

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



**Université Hadj Lakhdar Batna**  
**Faculté de Technologie**  
**Département d'Électronique**



## **Mémoire**

**Présenté pour l'obtention du diplôme de  
MAGISTER en Électronique**

### **OPTION**

Contrôle Industriel

**Par**

**BOUDISSA Ahmed**

**Thème**

---

## **Commande Non Linéaire en Impédance d'un Robot Manipulateur Industriel**

---

Soutenu devant le jury composé de :

Dr. <b>KHIREDDINE Mohamed Salah</b>	Prof.	Université <b>Batna</b>	Président
Dr. <b>ABDESSEMED Yassine</b>	M.C.A.	Université <b>Batna</b>	Rapporteur
Dr. <b>MESSAI Abderraouf</b>	Prof.	Université <b>Constantine</b>	Examineur
Dr. <b>ATHAMENA Noureddine</b>	M.C.A.	Université <b>Batna</b>	Examineur
Dr. <b>SLIMANE Noureddine</b>	Prof.	Université <b>Batna</b>	Examineur

## *DEDICACE*

*A ma mère, à ma mère, à ma mère et à mon défunt père  
A mon épouse et à mes fils Oussama Abdelmouiz, Anes et ma  
fille Soundouss Alaa.*

*A toute ma famille.*

*A tous mes collègues d'étude et de travail.*

*A mes amis et les habitants de ma ville Ain Elmelh.*

*Aux peuples frères, Algérien et Palestinien.*

# *REMERCIEMENTS*

*Tout d'abord je remercie Monsieur Y.ABDESSAMED pour l'encadrement qu'il m'a assuré tout au long de ce projet.*

*J'adresse mes remerciements à Mr.F.FRGUENE Professeur à L'USTHB*

*Je remercie les membres du jury, qui m'ont fait l'honneur de participer au jugement de ce travail.*

*Je remercie mes collègues d'étude NEZAR.F, BENAZIZA.W, GHOUMRI.S et RABIE.B ainsi Mr.MOUFID.B*

*Enfin, Je remercie toute personne ayant contribué de près ou de loin à l'accomplissement de ce travail.*

# SOMMAIRE

Introduction Générale.....	1
<b>Chapitre I      Modélisation du Robot Manipulateur</b>	
1.1 Introduction :.....	4
1.2 Les origines des robots industriels : .....	5
1.3 Définitions générales :.....	5
1.4 Classification des robots séries : .....	8
1.4.1 Structure et morphologie des robots:.....	8
1.4.2 La géométrie du robot : .....	9
1.4.3 Domaine d'application :.....	9
1.5 Modèle Géométrique :.....	10
1.5.1 Coordonnées opérationnelles : .....	10
1.5.2 Détermination du repère (lié au corps) : .....	10
1.5.3 Passage du repère au repère, détermination des paramètres de Denavit-Hartenberg modifié .....	11
1.5.4 Modèle géométrique direct :.....	13
1.5.5 Modèle géométrique inverse : .....	13
1.5.5.1 Méthode de Paul :.....	13
1.5.5.2 L'approche de calcul numérique : .....	14
1.5.5.3 L'approche géométrique : .....	14
1.5.6 Obtention du modèle géométrique direct et inverse du bras manipulateur de 2 DLL ..	15
1.5.6.1 Modèle géométrique direct :.....	15
1.5.6.2 Modèle géométrique inverse par la méthode géométrique :.....	16
1.6 Modèle cinématique directe et inverse :.....	18
1.6.1 Modèle cinématique directe : .....	18
1.6.2 Modèle cinématique inverse :.....	20
1.6.3 Obtention du modèle cinématique du bras manipulateur de 2 DLL : .....	20
1.6.4 Jacobien de la force :.....	21
1.7 Le modèle dynamique :.....	22
1.7.1 Le calcul des Forces d'inertie :.....	23
1.7.2 Le calcul de la matrice $Mq$ :.....	24
1.7.3 Les symboles de Christoffel :.....	26
1.7.4 Le calcul de la gravité : .....	26
1.7.5 La forme explicite du model dynamique :.....	27

1.7.6	Obtention du modèle dynamique du bras manipulateur de 2 DLL : .....	27
1.7.6.1	Le calcul de $Jv1$ et $Jv2$ : .....	28
1.7.6.2	Le calcul de $\omega1$ et $\omega2$ : .....	28
1.7.6.3	Le calcul des symboles de Christoffel : .....	29
1.7.6.4	Le calcul du vecteur de gravité : .....	29
1.8	Conclusion : .....	30

## **Chapitre II Commande en Impédance**

2.1	Introduction : .....	32
2.2	Repère de complianc (de contrainte) : .....	33
2.3	Commande en force directe : .....	34
2.3.1	Commande Hybride position/force (Structures parallèle) : .....	35
2.3.1.1	Commande hybride force/position avec somme des couples articulaires : .....	36
2.3.1.2	Commande avec somme des déplacements ou vitesses : .....	37
2.3.1.3	Commande hybride force/position avec somme de forces cartésiennes : .....	38
2.3.2	Commande Hybride Externe (Structures série) : .....	38
2.3.2.1	Commande Hybride Externe avec Correction Articulare de Position : .....	39
2.3.2.2	Commande Hybride Externe avec Correction Cartésienne de Position : .....	40
2.4	Commande en force indirecte : .....	40
2.4.1	Commande passive : .....	40
2.4.2	Commande par raideur active : .....	40
2.4.3	Commande par Raideur avec retour de force : .....	42
2.4.4	Commande par Matrice d'Amortissement : .....	43
2.4.5	Commande en impédance : .....	44
2.4.5.1	Principe : .....	44
2.4.5.2	Commande en Impédance sans retour de force : .....	47
2.4.5.3	Commande en Impédance avec retour de force : .....	48
2.4.5.4	Commande en Impédance Généralisée : .....	52
2.5	Conclusion: .....	54

## **Chapitre III Commande non Linéaire**

3.1	Introduction : .....	56
3.2	Commande par découplage non linéaire .....	57
3.2.1	Principe de la Commande par découplage non linéaire .....	57
3.2.1.1	Modèle certain .....	57
3.2.1.1.1	Trajectoire définie complètement ( $qd, \dot{q}d, \ddot{q}d$ ) .....	58
3.2.1.1.2	Trajectoire définie par la position ( $qd$ ) .....	58

3.2.1.2	Modèle incertain .....	59
3.3	La commande backstepping .....	60
3.3.1	La conception de la commande backstepping .....	60
3.3.2	La conception de la commande backstepping pour le manipulateur:.....	64
3.4	La commande par mode glissant :.....	66
3.4.1	Principe de fonctionnement :.....	67
3.4.2	Choix de la surface de glissement :.....	68
3.4.3	Condition de convergence .....	70
3.4.4	Calcul du temps d'attractivité : .....	71
3.4.5	La commande équivalente dans le sens de Filippov .....	72
3.4.6	La commande équivalente .....	73
3.4.7	Constitution de loi de commande .....	75
3.4.8	La robustesse .....	76
3.4.9	Le phénomène de réticence (Chattering) .....	76
3.4.10	Solutions pour la réticence .....	77
3.4.10.1	La couche limite (boundary layer) .....	77
3.4.10.2	Pseudo mode glissant (quasi sliding mode) .....	77
3.4.10.3	La fonction de saturation .....	78
3.4.10.4	Solution par des lois d'approche .....	79
3.4.10.5	Mode glissant asymptotique .....	79
3.4.10.6	Mode glissant d'ordre 2 .....	81
3.5	Conclusion :.....	81

**Chapitre IV            Chapitre-IV- Application des commandes non linéaires au robot manipulateur en espace contraint.**

4.1	Le scénario envisagé : .....	82
4.2	La commande hybride Force/Position :.....	85
4.3	La Commande en impédance :.....	88
4.3.1	La Commande Par raideur active :.....	88
4.3.2	La Commande Par matrice de raideur :.....	91
4.3.3	La Commande en impédance par découplage non linéaire :.....	95
4.3.4	La commande en impédance à mode glissant :.....	100
4.3.5	La commande en impédance par Backstepping:.....	105
4.4	Conclusion :.....	108
	Conclusion Générale.....	109

