandé par la structure de commande externe. Ainsi, nous éviterons toute instabilité que pourrait engendrer l'utilisation de la matrice de sélection.

Architecture de commande adoptée

Nous avons adopté une architecture hiérarchisée. En effet, les deux bras manipulateurs ont un rôle symétrique et contribuent simultanément au déplacement de l'objet. Ainsi, Les efforts et les déplacements sont répartis symétriquement sur chaque bras. Les deux bras manipulateurs sont considérés comme faisant partie d'un seul et même robot à deux bras, possédant un unique système de commande. Une structure de commande hybride en effort externe permet de réaliser un contrôle simultané des forces et des positions sur chacun des bras. La coordination de deux bras manipulateurs, asservis chacun par la méthode du contrôle hybride externe, passe donc par une répartition judicieuse de consignes.

Le rôle du niveau superviseur n'est donc plus d'assurer l'asservissement des consignes fournies par l'utilisateur mais de calculer à partir de ces dernières les consignes à envoyer à chacun des contrôleurs (figure 4).

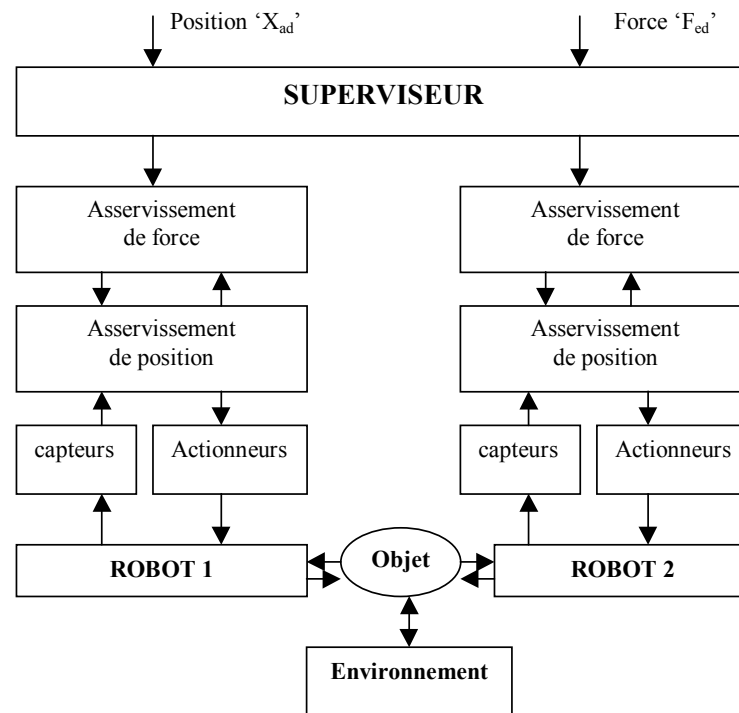


Figure 4. Architecture de commande adoptée.

Où X_{ad} et F_{ed} sont respectivement les consignes de position et d'effort désirées.

Le schéma de commande de chacun des robots est constitué d'une boucle interne de position dont l'entrée est modifiée par une commande issue de la boucle externe de force. Les deux bras manipulateurs ont un rôle symétrique et contribuent simultanément au déplacement de l'objet et aux efforts à exercer sur celui-ci. Chaque système reçoit du superviseur ses consignes de force et de position et utilise ses propres mesures de force et de position pour réaliser l'asservissement hybride. Dans ce cas le superviseur utilise les informations de position provenant de chacun des manipulateurs, après calcul de son propre modèle géométrique dans un repère commun au deux, le repère de référence.

L'application de la commande externe pour le contrôle des deux bras de façon symétrique, conduit à la structure de commande représentée dans la figure 5.

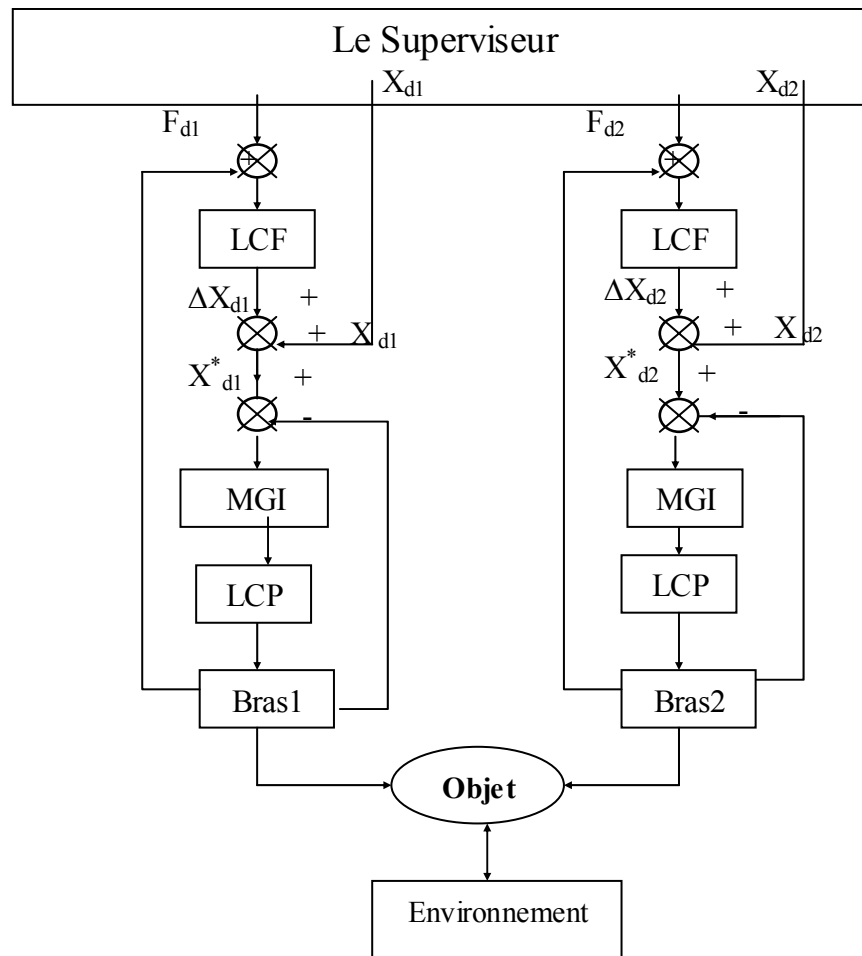


Figure 5. Schéma global de la commande externe de deux bras en coopération.

Où « LCP », « LCF » désignent respectivement la loi de commande en position et en effort. « MGI » désigne le modèle géométrique inverse (développé dans le chapitre III).

Avantages de l'architecture proposée:

Les asservissements hybrides force - position sont réalisés au niveau local de chaque système. Chacun contribue au déplacement absolu de l'objet et aux efforts à exercer sur celui ci en suivant les consignes fournies par le superviseur. Cette méthode est fondamentalement différente de la méthode Maître/Esclave et de la méthode Meneur/Suiveur. En effet, dans ces deux dernières méthodes au moins une des grandeurs, force ou position, doit être contrôlée au niveau global du superviseur.

De cette différence découle de nombreux avantages:

- La structure ne nécessite pas d'être commutée au cours de la phase de mise en contact des robots avec l'objet c'est à dire au moment de la fermeture de la chaîne cinématique.

- Le réglage de l'asservissement étant effectué indépendamment pour chaque bras et non pas globalement pour l'ensemble, il n'est donc pas nécessaire de décrire une cinématique ou une dynamique commune pour l'ensemble robots-objet.
- Les mesures de force et de position ne sont utilisées que par le manipulateur dont elles sont issues. Aucun échange d'information n'est effectué entre les contrôleurs locaux.
- En cas de rupture des communications, chaque robot reste asservi indépendamment.
- La facilité d'implémentation de son algorithme de contrôle qui demande moins de calcul que les autres méthodes et sans changement de repère.
- La mise en œuvre expérimentale de deux systèmes coopérants devient particulièrement simple.

Commande neuro_floue de deux bras manipulateurs redondants

Compte tenu des performances obtenues pour la structure de commande externe utilisant les concepts de la logique Neuro Floue, lors de la commande d'un bras manipulateur redondant, nous avons choisi de l'appliquer pour l'asservissement de chaque bras pour l'exécution d'une tâche de coopération de deux robots redondants. Le schéma d'asservissement de chaque bras est donné par la figure 6.

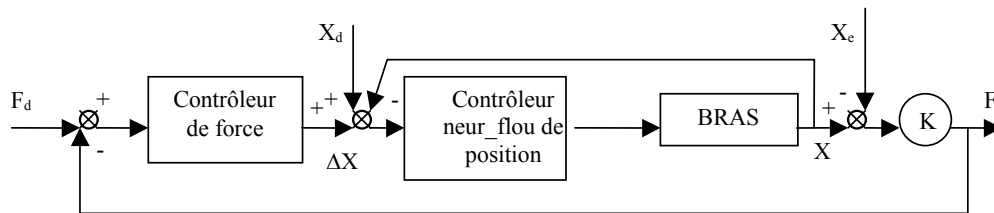


Figure 6. Structure de commande appliquée pour chaque bras.