# Table des Figures

Figure 1.1. Le porteur et le poignet du robot.....	6
Figure 1.2. Schéma illustrant l'espace articulaire et cartésien. ....	7
Figure 1.3. Différentes Structures du robot série .....	9
Figure 1.4. Robot à structure ouverte simple .....	11
Figure 1.5. Représentation des Paramètres géométriques de Denavit-Hartenberg modifié. ....	12
Figure 1.6. Manipulateur plan de 2DDL .....	15
Figure 1.7. Les deux solutions possibles du MGI. (a : le coude en haut (elbow up) ; b : le coude en bas (elbow down)) .....	17
Figure 1.8. Représentation géométrique du bras manipulateur plan de 2 DDL .....	17
Figure 1.9. Représentation des vitesses au point terminal du robot. ....	19
Figure 1.10. Manipulateur en interaction avec l'environnement.....	22
Figure 1.11. Schéma synoptique d'un manipulateur série de n DDL.....	24
Figure 1.12. Schéma synoptique de la gravité.....	27
Figure 1.13. Schéma du bras manipulateur avec les paramètres dynamiques.....	28
Figure 2.1. Principe de la commande hybride position / force.....	36
Figure 2.2. Commande hybride force/position avec somme des couples.....	37
Figure 2.3. Commande hybride force/position avec somme des déplacements cartésiens. ....	37
Figure 2.4. Commande hybride force/position avec somme des forces. ....	38
Figure 2.5. Schéma de principe de la commande hybride externe. ....	39
Figure 2.6. Commande hybride externe avec correction de la position articulaire. ....	40
Figure 2.7. Commande hybride externe avec correction de la position cartésienne. ....	40
Figure 2.8. Commande par raideur active. ....	42
Figure 2.9. Schéma de principe de la commande en raideur avec retour de la force. ....	43
Figure 2.10. Schéma de principe de la commande en amortissement avec retour de la force. ....	44
Figure 2.11. Schéma de Principe de la Commande d'impédance sans retour de force.....	46
Figure 2.12. Schéma de Principe de la Commande d'impédance avec retour de force. ....	47
Figure 2.13. Schéma de la Commande d'impédance sans retour de force.....	48
Figure 2.15. Schéma de la Commande en impédance par découplage non linéaire avec retour de force .....	51
Figure 3.1. Le principe de la commande par découplage non linéaire .....	59
Figure 3.2. Schématisation du mode de glissement dans le plan de phase.....	68
Figure 3.3. L'attraction de la surface de glissement.....	72
Figure 3.4. Illustration de la méthode de Filippove.....	73
Figure 3.5. Le phénomène de réticence.....	77
Figure 3.6. La fonction Cont(.).....	78
Figure 3.7. La fonction Sat(.) .....	78
Figure 4.1. Scénario d'interaction du robot avec son environnement. ....	82
Figure 4.2. Block Simulink des trajectoires désirées.....	84
Figure 4. 3. Les trajectoires désirées dans l'espace cartésien. ....	84
Figure 4.4. Commande hybride force/position par découplage non linéaire avec somme des couples articulaires. ....	86
Figure 4.5. Poursuite de la trajectoire suivant l'axe x. ....	87
Figure 4.6. Poursuite de la trajectoire suivant l'axe y .....	87
Figure 4.7. Les forces exercées sur l'environnement. ....	88
Figure 4.8. Couples actionneurs fournis.....	88

Figure 4.9. Poursuite de la trajectoire suivant l'axe x .....	89
Figure 4.10. Poursuite de la trajectoire suivant l'axe y .....	89
Figure 4.11. Les forces exercées sur l'environnement. ....	90
Figure 4.12. Couples actionneurs fournis.....	90
Figure 4.13. Poursuite de la trajectoire suivant l'axe x .....	92
Figure 4.14. Poursuite de la trajectoire suivant l'axe y .....	92
Figure 4.15. Les forces exercées sur l'environnement. ....	92
Figure 4.16. Couples actionneurs fournis.....	92
Figure 4.17. Principe de la commande par matrice de raideur en espace articulaire.....	93
Figure 4.18. Poursuite de la trajectoire suivant l'axe x .....	94
Figure 4.19. Poursuite de la trajectoire suivant l'axe y .....	94
Figure 4.20. Les forces exercées sur l'environnement. ....	94
Figure 4.21. Couples actionneurs fournis.....	94
Figure 4.22. Principe de la commande en impédance par découplage basée sur la position. ....	96
Figure 4.23. Poursuite de la trajectoire selon l'axe x. ....	97
Figure 4.24. Poursuite de la trajectoire selon l'axe y. ....	97
Figure 4.25. Les forces exercées sur l'environnement. ....	97
Figure 4.26. Couples actionneurs fournis.....	97
Figure 4.27. Poursuite de la trajectoire selon l'axe x. ....	99
Figure 4.28. Poursuite de la trajectoire selon l'axe y. ....	99
Figure 4.29. Couples actionneurs fournis.....	99
Figure 4.30. Les forces exercées sur l'environnement. ....	99
Figure 4.31. Poursuite de la trajectoire suivant l'axe x. ....	102
Figure 4.32. Poursuite de la trajectoire suivant l'axe y .....	102
Figure 4.33. Les forces exercées sur l'environnement. ....	102
Figure 4.34. Couples actionneurs fournis.....	102
Figure 4.35. Poursuite de la trajectoire suivant l'axe x .....	104
Figure 4.36. Poursuite de la trajectoire suivant l'axe y .....	104
Figure 4.37. Les forces exercées sur l'environnement. ....	104
Figure 4.38. Couples actionneurs fournis.....	104
Figure 4.39. Principe de la commande en impédance par Backstepping. ....	106
Figure 4.40. Poursuite de la trajectoire suivant l'axe x .....	107
Figure 4.41. Poursuite de la trajectoire suivant l'axe y .....	107
Figure 4.42. Les forces exercées sur l'environnement. ....	107
Figure 4.43. Couples actionneurs fournis.....	107

## ANNEXE A

Figure A.1. Trajectoire 2-glissant.	
Figure A.2 Convergence de l'algorithme Twisting dans le plan (s, s ).	
Figure A.3. La trajectoire de la commande super-Twisting dans le plan (s, s ).	
Figure A.4. Principe de l'algorithme sous-optimal.	
Figure A.5. Principe de l'algorithme prescribed convergence law.	

## Liste des Tableaux

<b>Tableau.1.1</b> Paramètres D-H du manipulateur plan de 2DDL.....	15
<b>Tableau.1.2.</b> Représente les articulations avec leur contribution au point terminal du robot..	19



# *Nomenclature*

## *Symboles :*

$q$  : Vecteur de positions articulaires.

$M(q)$ : Matrice d'inertie du robot.

$B(q)$  : Matrice des couples de Coriolis.

$G(q)$  : Matrice des couples centrifuges.

$V(q, \dot{q})$  : Vecteur des couples de Coriolis et centrifuges.

$G(q)$  : vecteur des termes de la gravité.

$\tau$  : Vecteur des couples articulaires.

$F_e$  (force exercée par l'effecteur terminal du robot sur l'environnement)

$k_e$  : Raideur de l'environnement.

$A$  : Matrice d'inertie

$B$  : Matrice d'amortissement

$K$  : Matrice de raideur

$c_1, c_2$  : Paramètres de la commande Backstepping.

$x_e$  : Position de l'environnement.

$T_{i,j}$  : Transformation homogène du repère  $j$  vers le repère  $i$ .

$R_{i,j}$  : Matrice de rotation du repère  $j$  par rapport au repère  $i$ .

${}^iJ(q)$ : Matrice jacobienne du robot par rapport au repère  $i$ .

$P$  : Estimée du paramètre  $P$ .

$Z$  : Impédance de sortie du robot.

$x_r$  : La position référence

$W_p$  : La loi de commande en position

$\tau_f$  : La loi de commande en effort

$W$ : Entrée auxiliaire de commande.

$S$  : Matrice de sélection.

## *Sigles :*

DDL : degré de liberté.

D-H : Paramètres de DENAVIT HARTENBERG

PD : Proportionnelle et Dérivée.

MGD : Modèle Géométrique Direct.

MGI : Modèle Géométrique Inverse.

MCD : Modèle Cinématique Direct.

MCI : Modèle Cinématique Inverse.

LCP : Loi de Commande en Position.

LCPA : Loi de Commande en Position Articulaire.

LCF : Loi de Commande en effort (Force).

PPP : Prismatique, Prismatique, Prismatique

RPP : Rotoïde, Prismatique, Prismatique

RRP : Rotoïde, Rotoïde, Prismatique

RRR : Rotoïde, Rotoïde, Rotoïde

## **Introduction générale :**

Lorsque le robot réalise une tâche en espace libre (non contraint), la commande en position dans cette situation est suffisante. Elle permet au manipulateur de suivre une trajectoire désirée dans l'espace articulaire ou cartésien, tout en appliquant les techniques de commande décentralisées PID qui permettent de contrôler chaque articulation à part, ou les techniques non linéaires qui traitent le système dans sa globalité, à l'instar de la commande par découplage non linéaire, mode de glissement, backstepping et l'intelligence artificielle... etc. Ces méthodes de commande robustes sont appliquées notamment aux systèmes non linéaires incertains (Manipulateur), mais si le robot est appelé à interagir avec l'environnement le cas par exemple d'un assemblage des pièces ajustées, les tâches de préhension d'objets fragiles ou fortement déformable, l'ébarbage et l'ébavurage, polissage et le domaine médical...etc, dans ces situations là, si on applique la commande en position, le robot va suivre son chemin sans tenir compte de la force de réaction due au contact avec l'environnement, par conséquent les forces d'interaction augmentent rapidement suite aux couples des actionneurs qui sont commandés par une loi de commande qui vise uniquement à minimiser les erreurs de position, provoquant ainsi un risque au robot et/ou à l'environnement.

La commande en position peut être appliquée dans les espaces contraints à moins que le modèle cinématique et dynamique du robot manipulateur soient précis, plus d'une connaissance parfaite de l'environnement (la géométrie et les caractéristiques mécaniques) [20] [27], mais pratiquement cette exactitude ne peut être réalisée qu'avec un coût colossal.

La solution de ce sérieux problème réside d'après N.Hogan dans le principe d'admittance/impédance c'est à dire tant que l'environnement se manifeste comme une admittance (accepte une force comme entrée et produit un déplacement), le robot doit se comporter comme étant une impédance (accepte un déplacement comme entrée et produit une force), sur la base de ce principe le robot doit avoir une caractéristique de compliance, donc au lieu de rejeter les forces générées par le contact, il s'adapte avec [8].

La commande adaptée donc dans l'espace contraint est bien la commande en effort, plusieurs types de cette commande sont présentés dans la littérature, Whitney en 1977 [36] abordât Les premiers travaux sur la commande en amortissement, Salisbury en 1980 [30] adopte la commande par raideur active, il considère l'outil terminal du manipulateur est muni des

ressorts virtuels dans toutes les directions avec des raideurs différentes suivant les contraintes imposées, Raibert et Craig en 1981 [28] développent la commande hybride force/position qui est largement utilisée, cependant cette dernière méthode utilise une matrice de sélection  $S$  qui décompose l'espace opérationnel en deux parties celles qui sont commandées en position et celles en force. Cette décomposition est basée sur la connaissance parfaite de l'environnement, la moindre erreur de modélisation provoque des mouvements indésirables dans les directions commandées en force, et des forces indésirables dans les directions commandées en position, ce problème s'amplifie durant la transition entre l'espace libre et contraint [39], en 1984, la commande en impédance a été introduite par Hogan [15].

Les deux approches principales pour le contrôle de l'interaction entre le manipulateur et son environnement sont :

- la commande hybride force /position (Hybrid Control Force-Position).
- la commande en Impédance.

Ces approches sont englobées dans un classement plus large, c'est la commande en force directe et indirecte, dans la littérature existe aussi le terme explicite ou implicite selon l'existence ou non d'une force désirée.

Donc la commande hybride traite la position et la force comme étant deux termes séparés, tandis que l'objectif de la commande en impédance est d'imposer dans chaque direction de l'espace opérationnel du manipulateur, une dynamique désirée entre la position de l'outil terminal du robot, et la force d'interaction avec son environnement [1].

Dans notre travail nous allons appliquer ces méthodes de commande, au robot manipulateur en contact avec l'environnement et voir son comportement sous les différentes commandes non linéaires.

Nous commençons dans le premier chapitre par la modélisation du robot manipulateur d'ordre  $n$  ; et l'obtention du modèle géométrique direct et inverse, le modèle cinématique direct et inverse, ainsi que le modèle dynamique d'un manipulateur de 2 DDL.

Le deuxième chapitre expose les aspects théoriques des différentes commandes en force et en impédance, la commande en impédance est traitée dans la partie dédiée à la commande en force indirecte, ainsi que la commande hybride dans la section consacrée à la commande en

force indirecte, ce choix permet de bien exposer la commande en impédance, les lois de commande et les illustrations sont présentées dans ce chapitre.

Dans le troisième chapitre on explique les différentes techniques de la commande non linéaire robuste, commençons par la commande par découplage non linéaires, son rôle principal est de rendre les équations dynamiques du robot, linéaires et découplées, ensuite la commande par mode de glissement qui permet de définir au préalable une surface de glissement  $S$  qui assure une convergence des erreurs (une dynamique désirée), la loi commande dans ce cas est conçue pour **assurer la convergence du système vers cette surface en temps fini et d'y rester**, donc le système devient **insensible aux perturbations**, c'est pourquoi cette commande est robuste vis-à-vis les incertitudes et les perturbations.

Le quatrième chapitre est consacré à l'application des différentes techniques de commande non linéaire avec les différentes approches de commande en impédance et en force au robot manipulateur, avant d'appliquer ces commande on modélise d'abord l'environnement, le modèle le plus répandue c'est une simple raideur  $Ke$ .

Une conclusion générale conclut ce travail, toute en présentant les avantages et les inconvénients des différentes commandes traitées dans cette étude, et les perspectives pour des futurs travaux de recherche.

# Chapitre-I-

## Modélisation du Robot Manipulateur

### 1.1 Introduction :

Le but de la modélisation des robots est de fournir des équations mathématiques qui permettent de simuler, concevoir ainsi pour commander et exécuter des tâches désirées, plusieurs types de modèle sont nécessaire pour décrire le comportement du robot tels que :

**Le modèle géométrique, Le modèle cinématique et Le modèle dynamique.**

- le modèle géométrique direct et inverse exprime la situation de l'organe terminal en fonction des variables articulaires du mécanisme et inversement;

- le modèles cinématique direct et inverse exprime la vitesse de l'organe terminal en fonction des vitesses articulaires et inversement;

– le modèles dynamique définissant les équations du mouvement du robot, qui permettent d'établir les relations entre les couples ou forces exercés par les actionneurs (selon le type des axes Rotoïde ou prismatique) et les positions, vitesses et accélérations des articulations.

Dans ce chapitre on va développer les différents types de model dans le cas général d'un manipulateur d'ordre  $n$  et dans le cas d'un bras manipulateur plan de 2 degré de liberté (DDL), ce type de robot est intéressant pour le développement d'une application de commande en impédance (interaction avec l'environnement).

### 1.2 Les origines des robots industriels : [17]

L'un des premiers robots industriels a été conçu par l'américain Willard L. G. Pollard Jr, qui a fait une demande de brevet pour son invention en 1934 [6], il s'agit d'un robot parallèle à deux DDL destiné à l'application de peinture sur la carrosserie d'une automobile. En 1941, DeVilbiss a fabriqué le premier robot industriel (un robot de Peinture série) sous la direction de Harold Roselund, en 1961 Unimation (entreprise Americaine developé par M. Joseph Engelberger) développe son premier prototype, l'Unimate.

En 1969, Victor Scheinman développe le Stanford Arm à l'Université Standford. C'est l'architecture utilisée aujourd'hui par presque tous les robots sériels à six DDL.