La tâche à réaliser consiste à déplacer verticalement une barre métallique rigide, de masse 3Kg et de longueur 0.8m, d'une position initiale vers une position finale tout en la maintenant horizontale. Afin d'éviter toute rotation de la barre durant le mouvement, il est indispensable d'utiliser deux robots et de bien coordonner leurs mouvements. Ceci dit, nous avons choisi d'utiliser deux robots manipulateurs identiques. Les deux bras manipulateurs utilisés sont redondants. Ils présentent les mêmes caractéristiques que ceux utilisés dans les chapitres précédents. Il est impératif que les orientations des repères $\sum_{01}$ et $\sum_{02}$ liés respectivement au premier et au deuxième bras manipulateurs soient opposées. C'est à dire: $\theta_{i_robot1} = -\theta_{i_robot2}$. La base des deux manipulateurs étant initialement écartée d'une distance de 1m, il existe des risques de collision entre les deux robots. La résolution de ce problème rentre dans le domaine de la planification de trajectoire, qui sort du cadre de notre présente étude.

L'analyse de la tâche à réaliser permet de déduire les trajectoires désirées pour chacun des bras manipulateurs. Cette tâche est réalisée par le superviseur. Dans le cas de notre étude, compte tenu de la tâche à réaliser, les trajectoires des deux robots manipulateurs redondants, sont données dans la figure 7. La période d'échantillonnage utilisée lors des simulations est 0.01s.

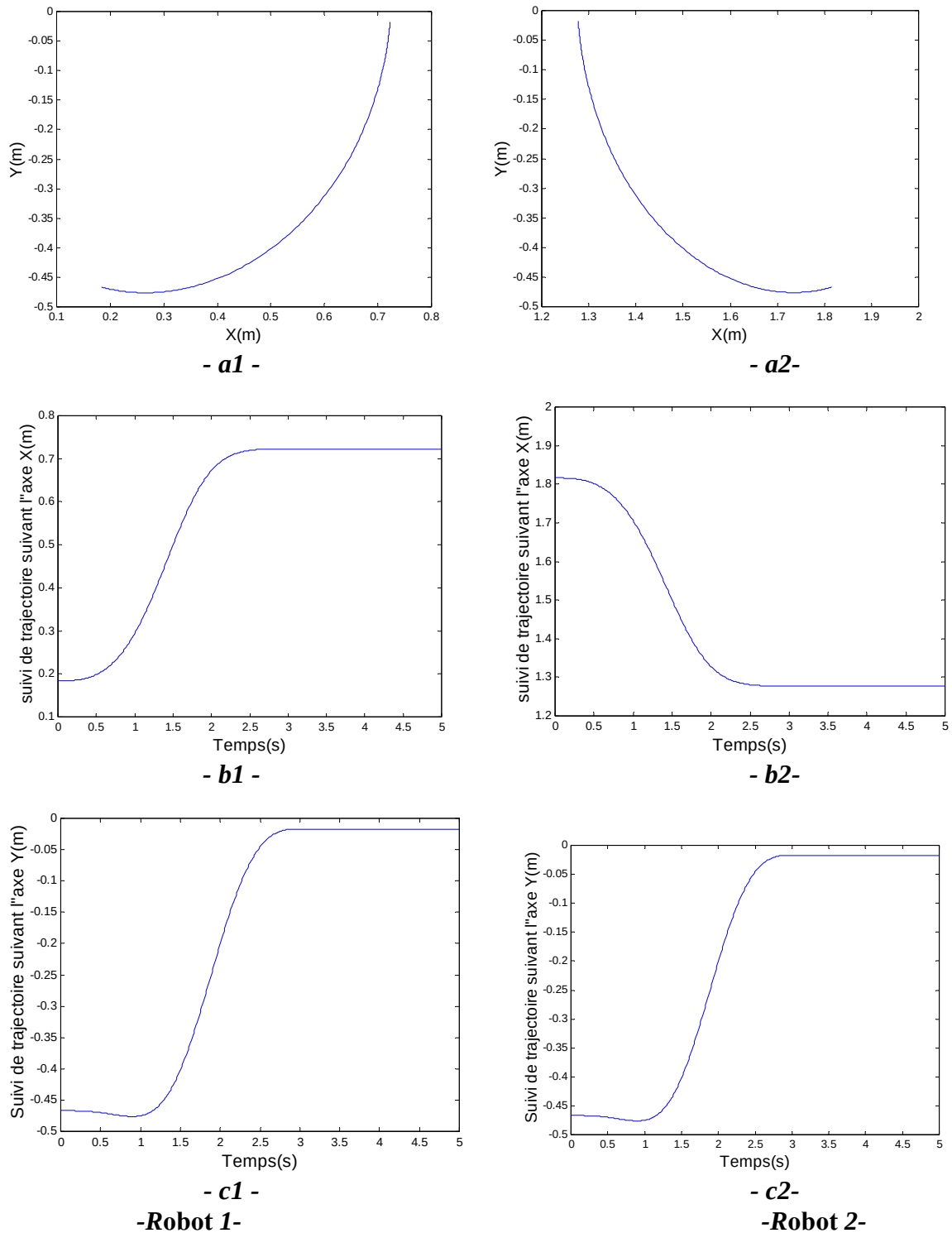


Figure VI.7 : Trajectories désirées des deux bras manipulateurs redondants dans l'espace tâche.

- a1. et a2. Trajectories désirées des effecteurs dans le plan.
- b1. et b2. Trajectory désirée des effecteurs selon l'axe « X ».
- c1. et c2. Trajectory désirée des effecteurs selon l'axe « Y ».

Afin d'évaluer les performances de la méthode proposée, nous avons simulé le mouvement de deux bras manipulateurs redondants. Les mouvements sont initialement réalisés dans le

repère tâche (X, Y) sans contraintes puis avec contraintes. En effet, à l'instant « $t = 1.75 \text{ s}$ », les deux effecteurs rentrent en contact avec l'objet qu'ils doivent déplacer.

L'objet considéré dans nos simulations est une barre rigide de masse 3 Kg et de longueur 0.8 m. les bras manipulateurs doivent la soulever en la maintenant horizontale. Dès lors que la saisie de l'objet est réalisée les efforts de serrages doivent être maintenus constants et égaux en valeur absolue à 50 N.

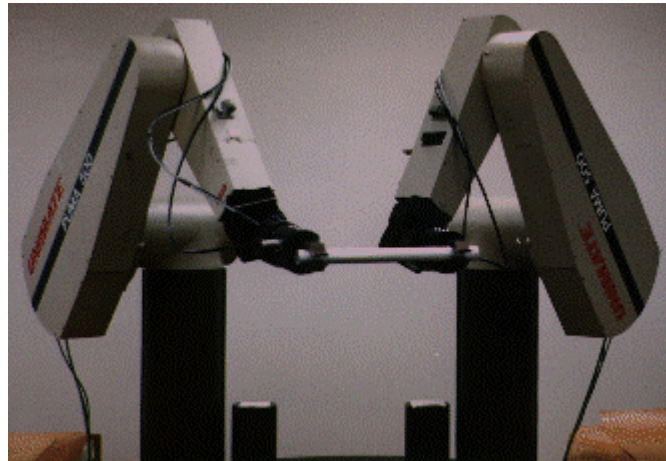


Figure 8. Présentation de la tâche.

Une bonne poursuite des trajectoires correspondant aux deux bras manipulateurs redondants considérés, a été obtenue par un réglage expérimental des gains. Ces derniers sont les mêmes pour les deux bras soit:

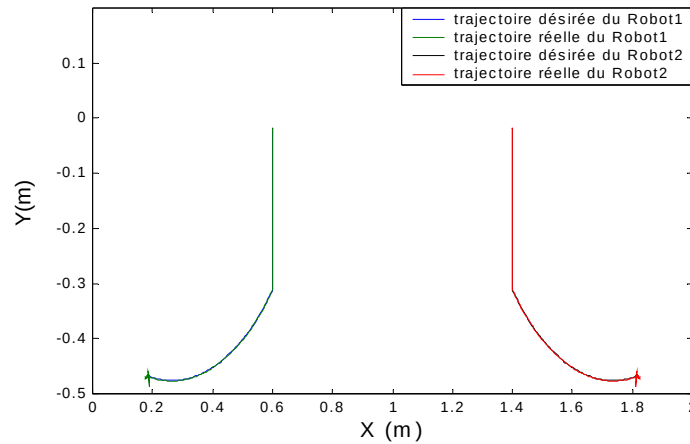
$$\begin{aligned} G_{u1} &= 180; & G_{e1} &= 6 \cdot \pi / 2; & G_{de1} &= 6.2 \cdot \pi / 2; \\ G_{xu2} &= 200; & G_{e2} &= 0.8 \cdot \pi; & G_{de2} &= 1.4 \cdot \pi; \\ G_{xu3} &= 70; & G_{e3} &= 1 \cdot \pi; & G_{de3} &= 3.9 \cdot \pi; \end{aligned}$$

$$\begin{aligned} k_{11} &= 0.18; & k_{12} &= 0.81; \\ k_{21} &= 0.18 \cdot k_{11}; & k_{22} &= k_{12}; \\ k_{31} &= 0.18 \cdot k_{11}; & k_{32} &= k_{12}; \end{aligned}$$