En 1968, Unimation vend une licence à Kawasaki Heavy Industries, qui développe le premier robot industriel au Japon en 1969, le Kawasaki-Unimate 2000,

En 1980, 19 000 robots industriels ont déjà été fabriqués au Japon par quelques 150 fabricants, dont Kawasaki, Yaskawa, Kitachi, Mitsubishi Heavy Industries, Fanuc et Nachi.

Le Japon devient ainsi le plus grand fabricant et utilisateur de robots industriels.

### **1.3 Définitions générales :**

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

#### **Le manipulateur :**

Mécanisme généralement composé d'éléments en série articulés ou coulissants l'un par rapport à l'autre, dont le but est la saisie et le déplacement d'objets suivant plusieurs degrés de liberté.

Il est multifonctionnel et peut-être commandé directement par un opérateur humain ou par tout système logique (système à cames, logique pneumatique, logique électrique câblée ou programmée), et le **Robot industriel** : "**Manipulateur automatique**, asservi en position, reprogrammable, polyvalent, capable de positionner et d'orienter des matériaux pièces, outils ou dispositif spécialisé au cours de mouvement variables et programmé pour l'exécution de tâches variées, il se présente souvent sous la forme d'un ou plusieurs bras se terminant par un poignet [3].

#### **Un robot série :**

Un robot série est composé d'une série de segments reliés par des articulations motorisées rotatives (en rotation) ou prismatiques (en translation). La grande majorité de robots industriels sont de type série [17].

#### **Un robot parallèle :**

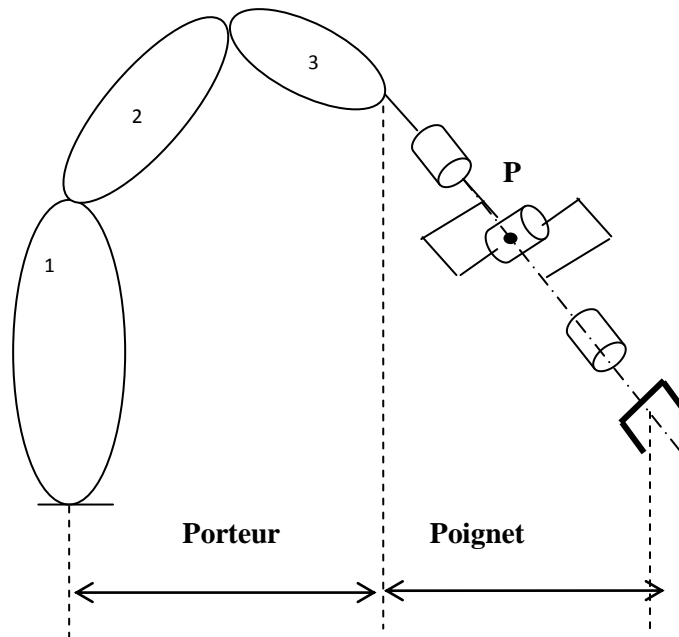
Un robot parallèle, l'effecteur est relié à la base via plusieurs "bras", et la plupart des articulations ne sont pas motorisées. Les robots parallèles peuvent aussi avoir six, cinq, quatre, trois ou même deux DDL. Les robots parallèles à six DDL les plus connus sont les hexapodes, comme ceux qui déplacent les cockpits des simulateurs de vol. Les robots parallèles sont généralement plus rigides et plus rapides que les robots séries. En revanche, ils sont beaucoup plus difficiles à étudier et il en existe de milliers d'architectures différentes [17].

**Porteur et le poignet :**

- **Le porteur** a pour rôle de fixer la position du point d'intersection, noté P, des axes des 3 dernières articulations (centre du poignet) ; cette position (P) ne dépend que de la configuration des solides (corps) 1, 2 et 3 (i.e., du porteur),

- **Le poignet** est destiné au positionnement de l'organe terminal (pince, outil).

Voir la figure suivante.



**Figure 1.1.** Le porteur et le poignet du robot

**Degré de liberté (DDL):**

Le nombre de degrés de liberté d'un manipulateur c'est le nombre des variables de position indépendants qui devront être précisées afin de localiser toutes les pièces du mécanisme. C'est un terme général utilisé pour n'importe quel mécanisme. Dans le cas de robots industriels typiques, le nombre des articulations est égal au nombre de degrés de liberté, car le manipulateur est habituellement une chaîne cinématique ouverte, et parce que chaque position d'articulation est généralement définie par une seule variable [11].

**Espace de travail :**

L'espace de travail représente l'ensemble des positions et orientations accessibles par le repère lié à l'organe terminal [5].



**L'espace articulaire** est l'espace qui a pour référence le repère lié à chaque articulation motorisée du robot. Les coordonnées associées sont appelées coordonnées articulaires.

**L'espace opérationnel** est l'espace qui a pour référence le repère lié à l'organe terminal du robot. Les coordonnées associées sont appelées coordonnées opérationnelles [2].

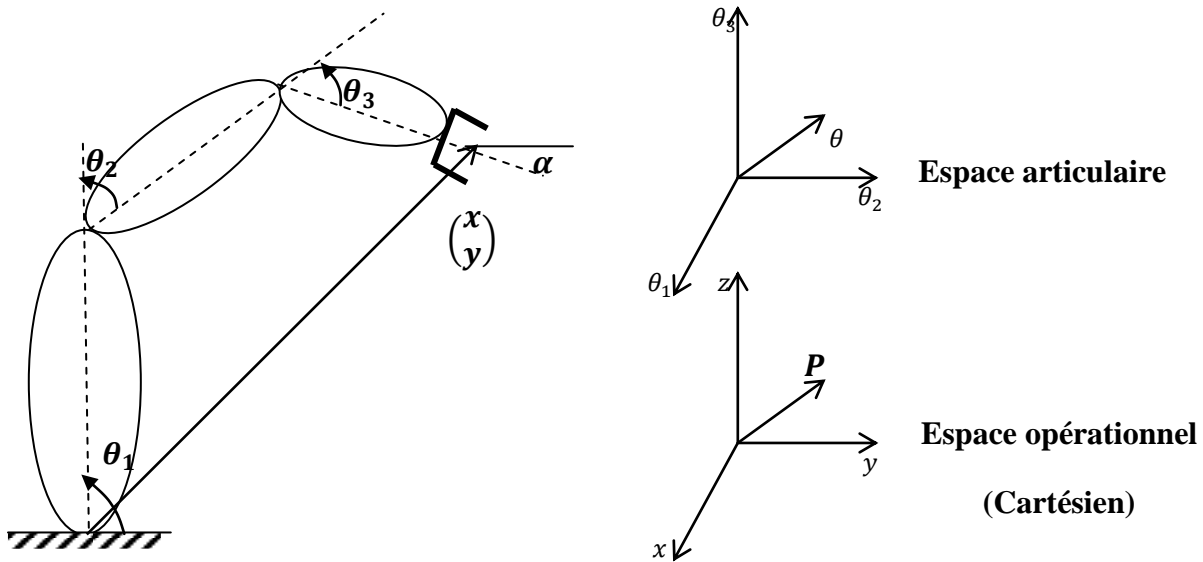


Figure 1.2. Schéma illustrant l'espace articulaire et cartésien.

**Répétabilité et précision :** [6]

### La répétabilité :

Ce paramètre caractérise la capacité que le robot a à retourner vers un point (position, orientation) donné. La répétabilité correspond à l'erreur maximum de positionnement sur un point prédéfini dans le cas de trajectoires répétitives. En général, la répétabilité est de l'ordre de 0,1 mm.

### La précision :

La précision représente l'erreur entre un point souhaité (réel) – défini par une position et une orientation dans l'espace cartésien – et le point atteint et calculé via le modèle géométrique inverse du robot. Cette erreur est due au modèle utilisé, à la quantification de la mesure de position, à la flexibilité du système mécanique. En général, l'erreur de positionnement absolu, également appelée précision, est de l'ordre de 1 mm [21].

### **Agilité :**

L'agilité est la propriété de se déplacer avec rapidité d'une configuration à une autre dans l'espace de travail. Elle est entièrement liée à la réponse dynamique du manipulateur [6], Le temps de transfert entre deux points est limité par :

- la vitesse maximum permise par chacun des degrés de liberté (les robots actuels atteignent des vitesses de l'ordre de **2 m/s** et des vitesses angulaires de **3 rad/s**),
- l'accélération maximum permise, elle-même limitée par les couples disponibles.
- le temps nécessaire à l'amortissement des oscillations aux extrémités du parcours.

Notons ici que les meilleurs manipulateurs modernes ont des accélérations maximum un ordre de grandeur inférieures à celles du bras humain.

### **Mobilité :**

La mobilité exprime la possibilité de pouvoir placer l'organe effecteur n'importe où dans l'espace de travail, avec une orientation arbitraire. Ceci nécessite un minimum de 6 degrés de liberté (DDL) [6].

### **La redondance :**

Un robot est redondant lorsque le nombre de degrés de liberté de l'organe terminal est inférieur au nombre de degrés de liberté de l'espace articulaire (nombre d'articulations motorisées). Cette propriété permet d'augmenter le volume du domaine accessible et de préserver les capacités de déplacement de l'organe terminal en présence d'obstacles.

### **La singularité :**

Pour tous les robots, qu'ils soient redondants ou non, il se peut que dans certaines configurations dites singulières, le nombre de degrés de liberté de l'organe terminal soit inférieur à la dimension de l'espace opérationnel.

La configuration singulière ou singularités se traduisent, physiquement, par la nullité du déterminant de la matrice Jacobéenne.

## **1.4 Classification des robots séries :**

### **1.4.1 Structure et morphologie des robots:**

La partie mécanique d'un robot est généralement constituée de deux sous ensembles distincts.

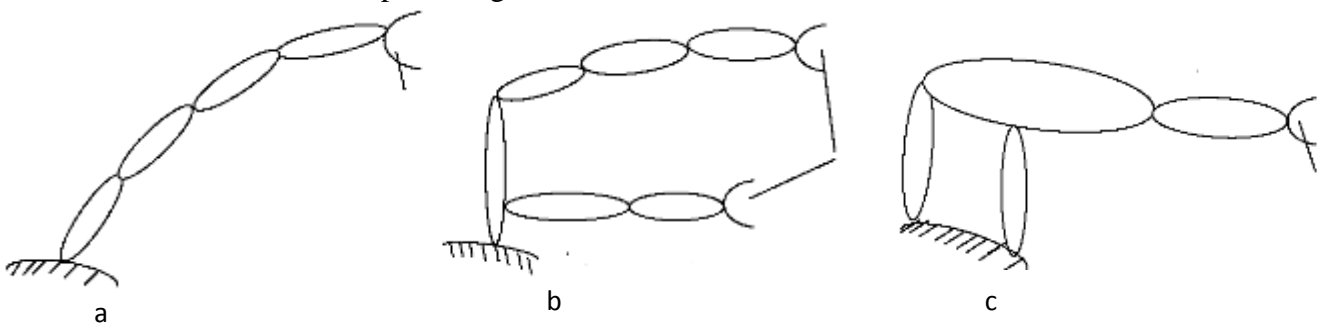
-Un (ou plusieurs) organe terminal considéré comme une interface permettant au robot d'interagir avec son environnement.

-Une structure mécanique articulée qui sert à amener l'organe terminal à une position et orientation désirées cette structure est composée du : bras, base, poignet coupleur et axe mécanique. Les robots peuvent être de:

-Structure à chaînes simples ouvertes figure 1.3a.

-Structure à chaînes arborescentes figure 1.3b.

-Structure à chaînes complexes figure 1.3c.



**Figure 1.3.** Différentes Structures du robot série

### 1.4.2 La géométrie du robot :

La plupart des manipulateurs aujourd'hui ont six ou inférieur degré de liberté, ces manipulateurs sont cinématiquement classés selon les trois premières articulations du bras (porteur) : [32]

- structure cartésienne (PPP).
- structure sphérique (ou polaire) (RRP).
- structure cylindrique (RPP)
- structure angulaire (RRR).
- structure SCARA.

### 1.4.3 Domaine d'application :

La plus grande aire d'application future de robots est dans l'assemblage. Donc, les robots sont souvent classés par application, des robots d'assemblage et non-assemblage [32].

Les Robots d'assemblage sont des petits robots, à entraînement électrique, soit rotoïde ou SCARA.

A ce jour, les principaux domaines d'application des robots de non-assemblage sont en soudage, pulvérisation, la peinture, de la manutention, et la machine de chargement et de déchargement.

## 1.5 Modèle Géométrique :

### 1.5.1 Coordonnées opérationnelles : [18]

Plusieurs possibilités existent pour la définition du vecteur  $X$  selon les méthodes utilisées pour spécifier la position et l'orientation.

En général, on définit la position d'un point du repère lié à l'organe terminal via :

- coordonnées cartésiennes (3 longueurs),
- coordonnées cylindriques (2 longueurs + 1 angle)
- coordonnées sphériques (1 longueur + 2 angles).

Il existe différentes possibilités pour définir l'orientation de l'organe terminal :

- $X - Y - Z$  fixe
- $Z - Y - X$  Les angles d'Euler,
- $Z - Y - Z$  Les angles d'Euler,
- D'autres types d'angles (Bryant, ...),
- Les cosinus directeurs. Leur utilisation revient à considérer l'ensemble des projections de vectrices unités portées sur les axes du repère, sur les axes du repère (repères orthonormés). Il en résulte paramètres, en effet :
  - 6 relations sont nécessaires pour indiquer que la base est orthonormée (3 pour indiquer des normes unitaires + 3 pour indiquer l'orthogonalité de la base),
  - Et 3 paramètres pour décrire l'orientation du repère.

### 1.5.2 Détermination du repère (lié au corps) : [21]

Une structure ouverte simple est composée de  $n + 1$  corps notés  $C_0, \dots, C_n$  et de  $n$  articulations. Le corps  $C_0$  désigne la base du robot et le corps  $C_n$  le corps qui porte l'organe terminal. L'articulation  $j$  connecte le corps  $C_j$  au corps  $C_{j-1}$  (figure 1.4).

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

– les corps sont supposés parfaitement rigides. Ils sont connectés par des articulations considérées comme idéales

(pas de jeu mécanique, pas d'élasticité), soit rotoïdes, soit prismatiques ;

– le repère  $R_j$  est lié au corps  $C_j$  ;

– l'axe  $z_j$  est porté par l'axe de l'articulation  $j$  ;

– l'axe  $x_j$  est porté par la perpendiculaire commune aux axes  $z_j$  et  $z_{j+1}$ . Si les axes  $z_j$  et  $z_{j+1}$  sont parallèles ou colinéaires, le choix de  $x_j$  n'est pas unique : des considérations de symétrie ou de simplicité permettent alors un choix rationnel.

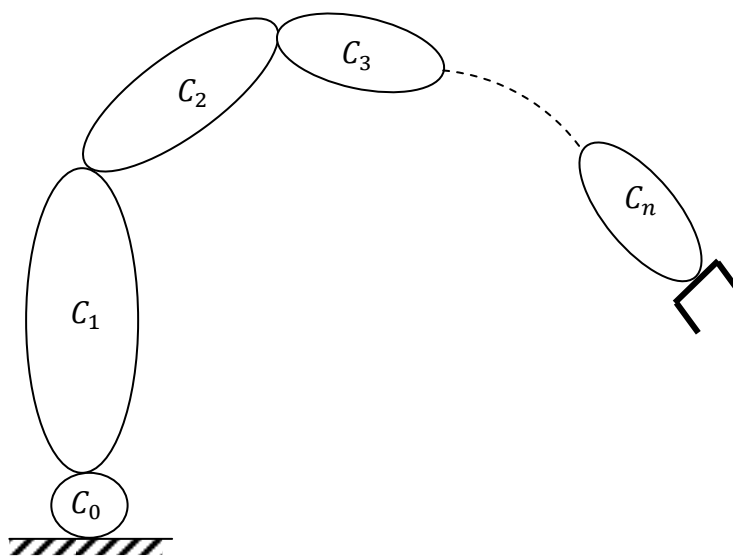


Figure 1.4. Robot à structure ouverte simple

### 1.5.3 Passage du repère au repère, détermination des paramètres de Denavit-Hartenberg modifié :

(elle est connue aussi par la notation de Khalil et Kleinfinger [21]):

Le passage du repère  $R_{j-1}$  au repère  $R_j$  s'exprime en fonction des quatre paramètres géométriques suivants (figure 1.5) :

- $\alpha_j$ : Angle entre les axes  $z_{j-1}$  et  $z_j$  correspondant à une rotation autour de  $x_{j-1}$  en utilisant la règle de la main droite. Il est important de se rappeler que  $\alpha_j$  est toujours une constante.