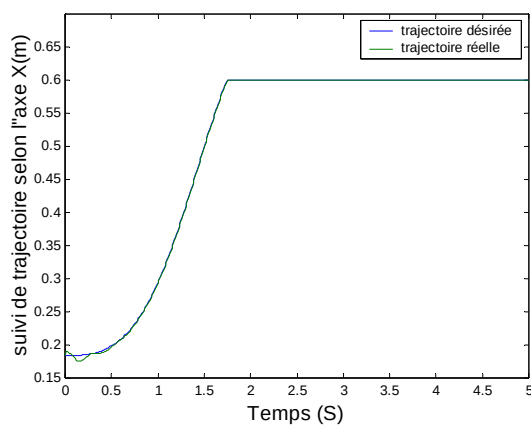
Les résultats obtenus sont consignés sur la figure 9. Cette dernière illustre les trajectoires réelles et désirées des deux bras manipulateurs. Ces courbes font apparaître deux phases de mouvement. Durant la première phase ($t \in [0, 1.74 \text{ s}]$) les effecteurs se déplacent sans aucune contrainte d'effort, tout en respectant les trajectoires désirées spécifiées par le superviseur. A l'instant $t = 1.75 \text{ s}$ les deux bras manipulateurs rentrent en contact avec l'objet, le saisissent en exerçant un effort désiré de 50 N. A partir de cet instant, l'objet est maintenu par les deux bras et est déplacé verticalement jusqu'à la position finale. Cette position correspond respectivement aux points de coordonnées (0.6, 0) et (1.4, 0) pour le premier et le deuxième bras manipulateur. Ces premières courbes montrent un bon suivi de trajectoire pour chaque bras manipulateur.

Les figures 9.d1, d2, e1 et e2 donnent les erreurs de suivis de trajectoires. Les erreurs initiales (respectivement $-5.9 \cdot 10^{-3} \text{ m}$ selon X et $1.78 \cdot 10^{-3}$ selon l'axe Y pour le premier et $-2.95 \cdot 10^{-3}$ selon X et $2.6 \cdot 10^{-3}$ selon l'axe Y pour le second bras) sont dues essentiellement à l'inertie des bras. De plus, initialement les positions des effecteurs ne coïncident pas avec les positions

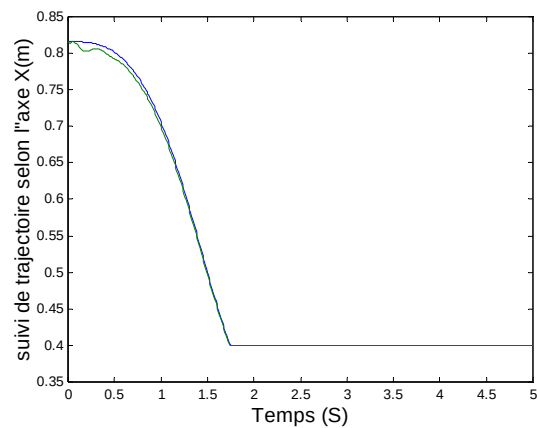
initiales désirées. A la mise en contact avec l'objet ($t = 1.75s$), les erreurs de suivi suivant l'axe X décroît brusquement aux valeurs $1.30 \cdot 10^{-7} m$ et $1.28 \cdot 10^{-7} m$. Ceci peut s'expliquer par le fait que pour compenser l'effort de contact le bras manipulateur prend une configuration permettant de maintenir la consigne d'effort désirée et ceci grâce au contrôleur d'effort. Suivant l'axe Y un pic suivi de faibles oscillations apparaissent lors du contact sur l'erreur de suivi. Ces oscillations s'amortissent au cours du mouvement et se stabilisent enfin respectivement aux valeurs $8.03 \cdot 10^{-6} m$ et $8.93 \cdot 10^{-6} m$ pour le premier et pour le second bras.



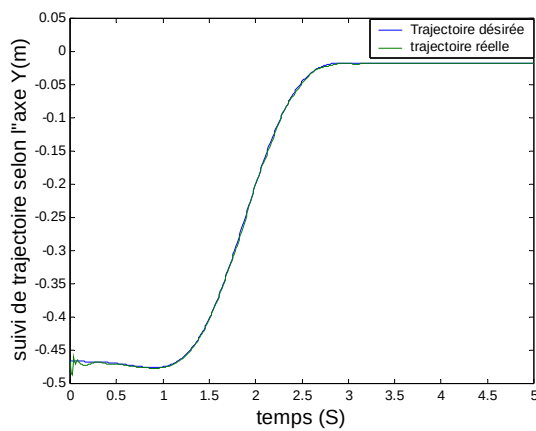
- a -



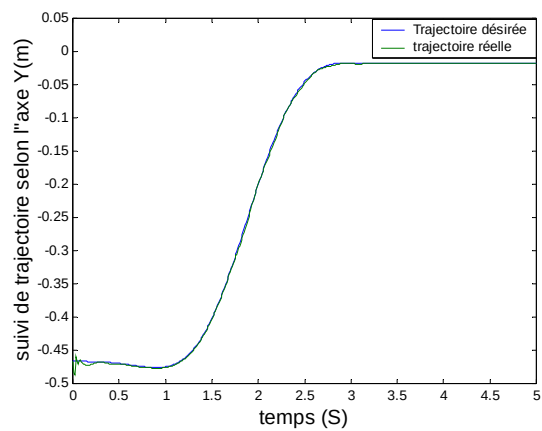
- b1 -



- b2 -



- c1 -



- c2 -

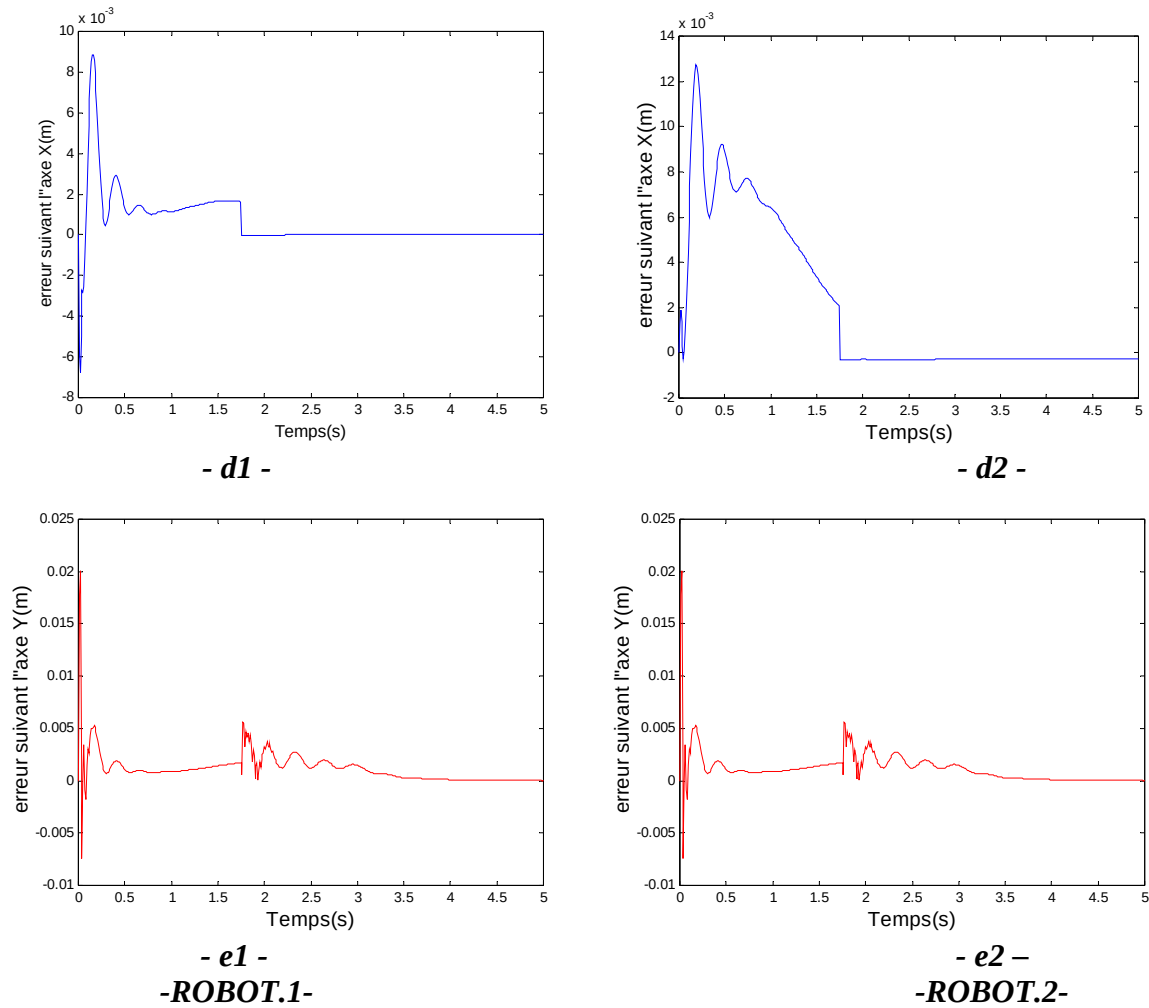
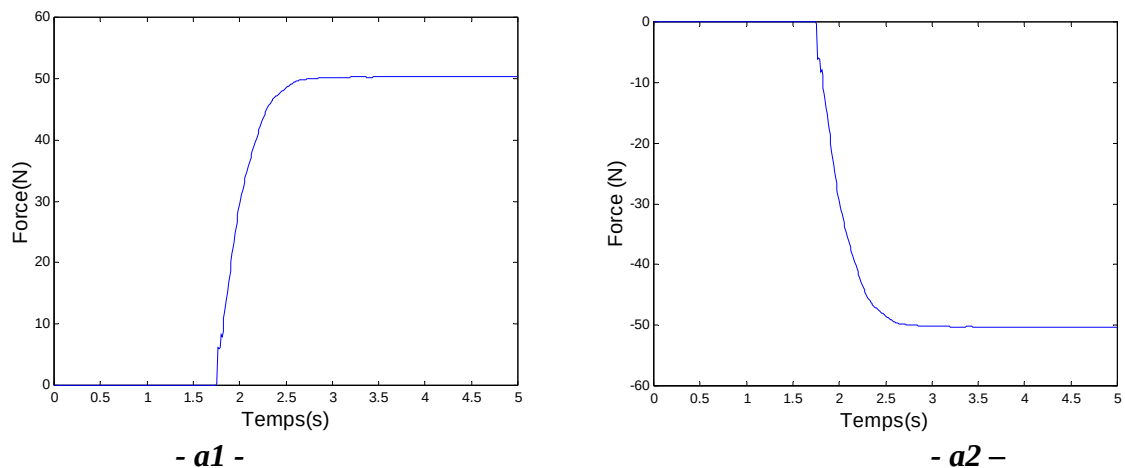