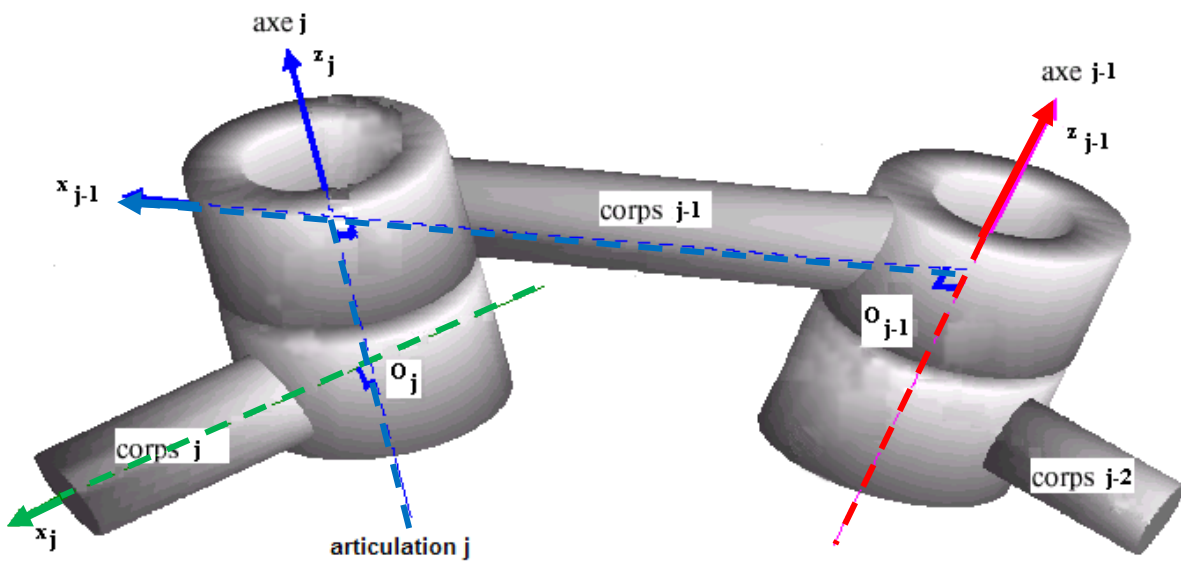
- $d_j$  : distance entre  $z_{j-1}$  et  $z_j$  le long de  $x_{j-1}$  ;
- $\theta_j$  : Angle entre les axes  $x_{j-1}$  et  $x_j$  correspondant à une rotation autour de  $z_j$  ; Il est important de comprendre que si l'articulation  $i$  est rotoïde,  $\theta_j$  est une variable articulaire, alors que si l'articulation  $i$  est prismatique,  $\theta_j$  est une constante.
- $r_j$  : Distance entre  $x_{j-1}$  et  $x_j$  le long de  $z_j$ , Il est important de se rappeler que  $r_j$  est toujours une constante.

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

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j d_j \quad (1.1)$$

Avec :

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



**Figure 1.5.** Représentation des Paramètres géométriques de Denavit-Hartenberg modifié.

De ces 4 changements de repères successifs, il résulte la matrice de transformation homogène suivante :

$$T_{j-1,j} = Rot(x, \alpha_j) \times Tran(x, d_j) \times Rot(z, \theta_j) \times Tran(x, r_j)$$

$$T_{j-1,j} = \begin{bmatrix} C\theta_j & -S\theta_j & 0 & d_j \\ C\alpha_j S\theta_j & C\alpha_j C\theta_j & -S\alpha_j & -r_j S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j & r_j C\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.2)$$

Où :  $S\alpha_j = \sin(\alpha_j)$ ,  $C\theta_j = \cos(\theta_j)$

$Rot(x, \alpha_j)$ ,  $Tran(x, d_j)$  Sont des matrices de transformation homogène (4x4) représentant respectivement une rotation  $\alpha_j$  autour de l'axe  $x$  et une translation  $d_j$  le long de  $x$ .

#### 1.5.4 Modèle géométrique direct :

Le modèle géométrique direct (MGD) est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal, en fonction de ses coordonnées articulaires. Dans le cas du robot série, il peut être représenté par la matrice de transformation  $T_{0,n}$  (du repère  $n$  vers le repère  $0$ ) :

$$T_{0,n} = T_{0,1}(q_1) \times T_{1,2}(q_2) \times \dots \times T_{n-1,n}(q_n) \quad (1.3)$$

Le modèle géométrique direct du robot est donné par la relation :

$$X = f(q) \quad (1.4)$$

$q = [q_1 \ q_2 \ \dots \ q_n]^T$ : étant le vecteur des variables articulaires.

$X = [x_1 \ x_2 \ \dots \ x_m]^T$ : Les coordonnées opérationnelles.

Le vecteur  $X$  comporte trois paramètres pour décrire la translation, et 9 pour la rotation dans le cas des cosinus directeurs.

$$X = [P_x \ P_y \ P_z \ s_x s_y s_z \ n_x n_y n_z \ a_x a_y a_z]^T \quad (1.5)$$

D'autres représentations sont couramment utilisées pour les rotations, comme les angles d'Euler, les angles de Roulis-Tangage-Lacet (RTL) ou les paramètres d'Euler (quaternions). On peut facilement passer des cosinus directeurs  $s$ ,  $n$ ,  $a$  à l'une quelconque de ces représentations et inversement [21].

#### 1.5.5 Modèle géométrique inverse :

Il s'agit de déterminer les coordonnées articulaires correspondant à une situation désirée pour l'organe terminal et spécifiée par les coordonnées opérationnelles  $X$ .

Il y a rarement unicité de solution au problème d'inversion du modèle géométrique direct, les solutions possibles constituent le modèle géométrique inverse.

Il existe un certain nombre de méthodes pour calculer le modèle géométrique inverse, notamment la méthode de **Paul** qui traite séparément chaque cas particulier et qui convient pour la plupart des robots industriels.

### 1.5.5.1 Méthode de Paul :

Dans le cas de robots à géométrie simple (pour lesquels la plupart des distances  $d_j$  et  $r_j$  sont nulles et les angles  $\theta_j$  et  $\alpha_j$  sont égaux  $0, \mp \pi/2$ ), le modèle géométrique inverse (M.G.I.) peut être obtenu analytiquement via la méthode de Paul.

Considérons le robot décrit par la matrice de transformation (1.3)

Le M.G.I. est obtenu en résolvant l'équation matricielle suivante :

$$U_0 = T_{0,1}(q_1) \times T_{1,2}(q_2) \times \dots \times T_{n-1,n}(q_n) \quad (1.6)$$

La méthode de Paul permet la détermination de  $q_1$ , puis  $q_2$  et ainsi de suite jusqu'à  $q_n$ . Il s'agit de déplacer l'une après l'autre chacune des variables articulaires ( $q_1 \dots q_n$ ) dans le membre de gauche de l'équation. Pour cela, on multiplie par  $T_{j,j-1}$  (en prenant successivement  $j = 1 \dots n$ ) de part et d'autre dans l'équation.