Figure 9. Mouvement des deux bras manipulateurs redondants dans le repère tâche.

- a. Suivis de trajectoires des deux bras dans le plan (X, Y).
- b1. et b2. Trajectoires des effecteurs selon l'axe «X».
- c1. et c2. Trajectoire des effecteurs selon l'axe «Y».
- d1. et d2. Erreur de suivi de trajectoires selon l'axe « X ».
- e1. et e2. Erreur de suivi de trajectoires selon l'axe « Y ».

Sur la figure 10 sont représentées les courbes des efforts obtenus.



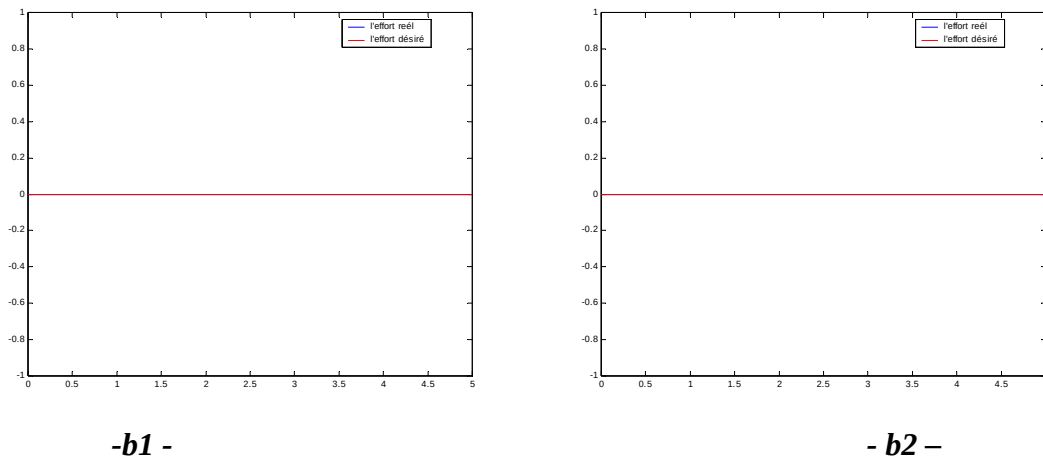


Figure 10. Réponse en effort.

- $a1$ et $a2$ Effort selon l'axe X respectivement du premier et deuxième bras
- $b1$ et $b2$ Effort selon l'axe y respectivement du premier et deuxième bras

Les courbes obtenues montrent que dans l'intervalle de temps $[0 \text{ s}, 1.75 \text{ s}]$ les forces d'interactions des deux bras manipulateurs sont nulles. En effet, les mouvements sont dans cette première phase non contraints (absence de tout contact). Une fois le contact établi avec l'objet, les robots doivent maintenir un effort désiré selon la direction X normal à la surface de l'objet de 50 N sur chaque extrémités de la barre. Les efforts selon l'axe Y restent nuls (figure 10.b1 et b2). Ceci s'explique par le fait que la saisie de l'objet est réalisée suivant l'axe X . Quant à la courbe des efforts suivant l'axe « X », elle fait apparaître au moment du contact, une augmentation en valeur absolue des efforts pour atteindre l'effort de serrage désiré (50 N). Cette valeur est atteinte au bout de 0.75 s .

Conclusion

Dans ce chapitre, une extension de la commande en effort externe basée sur les concepts neuro-flous est proposée pour la réalisation d'une tâche de coopération de deux bras manipulateurs redondants. L'architecture choisie est hiérarchisée. Ceci dit, les deux bras manipulateurs jouent un rôle symétrique et contribuent simultanément au déplacement de l'objet. Ainsi, Les efforts et les déplacements sont répartis symétriquement sur chaque bras.

Le rôle du niveau superviseur est de calculer à partir des grandeurs mesurées les consignes à envoyer aux boucles d'asservissements de chaque bras manipulateur. Une structure de commande hybride utilisant les concepts neuro-flous permet de réaliser sur chacun des bras un contrôle simultané des forces et des positions. La coordination de deux bras manipulateurs passe donc par une répartition judicieuse des consignes.

Afin d'évaluer les performances de la structure de commande proposée, nous nous sommes intéressés au problème du transport d'une barre rigide. La barre doit être déplacée en maintenant tout au long du mouvement un effort de serrage de 50 N . Compte tenu, de la configuration de la tâche dans le plan, la saisie de la barre est réalisée selon la direction X .

Cette tâche fait apparaître deux phases de mouvements. Dans la première les bras manipulateurs utilisés se déplacent dans un espace libre. Dans la seconde phase ils entrent en contact avec la barre, la saisissent et la transportent jusqu'à la position finale spécifiée. Ces mouvements sont donc contraints.

Les résultats obtenus montrent que les suivis de trajectoires sont satisfaisants. Toute fois, au démarrage les couples sont élevés. Ces couples sont nécessaires pour vaincre l'inertie initiale des bras. Lors de la prise de l'objet quelques oscillations apparaissent au niveau des couples de commande.

Les courbes représentant les forces de contacts montrent que la composante de l'effort selon la direction Y reste nulle tout au long du mouvement. Alors que la composante de l'effort selon la direction X à l'occurrence du contact augmente en valeur absolue pour atteindre l'effort de serrage spécifié. Ceci s'explique par le fait que la saisie de la barre est réalisée selon cette direction.

Conclusion Générale

Au terme de ce manuscrit, nous nous proposons de faire un récapitulatif de notre travail, d'analyser globalement les résultats obtenus et enfin de dresser des perspectives qui permettent d'envisager de nombreux axes de recherche comme suite logique de ce travail.

Nous avons présenté dans ce mémoire, trois nouvelles approches de commandes pour traiter le problème du contrôle en effort des robots manipulateurs redondants compliantifs actifs qui sont en interaction avec leurs environnements. Ce type de tâches implique des contraintes très sévères et constitue en général un processus qui évolue dans le temps et l'espace. Les approches proposées sont basées soit sur la commande dynamique, soit sur les concepts de la logique floue ou sur les concepts des réseaux neuro-flous.

Pour exposer notre démarche, nous avons dans une première étape (chapitre I) présenté la problématique du contrôle en effort dans le cas d'une tâche de coopération de deux bras manipulateurs redondants. Il a été montré que les méthodes à compliance active pouvaient résoudre ce problème. Une stratégie de mise en œuvre d'une tâche de saisie et de transport d'objet par deux robots a été ainsi présentée. Cette stratégie présente deux contraintes à respecter : imposer un effort désiré indépendamment du mouvement, coordonner les mouvements des deux bras manipulateurs.

Pour mettre en œuvre une structure de commande, nous devons disposer d'une représentation mathématique des modèles du système considéré. Ceci dit, nous avons présenté dans le deuxième chapitre les différents modèles du robot redondant considéré.

Les tâches robotisées étant généralement spécifiées dans le repère opérationnel. Il est nécessaire de convertir trajectoires au repère articulaire. Cette opération consiste à la résolution du modèle géométrique inverse. Nous avons présenté dans ce mémoire (chapitre III) différentes méthodes permettant de résoudre ce problème. Il a été montré que l'inconvénient majeur de certaines d'entre elles est la difficulté de la détermination du pas d'apprentissage. Pour d'autres des divergences ou des oscillations peuvent se produire, d'où la nécessité de passer dans ce cas au modèle quadratique du second ordre pour assurer la convergence.

Deux méthodes de résolution du modèle géométrique inverse sont proposées. La première est basée sur une approximation linéaire en utilisant l'algorithme de Levenberg-Marquardt. La seconde est basée sur la méthode d'identification du modèle inverse en utilisant une approche connexionniste. En effet, elle utilise les réseaux de neurones à mémoires. Ainsi, nous avons décrit l'outil réseau de neurones à mémoires et son principe d'utilisation pour l'identification de systèmes.

Pour mettre en évidence les limites des commandes conventionnelles, nous avons présenté dans le quatrième chapitre différentes structures de commandes hybrides force-position. De plus, certaines d'entre elles nécessitent des transformations géométriques et une matrice de sélection destinée à satisfaire l'orthogonalité, rendant ainsi leurs synthèses et leurs mises en œuvre difficiles contrairement à la structure de commande en effort externe. En effet, cette dernière est basée sur une hiérarchisation de la boucle d'effort sur la boucle de

position. Ainsi, elle n'utilise pas de matrice de sélection. Cette structure est donc adoptée pour la suite de notre travail.

Nous avons présenté dans le chapitre V la synthèse des trois structures de commandes en effort externe proposées. La première (Partie I) utilise la méthode de commande dynamique pour la synthèse de la loi de commande en position. Sa mise en œuvre dans le cas d'un bras manipulateur redondant a permis de la valider. En effet, pour les différentes tâches considérées les résultats obtenus sont satisfaisants. Toutefois, son inconvénient majeur réside dans la nécessité de disposer d'un modèle fiable de l'ensemble robot-environnement.

Afin de contourner ce problème deux approches de commandes basées sur l'intelligence artificielle (I.A) sont proposées. L'une utilise le concept de la logique floue (Partie II), l'autre le concept de la logique neuro-floue (Partie III). Ces deux approches de commandes ont été, elle aussi, validées sur le système considéré pour différentes configurations de tâches. La mise en œuvre des structures proposées a été réalisée avec le logiciel MATLAB. Toutefois, aucune fonction préprogrammée n'est utilisée. Les résultats obtenus sous forme de courbes ont montré que les suivis sont de bonnes qualités pour toutes les tâches considérées.

Pour que notre étude soit complète, nous présentons dans la dernière partie de ce chapitre (Partie IV) une étude comparative des trois structures de commandes proposées. Les tâches robotisées étant généralement spécifiées dans le repère tâche; nous nous sommes limités, dans la présente étude, aux tâches spécifiées dans le repère opérationnel. Les résultats obtenus sous forme de courbes font apparaître un bon suivi de trajectoires pour les trois approches de commandes. Toutefois, la commande basée sur les concepts des réseaux neuro-flous présente une erreur en régime permanent plus faible. Ainsi, nous l'avons adoptée pour la suite de notre travail.

Dans le chapitre VI nous nous sommes intéressés à une tâche de coopération de deux bras manipulateurs redondants. Ceci dit, nous avons mis en œuvre la structure de commande en effort externe basée sur les concepts neuro-flou pour la réalisation d'une tâche de saisie et de transport d'objets. Une analyse des avantages et inconvénients des architectures existantes nous a permis d'adopter une architecture de commande hiérarchisée symétrique. Les résultats obtenus prouvent l'efficacité de l'approche proposée.

Une analyse de l'approche de commande retenue montre qu'elle présente les avantages suivants :

- ❖ Facilité de mise en œuvre;
- ❖ Facilité de configuration à partir d'un ensemble d'entrées-sorties;
- ❖ Facilité de programmation d'une tâche;
- ❖ Non nécessité de disposer d'une connaissance fine du système robot-environnement;
- ❖ Très utile pour remédier au problème de complexité de la modélisation.

Cependant, la seule contrainte liée à l'utilisation de cette approche est :

- ❖ Le calcul des gains de normalisation.

Compte tenu des différents points que nous avons abordés au cours de ce travail, nous proposons dans le cadre de travaux futurs de traiter les aspects suivants:

- ❖ Implémenter un algorithme d'adaptation en lignes des paramètres de la commande proposée.

- ❖ Comparer les deux méthodes d'inversion du modèle géométrique de bras manipulateur redondant.
- ❖ Tester la commande proposée sur un bras manipulateur redondant réel.
- ❖ Valider l'approche dans d'autres types de tâches comme le suivi de contour à grande vitesse;
- ❖ Extension de l'approche pour traiter le problème complexe du choc lorsque la vitesse d'accostage est très élevée;
- ❖ Implémenter un module pour traiter le problème d'évitement de collisions entre robots manipulateurs;
- ❖ Comparer les performances de l'approche par rapport à d'autres approches de commande (d'autres structures de commande).

ANNEXE 1

TRANSFORMATION DE DENAVIT - HARTENBERG

Tout manipulateur peut être considéré comme une chaîne de liaisons connectées par des articulations. Chaque liaison est caractérisée par son repère.

En utilisant les transformations homogènes, on peut décrire la position et l'orientation de chaque repère par rapport à un autre. Denavit et Hartenberg ont établi une transformation qui permet le passage d'une liaison à la suivante, en utilisant quatre paramètres.

L'implémentation des repères dans chaque liaison; en utilisant la transformation de Denavit et Hartenberg, est présentée ci-dessous [Pru 88] [Vib 87]:

Pour l'application de cette transformation, il faut d'abord fixer les différents repères dans chaque articulation d'une façon successive, en procédant suivant les étapes suivantes :

Etape 1 : Numéroté chaque liaison et articulation, en commençant de la base, notée liaison 0, et l'élément terminal, notée liaison n voir figure 1.

La liaison i se déplace soit autour (rotative) ou le long (prismatique) de la liaison $i-1$.

Etape 2 : Etablir le repère de chaque articulation, en suivant les règles suivantes :

- L'axe Z_{i-1} est choisi le long de l'axe de l'articulation i .
- L'axe X_i est choisi perpendiculaire à Z_{i-1} , dont le sens peut être choisi arbitrairement ($X_i = Z_{i-1} \times Z_i$).
- L'axe Y_i est choisi, de telle sorte à former un trièdre droit.

Etape 3 : définir les paramètres θ_i , d_i , a_i et α_i :

- θ_i : Angle entre X_{i-1} et X_i , obtenu par rotation de X_{i-1} vers X_i autour de Z_{i-1} .
- d_i : coordonnées de O_i dans R_{i-1} , le long de Z_{i-1} .
- a_i : distance entre Z_{i-1} et z_i , selon l'axe X_i .
- α_i : angle entre Z_{i-1} et Z_i .

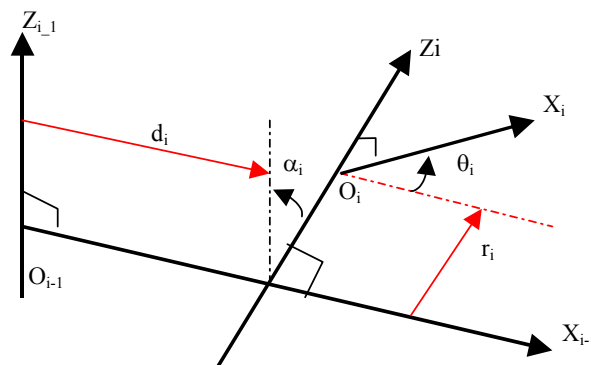


Figure 1. Paramétrage de Denavit - Hartenberg.

L'implémentation de cet algorithme dans le cas du robot redondant à trois degrés de liberté.

Les paramètres de la transformation de Denavit et Hartenberg du robot, son regroupés dans le tableau suivant :

Articulation	θ_i	a_i	d_i	α_i
1	θ_1	l_1	0	0
2	θ_2	l_2	0	0
3	θ_3	l_3	0	0

Tab.1 : Paramétrage de Denavit - Hartenberg du robot redondant.

Soit ${}_{i-1}T^i$ la matrice de coordonnées homogènes (ou matrice de passage) qui relie le repère R_i au repère R_{i-1} , elle s'exprime sous la forme suivante.

$${}_{i-1}T^i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ainsi nous tirons les matrices de transformation, représentant les trois rotations d'un repère à un autre.

$$T_0^1 = \begin{bmatrix} c_1 & -s_1 & l_1 c_1 \\ s_1 & c_1 & l_1 s_1 \\ 0 & 0 & 1 \end{bmatrix} \quad T_1^2 = \begin{bmatrix} c_2 & -s_2 & l_2 c_2 \\ s_2 & c_2 & l_2 s_2 \\ 0 & 0 & 1 \end{bmatrix} \quad T_2^3 = \begin{bmatrix} c_3 & -s_3 & l_3 c_3 \\ s_3 & c_3 & l_3 s_3 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Rightarrow T_0^3 = \begin{bmatrix} c_2 & -s_2 & 0 & l_2 c_2 \\ s_2 & c_2 & 0 & l_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

avec : $c_i = \cos \theta_i$, $s_i = \sin \theta_i$, $c_{ij} = \cos(\theta_i + \theta_j)$ et $s_{ij} = \sin(\theta_i + \theta_j)$.