Considérons un robot à 6 D.D.L, en utilisant les équations suivantes on peut calculer les variables articulaires : [18]

$$\begin{aligned} U_0 &= T_{0,1}(q_1) \times T_{1,2}(q_2) \times T_{2,3}(q_3) \times T_{3,4}(q_4) \times T_{4,5}(q_5) \times T_{5,6}(q_6) \\ T_{1,0}(q_1) \times U_0 &= T_{1,2}(q_2) \times T_{2,3}(q_3) \times T_{3,4}(q_4) \times T_{4,5}(q_5) \times T_{5,6}(q_6) \\ T_{2,1}(q_2) \times U_1 &= T_{2,3}(q_3) \times T_{3,4}(q_4) \times T_{4,5}(q_5) \times T_{5,6}(q_6) \\ T_{3,2}(q_3) \times U_2 &= T_{3,4}(q_4) \times T_{4,5}(q_5) \times T_{5,6}(q_6) \\ T_{4,3}(q_4) \times U_3 &= T_{4,5}(q_5) \times T_{5,6}(q_6) \\ T_{5,4}(q_5) \times U_4 &= T_{5,6}(q_6) \end{aligned} \quad (1.7)$$

Avec  $U_j = T_{j,6} = T_{j,j-1} \times U_{j-1}$  pour  $j = 1,2,3,4$ .

La résolution de ces équations est intuitive, mais fait intervenir (en principe) quelques types d'équations dont la solution analytique est connue.



**1.5.5.2 L'approche de calcul numérique :**

Lorsque le modèle géométrique inverse n'existe pas, on peut calculer une solution particulière du problème inverse par des procédures numériques, qui est une solution locale au sens où elle dépend des conditions initiales.

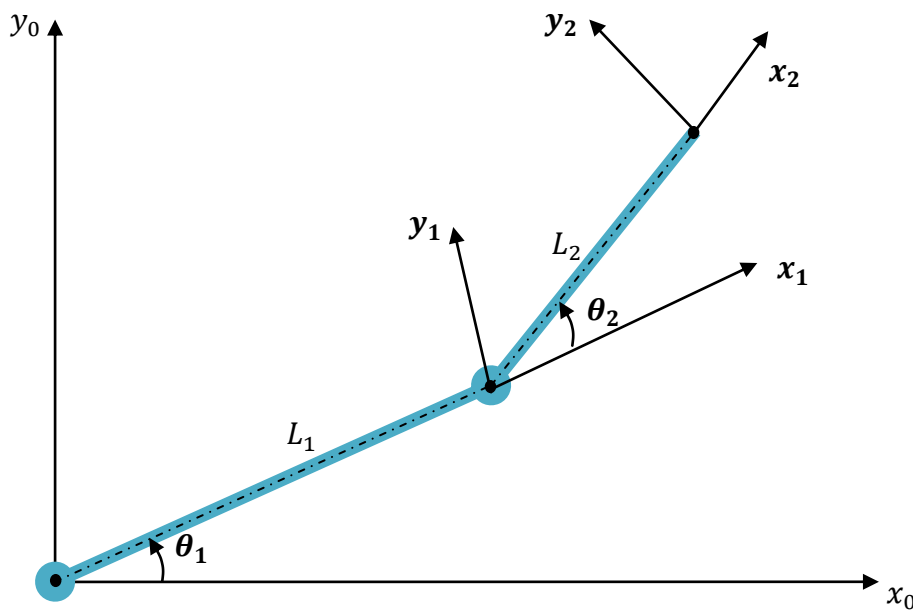
Notons que de telles méthodes peuvent être pénalisantes du point de vue du temps de calcul.

**1.5.5.3 L'approche géométrique :**

L'inexistence ou peu de méthodes analytiques générales permettant de résoudre le problème du modèle géométrique inverse d'un robot ayant une configuration arbitraire motive l'application des méthodes géométriques. Elle est utilisée généralement aux robots ayant des architectures relativement simples comme le cas des robots industriels actuels [32].

**1.5.6 Obtention du modèle géométrique direct et inverse du bras manipulateur de 2 DLL :**

**1.5.6.1 Modèle géométrique direct :**



**Figure 1.6.** Manipulateur plan de 2DDL

liaison	$L_i$	$\alpha_i$	$d_i$	$\theta_i$
1	$L_1$	0	0	$\theta_1$
2	$L_2$	0	0	$\theta_2$

Utilisant la formule générale de la matrice de transformation entre deux repères (1.2) on trouve :

$$T_{0,1} = \begin{bmatrix} c_1 & -s_1 & 0 & L_1 c_1 \\ s_1 & c_1 & 0 & L_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.8)$$

$$T_{1,2} = \begin{bmatrix} c_2 & -s_2 & 0 & L_2 c_2 \\ s_2 & c_2 & 0 & L_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.9)$$

$$T_{0,2} = T_{0,1} \times T_{1,2} = \begin{bmatrix} c_{12} & -s_{12} & 0 & L_1 c_1 + L_2 c_{12} \\ s_{12} & c_{12} & 0 & L_1 s_1 + L_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.10)$$

$T_{0,2}$  : La représentation du repère 2 par rapport le repère 0.

Les trois premiers composants de la dernière colonne du  $T_{0,2}$  représentent les coordonnées xyz de l'origine  $O_2$  par rapport au repère de base  $O_0$

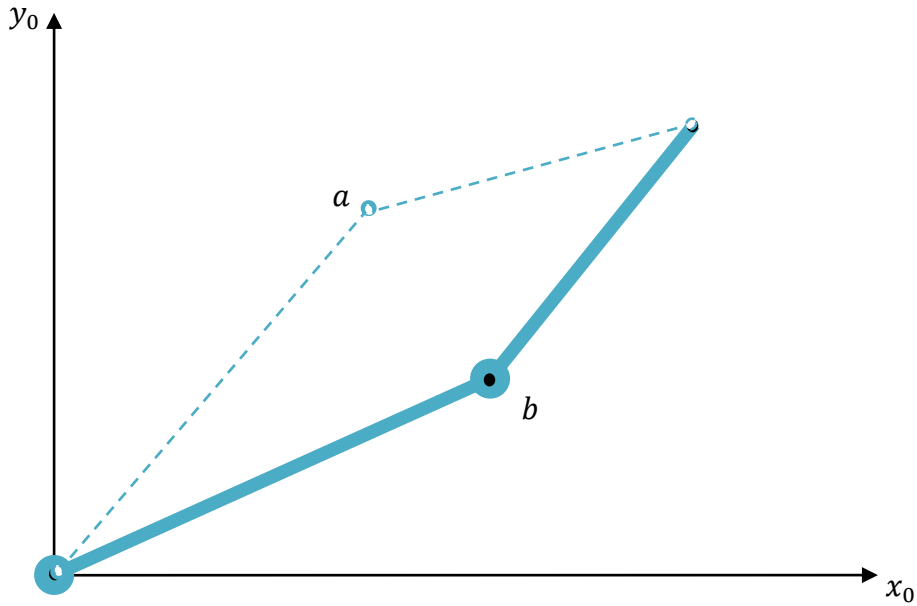
$$\begin{cases} x = L_1 c_1 + L_2 c_{12} \\ y = L_1 s_1 + L_2 s_{12} \\ z = 0 \end{cases} \quad (1.11)$$

Ces coordonnées de l'équation (1.11) Sont celles de l'outil terminal par rapport au repère de la base [32].

La matrice 3x3 (3 premières lignes / 3 premières colonnes) donne l'orientation de l'outil terminal par rapport à la base.