ANNEXE 2

EQUATIONS DE PROPAGATION ET DE RETROPROPAGATION D'UN RESEAU DE NEURONES A MEMOIRES MULTICOUCHE

Soit la structure de réseau de neurones à mémoires représentée sur la figure suivante:

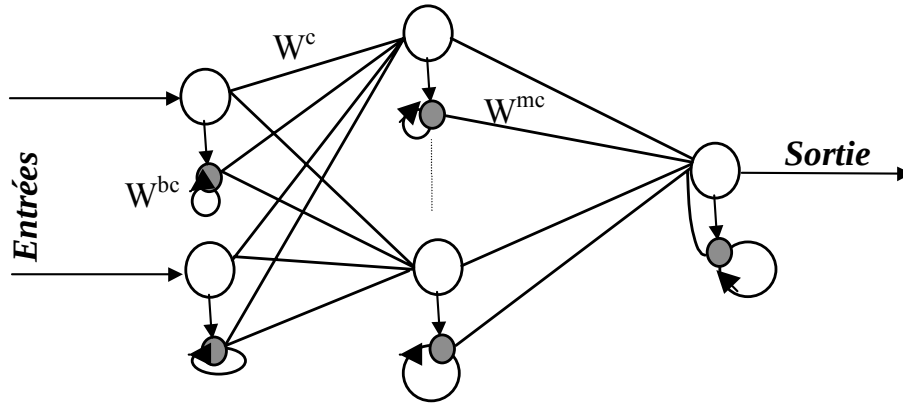


Figure 1. Structure du réseau de neurones à mémoires

Avec:

- W^c , W^{mc} , W^{bc} sont les poids des connexions (neurones à neurones, cellules mémoire à neurone, cellule mémoire à elle même) entre la couche « c » et la couche « c+1 ».
- W^0 est choisi nul pour éviter une connexion directe entre la couche d'entrée et la couche de sortie.
- B^c est le vecteur des biais associés aux neurones de la couche cachée « c » (avec $c \neq 0$: couche d'entrée).
- A^c Vecteur des activations des neurones de la couche « c »
- V^c est le vecteur de sortie des neurones de la couche cachée « c »
- V^{mc} et le vecteur de sortie des cellules mémoires de la couche « c »
- NBCC : nombre de neurones de la couche cachée.
- NBCE : nombre de neurones de la couche d'entrée du réseau.
- NBCS : nombre de neurones de la couche de sortie.

La base d'apprentissage est composée de deux vecteurs: L'entrée « u » et la sortie « Y ».

A. PROPAGATION

Equations de la couche d'entrée:

$$V_i^0(k) = u_i(k)$$

$$V^{m0}(k) = M^0 \cdot V^{m0}(k-1) + (I - M^0) \cdot V^0(k-1) \text{ . Avec } M^0 = \text{matrice diagonale } (W^{b0})$$

Equations de toutes les couches cachées « c » variant de 1 à NBCC

$$V^c(k) = f^c(A^c(k)) \text{ .}$$

Avec:

$A^c(k) = W^{m(c-1)} \cdot V^{m(c-1)}(k) + W^{(c-1)} \cdot V^{(c-1)}(k) + B^c$,
 f est la fonction sigmoïde. W et W^m sont des vecteurs.

D'où:

$$V^c(k) = f^c \left(\sum_{j=1}^{NBC(c-1)} (W_j^{m(c-1)^T}(k) \cdot V_j^{m(c-1)}(k) + W_j^{(c-1)^T}(k) \cdot V_j^{(c-1)}(k)) + B^c \right)$$

$$V^{mc}(k) = M^c \cdot V^{mc}(k-1) + (I - M^c) \cdot V^c(k-1)$$

Avec $M^c = \text{matrice diagonale } (W^{bc})$.

Equation de la sortie du neurone « s » de la couche de sortie N:

$$V_s^N(k) = A_s^N(k) = W_s^{N-1^T} \cdot V^{N-1}(k) + W_s^{m(N-1)^T} \cdot V^{m(N-1)} + B_s^N$$

B. EQUATIONS DE LA DOUBLE RETROPROPAGATION:

Equations de Lagrange:

Critère $J = \frac{1}{NBC(N)} \sum_{s=1}^{NBC(N)} J_s$

Avec $J_s = \frac{1}{2} \sum_k (y_s(k) - yd_s(k))^2$

Ainsi le Lagrangien de notre problème d'optimisation est:

$$\begin{aligned} L(X, \psi, W) = & \frac{1}{2 \cdot NBC(N)} \sum_{s=1}^{NBC(N)} \sum_k (((W_s^{N-1^T} \cdot V^{N-1}(k) + W_s^{m(N-1)^T} \cdot V^{m(N-1)}(k) + B_s^N) - yd_s(k))^2) + \\ & \sum_{c=2}^{N-1} \left(\sum_k \psi_k^{c^T} \cdot (V^c(k) - f^c \left(\sum_{j=1}^{NBC(c-1)} (W_j^{m(c-1)^T}(k) \cdot V_j^{m(c-1)}(k) + W_j^{(c-1)^T}(k) \cdot V_j^{(c-1)}(k)) + B^c \right) \right. \\ & \left. + \sum_k \psi_k^{mc^T} \cdot (V^{mc}(k) - M^c \cdot V^{mc}(k-1) - (I - M^c) \cdot V^c(k-1)) \right) + \\ & \sum_k \psi_k^{1^T} \cdot (V^1(k) - f^1 \left(\sum_{j=1}^{NBC(0)} (W_j^{m0^T}(k) \cdot V_j^{m0}(k)) + B^1 \right) + \\ & \sum_k \psi_k^{m1^T} \cdot (V^{m1}(k) - M^1 \cdot V^{m1}(k-1) - (I - M^1) \cdot V^1(k-1)) + \\ & \sum_k \psi_k^{m0^T} \cdot (V^{m0}(k) - M^0 \cdot V^{m0}(k-1) - (I - M^0) \cdot V^0(k-1)) \end{aligned}$$

Dérivées partielles:

Pour la couche N-1: (Cas particulier car pas de fonction nonlineaire)

$$\frac{\partial L}{\partial V^{N-1}(k)} = \sum_{s=1}^{NBC(N)} ((y_s(k) - yd_s(k)) \cdot W_s^{N-1} + \psi_k^{N-1} - (I - M^{N-1}) \cdot \psi_{k+1}^{m(N-1)}) = 0$$

$$\frac{\partial L}{\partial V^{m(N-1)}(k)} = \sum_{s=1}^{NBC(N)} ((y_s(k) - yd_s(k)) \cdot W_s^{m(N-1)} + \psi_k^{m(N-1)} - M^{N-1} \cdot \psi_{k+1}^{m(N-1)}) = 0$$

Pour toute couche cachée « c » avec « c » compris entre 1 et N-2:

$$\frac{\partial L}{\partial V^c(k)} = \psi_k^c - (I - M^c) \cdot \psi_{k+1}^{mc} - W^{c^T} F_k^{c+1} \psi_k^{c+1} = 0$$

$$\frac{\partial L}{\partial V^{mc}(k)} = \psi_k^{mc} - M^c \cdot \psi_{k+1}^{mc} - W^{mc^T} F_k^{c+1} \psi_k^{c+1} = 0$$

Pour la couche 0 (Cas particulier car W^0 et nul):

$$\frac{\partial L}{\partial V^{m0}(k)} = -W^{m0^T} F_k^1 \psi_k^1 + \psi_k^{m0} - M^0 \psi_{k+1}^{m0} = 0$$

La résolution de ce système d'équations nous permet de déduire les équations suivantes :

Equation de récurrence pour la double rétro-propagation:

Les formules suivantes sont dans ordre de la rétropropagation spatiale du haut vers le bas (index c), et de droite à gauche pour la rétropropagation temporelle (index k).

Pour la couche N-1:

$$\psi_k^{m(N-1)} = - \sum_{s=1}^{NBC(N)} ((y_s(k) - yd_s(k)) \cdot W_s^{m(N-1)}) + M^{N-1} \cdot \psi_{k+1}^{m(N-1)} \quad \forall k$$

$$\psi_K^{m(N-1)} = - \sum_{s=1}^{NBC(N)} ((y_s(K) - yd_s(K)) \cdot W_s^{m(N-1)}) \text{ au dernier l'instant } k$$

$$\psi_k^{N-1} = (I - M^{N-1}) \cdot \psi_{k+1}^{m(N-1)} - \sum_{s=1}^{NBC(N)} ((y_s(k) - yd_s(k)) \cdot W_s^{N-1}) \quad \forall k$$

$$\psi_K^{N-1} = - \sum_{s=1}^{NBC(N)} ((y_s(K) - yd_s(K)) \cdot W_s^{N-1}) \text{ au dernier instant } k.$$

Pour la couche c , c allant de 1 à N-2 :

$$\psi_k^{mc} = M^c \cdot \psi_{k+1}^{mc} + W^{mc^T} F_k^{c+1} \psi_k^{c+1}$$

$$\psi_k^c = (I - M^c) \cdot \psi_{k+1}^{mc} + W^{c^T} F_k^{c+1} \psi_k^{c+1}$$

Pour la couche 0 :

$$\psi_k^{m0} = M^0 \psi_{k+1}^{m0} + W^{m0^T} F_k^1 \psi_k^1 \text{ à l'instant } k.$$

$$\psi_K^{m0} = W^{m0^T} F_K^1 \psi_K^1 \text{ à l'instant final.}$$

Dérivées partielles pour le calcul du gradient

$$\text{Au point optimal: } \nabla J = \nabla L \Rightarrow \frac{\partial J}{\partial W} = \frac{\partial L}{\partial W} \quad \forall W \in \{W^c, W^{mc}, W^{bc}, B^c\}.$$

$$\frac{\partial L}{\partial W^0} = 0, \quad \frac{\partial L}{\partial W_i^c} = - \sum_k F_k V_i^c(k) \psi_k^{c+1} \text{ avec } c \text{ variant de } 1 \text{ à } N-2$$

$$\frac{\partial L}{\partial W_s^{N-1}} = \sum_k (y_s(k) - yd_s(k)) V^{N-1}(k)$$

$$\frac{\partial L}{\partial W_i^{mc}} = - \sum_k F_k V_i^{mc}(k) \psi_k^{c+1} \text{ avec } c \text{ variant de } 0 \text{ à } N-2$$

$$\frac{\partial L}{\partial W_s^{m(N-1)}} = \sum_k (y_s(k) - yd_s(k)) V^{m(N-1)}(k)$$

$$\frac{\partial L}{\partial W^{bc}} = \frac{\partial L}{\partial M^c} = - \sum_k (V^{mc}(k-1) - V^c(k-1)) \psi_k^{mc} \text{ avec } c \text{ variant de } 0 \text{ à } N-1$$

$$\frac{\partial L}{\partial B^c} = - \sum_k F_k \psi_k^c \text{ avec } c \text{ variant de } 1 \text{ à } N-1 \text{ et } \frac{\partial L}{\partial B_s^N} = \sum_k (y_s(k) - yd_s(k))$$

Les poids et les biais du réseau de neurone seront adaptés comme suit:

$$W_{k+1} = W_k + \eta \Delta W_k \text{ et } B_{k+1} = B_k + \eta \Delta B_k.$$

Le pas d'apprentissage η est adapté en utilisant l'algorithme de Goldstein

Remarque:

La règle d'optimisation unidimensionnelle appliquée au pas d'apprentissage est: "la règle de **Goldstein**". Cette règle peut s'écrire comme suit:

a) Posons $\eta_{\min} = 0$ and $\eta_{\max} = 10$

Calculer $\dot{g}(0) = \nabla E^t(w) \cdot P$

Choisir une valeur initiale pour η

b) Calculer $g(\eta) = E(w + \eta \cdot P)$

Si $g(\eta) \leq g(0) + \xi_1 \cdot \eta \cdot \dot{g}(0)$ go to (c) (*)

Sinon faire $\eta_{\max} = \eta$ et aller en (e)

c) Comparer $\dot{g}(\eta)$ et $\dot{g}(0) + \xi_2 \cdot \eta \cdot \dot{g}(0)$

si $\dot{g}(\eta) \geq \dot{g}(0) + \xi_2 \cdot \eta \cdot \dot{g}(0)$ Fin de l'algorithme (**)

sinon $\eta_{\min} = \eta$

e) Choisir une nouvelle valeur de η dans l'intervalle $[\eta_{\min}, \eta_{\max}]$ et retourner en (a).

L'équation (*) garantit que η n'est pas très grand et l'équation (**) assure que η n'est pas trop petit ($\xi_1 \in]0, 1[$ $\xi_2 \in]\xi_1, 1[$).

Références

Bibliographiques

- [Ahm 93] S. Ahmed, C. Klein and C.Chu-Jeng "Use of an extended Jacobian method to map algorithmic singularities" in IEEE Conference on Robotics and Automation, (Atlanta), PP.632-637, 1993.
- [Ama 97] A. Ramdane-Chérif "Inversion des modèles géométrique et cinématique d'un robot redondant : une solution neuronale adaptative", Thèse de Doctorat en informatique université Pierre et Marie Curie, Paris VI, 1998.
- [Bab 98] R. Babuska, "Fuzzy Modelling : Principals, Methods and Application", Control Enrgineering Labortory, Faculty of Information Technology and systems 1998.
- [Bal 84] J. Baillieul, J. Hollerbach and R.Brockett, "Programmming and Control of Kinematically redundant mnipulators", In of 23rd Conference on Decision and control, PP.768-774, Decembre 1984.
- [Bal 95] N. Bali "Etude des performances de la commande prédictive généralisée: application au Robots PUMA et SCARA », Thèse de magister, école nationale polytechnique (E.N.P)1995.
- [Ben 02] A. Benallegue, B. Daachi and A.Ramdane-Chérif, "Kinematic Inversion", In IEEE International Conference on Intelligent Robots and systems, PP.1904-1912, Octobre 2002.
- [Ber 03] A. Berni, A. Ramdane-Chérif, N. Saadia et N. Levy "Exploring Cognitive approach through the Neural Network paradigm », The 2nd IEEE International Conference on Cognitive Informatics, ICCI'03., pp.47-54, 18-20 Aout 2003, London, Angleterre.
- [Ber 95] A. Berthe "Etude de la robustesse de schémas de commande force-position pour un robot à deux bras », Thèse de Doctorat Montpellier II, 1995.
- [Bou 02] D. Boukhetala "Contribution à la commande décentralisée des systèmes complexes interconnectés : Application à la robotique », Thèse de Doctorat, école nationale polytechnique (E.N.P) 2002.
- [Bou 99] M. BouYoucef « La commande à structure variables par régime glissant et à logique floue appliquée à un robot manipulateur », Thèse de Magistère 1999 (U.S.T.H.B).
- [Bül 94] H. Bhüler, "Réglage par Logique floue" PPR, 1994.

- [Che 88] A. C. Chevallereau and W. Khalil, "*A New method for solution of the inverse kinematics redundant robots*", In IEEE International Conference on Robotics and Automation, (Atlanta), PP.37-42, 1988.
- [Chi 92] P. Chiacchio, S. Chiaverini and B. Siciliano "*Cooperative control schemes for multiple robot manipulators systems*", Proceedings of IEEE International Conference on Robotics and Automation, PP.2218-2222, 1992.
- [Chi 94] Chin-teng Lin , C.S.G. Lee "*Reinforcement structure/ parameter Learning for Neural Network based Fuzzy Logic systems*", IEEE Transaction on Fuzzy systems 1994.
- [Chu 94] Chuen-Tsai Sun "*Rule base structure identification in an adaptive network based fuzzy inference system*", IEEE Transaction on Fuzzy systems 1994.
- [Chu 95] Chuen-Tsai Sun, J.R.Jang "*Neuro-Fuzzy Modeling and control*", IEEE Transaction on Fuzzy systems. Vol.83. No3. pp 378-405, Mars 1995.
- [Coi 81] P. Coiffet, "*Les Robots, Tome 1: Modélisation et commande*", Hermès, Paris 1981.
- [Deg 93] E. Degoulange, "*Commande en effort d'un robot manipulateur a deux bras : application au contrôle de la déformation d'une chaîne cinématique fermée*", Thèse de Doctorat université Montpellier II. 1993.
- [Der 03] N. Derragui, "*Commande coordonnée de robots coopérants*", Thèse de Magistère en électronique USTHB 2002/2003.
- [Dou 97] S. Dourlens, "*Contrôle latéral de véhicule par réseau de neurones dynamique*", Rapport de DEA, Université Pierre et Marie Curie, Paris VI, 1997.
- [Dro 97] A.Ramdane-Chérif, V.Perderau, and M.Drouin, "*Inverse kinematic solution based on Lyapunov function for redundant and non-redundant robots*", In CIRA'97, IEEE International symposium on computational intelligence in robotics and automation, PP. 226-231, Juillet 10-11 1997.
- [Fra 94] P. Fraisse, "*Contribution à la commande robuste position-force des robots manipulateurs à architecture complexes : application à un robot à deux bras*", Thèse de Doctorat Université Montpellier II 1994.
- [Gue 88] A. Guez and Z. Ahmed, "*A solution to the inverse kinematic problem in robotics by neural networks*", IEEE International conference on Neural Networks. pp.617-621, Juillet 1988.
- [Gun 95] L. Guenfaf, "*Etude de différentes stratégies de la commande adaptative*", Thèse de magister, école nationale polytechnique (E.N.P)1995.
- [Guo 00] A. Gouassmia, "*Application de la commande floue et neuro-floue à structure décentralisée au robot à deux axes rotoïdes*", Thèse d'ingénieur d'état en Electronique 2000 (U.S.T.H.B).

- [Hay 88] S.Hayati, "*Position and force control coordinated mutiple arms*", IEEE Transactions on Aerospace and electronic systems, Vol.24.No 5. pp.584-590, Septembre 1988.
- [Hog 87] N. Hogan, "Stable execution of contact tasks using impedance control", IEEE International confrence on robotics and Automation, U.S.A, pp 1047-1045, Avril 1987.
- [Kha 01] O. Khatib, "Fuzzy logic control of a robot manipulator based on visual servoing", IEEE Transaction on Robotics and Automation 2001.
- [Kha 86] O. Khatib et J. Burdick, "*Motion and force control of robot manipulators*", Proceeding of the IEEE International Conference on Robotics and Automation, San Francisco, USA, Avril 1986, pp. 1381-1386.
- [Kha 87] O. Khatib, "*Object manipulation in a multi-effector robot system*", Proceeding of the International Symposium of Robotics Research, Santa Cruz, USA, pp. 131-138. Août 1987.
- [Kha 99] W. Khalil, "Modélisation, Identification et Commande », Hermès, Paris 1999.
- [Kig 95] K. Kiguchi, " *Fuzzy neural controller for robot manipulator force control*", IEEE Transaction on Robotics and Automation 1995.
- [Kle 95] C. A. Kleinn, C. Chu-Jeng and A. Shamim, "*A New formulation of the extended jacobian method and use in mapping algorithmic singularities for kinematticaly redundant manipulators*", IEEE Transaction on Robotics and Automation, Vol.10, No.1, Février 1995.
- [Lad 00] A. Ladj, "Etude comparative de deux structures de commande hybride force-position: Application au suivi de contour", Thèse d'ingénieur d'état en électronique 2000 (U.S.T.H.B).
- [Las 02] B. Lassami, "*Développement d'un logiciel d'animation et de commande pour bras manipulateurs*", Thèse d'ingénieur d'état en automatique école nationale polytechnique (E.N.P), 2002.
- [Lee 90a] C. C. Lee, "Fuzzy Logic in control systems: fuzzy logique controller- part I", IEEE Trans. Syst. Man and Cybern, Vol 20,No 2. pp 404-418 Février 1990.
- [Lee 90b] C. C. Lee, "Fuzzy Logic in control systems: fuzzy logique controller- part II", IEEE Trans.syst. Man and Cybern, Vol 20,No 2. pp 4194-435 ,1990.
- [Lor 88] L. Sciavicco and B.Siciliano, "*A solution algorithm to the inverse kinematic problem for redundant manipulators*", IEEE Journal of Robotics and Automation, Vol.4, No.4, Aout 1988.

- [Lor 90] R. Loranço, "Adaptative hybrid force-position control for redundant manipulators", Proceeding of 29th conference on decision and control, 1990.
- [Mas 81] M.T. Mason, "*Compliance and force control for computer controlled manipulators*", IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-11, N°6, Janvier 1981, pp. 418-432.
- [Mez 03] Z.Mezerrag, "Etude, Conception et Réalisation d'une carte de commande à base d'un microcontrôleur pour un robot manipulateur à trois axes", Thèse d'ingénieur d'état en électronique (U.S.T.H.B), 2003.
- [Moh 96] A. Mohamed and A.C. Chevallereau, "*Resolution of robot redundancy in the cartesian space by criteria optimization*", In IEEE Conference on Robotics and Automation, (Atlanta), PP.646-651, 1993.
- [Per 03a] V. Perdereau, C. Remond et M. Drouin, "*Commande d'une main robotisée : Optimisation des forces*", Rapport interne de LISIF, Université de Pierre Marie Curie, 2003.
- [Per 03b] V. Perdereau, C. Remond et M. Drouin, "*Reconfiguration de la prise d'un objet par une main articulée*", Rapport interne de LISIF, Université de Pierre Marie Curie, 2003.
- [Per 91] V. Perdereau, "Contribution à la commande hybride force-position", Thèse de Doctorat Pierre et Marie curie, Paris VI, 1991.
- [Pru 88] A. Pruski, « *Robotique Générale* », ed. Ellipses, Paris 1988.
- [Puj 95] A. Pujas, "*Etude de la robustesse de schémas de commande Position-Force pour robots à deux bras*", Thèse de Doctorat, Université Montpellier II, 29 juin 1995.
- [Rai 81] M. H. Raibert et J.J. Craig, "*Hybrid Position-Force control of manipulators*", Transactions of ASME, Vol. 102, pp. 126-133, 1981,
- [Ram 02] M.C.Ramos, Jr. And A.J.Koivo, Life Fellow "*Fuzzy Logic-based optimization for redundant manipulators*" IEEE Transaction on fuzzy systems, Vol 10, No.4 Aout 2002.
- [Ram 94] A. Ramdane-Chérif, V. Perdereau, and M.Drouin, "*Inverse kinematic resolution using neural network*", in IASTED, IEEE International conference on Robotics and Manufacturing, pp.170-172, Janvier 1995.
- [Reb 85] C. Reboulet, A. Robert, H. Poilve, et A. Gaillet, "*Hybrid position force control: Application to assembly*", Acts of the 15th International symposium on Industrial Robot, Tokyo, Japan, Septembre 1985, pp. 157-164.
- [Reb 85b] C. Reboulet et A. Robert, "*Hybrid control of manipulator equipped with an active compliant wrist*", Acts of the 3rd international symposium on Robotics Research, Gouvieux, France, Octobre 1985, pp. 237-241.
- [Ren 84] B. Gorla, M. Renaud, "*Modèles des Robots manipulateurs, application à leur commande*", ed. Cepadues, Toulouse 1984.
- [Rod 97] Rodini, "*Modélisation, analyse et méthodologies de la commande floue d'un bras manipulateur de robot flexible* » Thèse de magistère, Ecole polytechnique (ENP), 1997 .

- [Saa 95] N. Saadia, Y. Amirat, "*Commande hybride force-position selon une approche neuronale: Application à la réalisation d'une tâche d'insertion*", Rapport interne, LIIA, Paris XII, Juillet 1995.
- [Saa 97] N. Saddia, "Contribution à la commande hybride force-position des robots compliant selon un approche neuronale", Thèse de Doctorat, Université Paris XII 1997.
- [Sas 94] P. S. Sastry, G. Santharam and K. P. Unnikrishnan, "Memory neuron networks for identification and control Of dynamical systems", IEEE .1994
- [Ser 86] H. Seraji, "*On decentralized control of non-linear three link robots*" IEEE Transaction on Robotics and Automation, pp.134-138, 1986.
- [Ser 89a] H. Serraji, "*Decentralized adaptive control of manipulators: theory, simulation, and experimentaion*", IEEE Transaction on Robotics and Automation. Vol 5, No.2, pp. 183-201, Avril 1989.
- [Ser 89b] H. Serraji, "*Configuration Control of redundant manipulators: theory, simulation, and implementaion*", IEEE Transaction on Robotics and Automation. Vol 5, No.2, pp. 472-490, Avril 1989.
- [Ser 89c] H. Seraji, "*Case studies in configuration control for redundant robots*", Proceeding of 28th conference on decision and control, pp. 698-704, Decembre 1989.
- [Sic 87] L. Sciavicco and B.Siciliano, "*A dynamic solution to the inverse kinematic problem for redundant manipulators*", IEEE International conference on Robotics and Automation, pp.1081-1087, Avril 1987.
- [Uch 87] M. Uchiyama, N. Iwasawa and K. Hakomori, "*Hybrid position / force control for coordination of two arm robots*", Actes IEEE International Conference on Robotics and Automation, PP.1242-1247, Avril 1988.
- [Uch 88] M. Uchiyama et P. Dauchez, "*A Symetric hybrid position / force control scheme for coordination of two robots*", Proceedings of IEEE International Conference on Robotics and Automation, Vol.1, PP.350-356, 1988.
- [Vib 87] C. Vibet, "ROBOT, *Principes et contrôle* », ed. Ellipses, Paris 1987.
- [Wei 97] Wei Li, "*Method for design of a hybrid neuro-Fuzzy control based on behavior modeling*", IEEE Transaction on Fuzzy systems, 1997.
- [Xav 92] X. Delebarre, "Commandes position et force de deux bras manipulateurs pour l'exploration planétaire ", Thèse de Doctorat université Montpellier II, 1992.
- [Yan 92] Y. E. Bouffard, "*Commande hybride et coordination de deux bras manipulateurs en environnement non structuré* », Thèse de Doctorat Université Montpellier II 1992.

- [Yos 90] T. Yoshikawa, "*Modeling and control of a three degree of freedom manipulator with two flexible links*", Proceeding of the 29th Conference on decision and control. pp. 2532-2537, Decembre1990
- [Zhe 86] Y. F. Zheng and J.Y.Luh, "*Joint torque for control of two coordinated robots*" IEEE International Conference on Robotics and Automation, PP.1375-1380, Avril 1986.
- [Zhe 89] Y. F. Zheng, "*Kinematics and Dynamics of two Industrial robots in assembly*", Proceedings of IEEE International Conference on Robotics and Automation, Vol.3, PP.1360-1365, 1989.