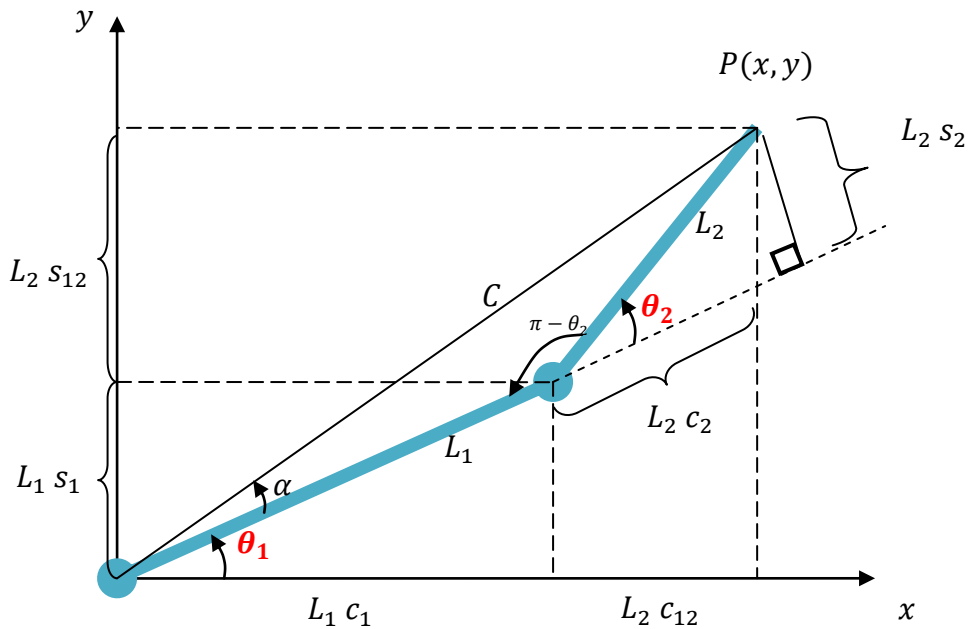
**1.5.6.2 Modèle géométrique inverse par la méthode géométrique :**

Dans cette situation nous connaissons les coordonnées  $P(x, y)$ , et nous allons chercher les variables articulaires  $\theta(\theta_1, \theta_2)$



**Figure 1.7.** Les deux solutions possibles du MGI.

(a : le coude en haut (elbow up) ; b : le coude en bas (elbow down))



**Figure 1.8.** Représentation géométrique du bras manipulateur plan de 2 DDL

On prend uniquement la solution ou le coude est bas (elbow-down)

$$C^2 = x^2 + y^2 \quad (1.12)$$

Appliquant le Théorème de Pythagore généralisé sur le triangle  $CL_1L_2$

$$\begin{aligned} C^2 &= L_1^2 + L_2^2 - 2L_1L_2 \cos(\pi - \theta_2) \\ C^2 &= L_1^2 + L_2^2 + 2L_1L_2 \cos(\theta_2) \end{aligned} \quad (1.13)$$

$$\theta_2 = \arccos\left(\frac{C^2 - L_1^2 - L_2^2}{2L_1L_2}\right) \quad (1.14)$$

Nous avons :

$$\begin{aligned} \tan(\theta_1 + \alpha) &= \frac{y}{x} \\ \tan(\alpha) &= \left(\frac{L_2 \sin(\theta_2)}{L_1 + L_2 \cos(\theta_2)}\right) \end{aligned}$$

Donc :

$$\theta_1 = \arctan\left(\frac{y}{x}\right) - \arctan\left(\frac{L_2 \sin(\theta_2)}{L_1 + L_2 \cos(\theta_2)}\right) \quad (1.15)$$

## 1.6 Modèle cinématique directe et inverse :

### 1.6.1 Modèle cinématique directe :

Le modèle cinématique directe du robot décrit les variations élémentaires des coordonnées opérationnelles en fonction des variations élémentaires des coordonnées articulaire.

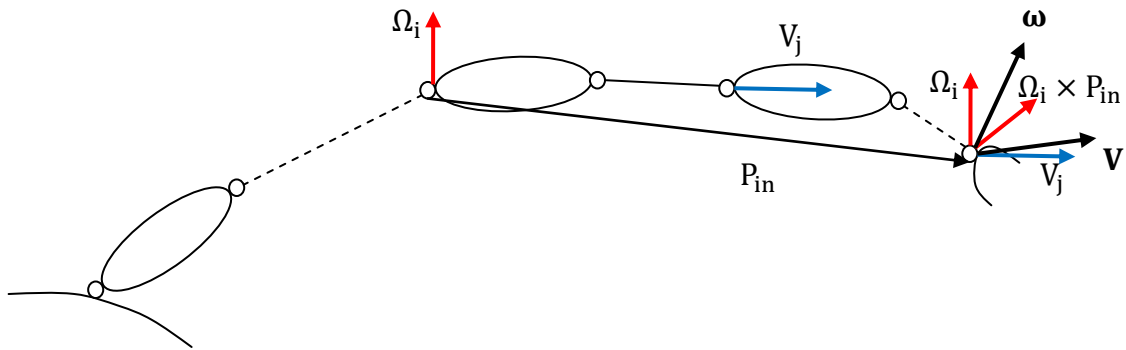
Il est noté :

$$dX = J(q)dq = \begin{pmatrix} \dot{P} \\ \omega \end{pmatrix} dq \quad (1.16)$$

$J(q)$  : désigne la matrice jacobéenne de dimensions (m x n) du mécanisme, elle est en fonction de la configuration articulaire  $q$ ,

$\dot{P}$  : Représentant la vitesse linière absolue de l'outil terminal par rapport à  $R_0$ .

$\omega$  : Représentant le vecteur de rotation absolue de l'outil terminal par rapport à  $R_0$ .



**Figure 1.9.** Représentation des vitesses au point terminal du robot.

**Tableau.1.2.** Représente les articulations avec leur contribution au point terminal du robot.

Articulation	Prismatique	rotoide
Vitesse linéaire	$V_j$	$\Omega_i \times P_{in}$
Vitesse angulaire	neant	$\Omega_i$

$\times$ : *produit scalaire*

La vitesse linéaire de l'outil terminal :

$$V = \sum_{i=1}^n [ \sigma_i Z_i + \bar{\sigma}_i (Z_i \times P_{in}) ] \dot{q}_i \tag{1.17}$$

La vitesse angulaire de l'outil terminal :

$$\omega = \sum_{i=1}^n (\bar{\sigma}_i Z_i) \dot{q}_i \tag{1.18}$$

Avec :

$$q_i = \bar{\sigma}_i \theta_i + \sigma_i d_i \quad , \quad \sigma_i = \begin{cases} 0 & \text{Rotoide} \\ 1 & \text{Prismatique} \end{cases}$$

Et :

$$\begin{aligned} V_i &= Z_i \dot{q}_i \\ \Omega_i &= Z_i \dot{q}_i \end{aligned} \tag{1.19}$$

Donc la forme générale de la matrice Jacobéenne exprimée par rapport le repère de la base est donnée par :

$${}^0J(q) = \begin{pmatrix} J_v \\ J_w \end{pmatrix} = \begin{bmatrix} \frac{\partial^0(P)}{\partial q_1} & \frac{\partial^0(P)}{\partial q_2} & \dots & \frac{\partial^0(P)}{\partial q_n} \\ \bar{\sigma}_1({}^0_1R Z) & \bar{\sigma}_2({}^0_2R Z) & \dots & \bar{\sigma}_n({}^0_nR Z) \end{bmatrix} \quad (1.20)$$

${}^0_iR$  : C'est une matrice de rotation exprime le repère  $R_i$  dans la base  $R_0$ .

$Z_i = (0,0,1)$ : Vecteur unitaire porté par l'axe  $Z_i$  de l'articulation  $q_i$ .

**Remarque :**

La relation de transformation de la matrice jacobienne entre repères suivante :

$${}^0J_n = \begin{bmatrix} {}^sA_i & 0_3 \\ 0_3 & {}^sA_i \end{bmatrix} {}^iJ_n \quad (1.21)$$

Où :  ${}^sA_i$  est la matrice de rotation de dimension (3x3), du repère  $R_i$  exprimé dans le repère  $R_s$ .

**1.6.2 Modèle cinématique inverse :**

L'objectif du modèle cinématique inverse est de trouver les vitesses  $\dot{q}$  des articulations pour satisfaire une vitesse des coordonnées opérationnelles  $\dot{X}$  imposée

$$\dot{q} = J^{-1}(q) \dot{X} \quad (1.22)$$

Pour obtenir le modèle cinématique inverse, on inverse le modèle cinématique direct en résolvant un système d'équations linéaires dont la solution est donnée par la relation suivante :

$$\begin{pmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{pmatrix} = J^{-1}(q) \begin{pmatrix} \dot{P} \\ \omega \end{pmatrix} \quad (1.23)$$

**1.6.3 Obtention du modèle cinématique du bras manipulateur de 2 DLL :**

A partir de l'équation (1.20)

$${}^0J = \begin{bmatrix} z_0 \times (O_2 - O_0) & z_1 \times (O_2 - O_1) \\ z_0 & z_1 \end{bmatrix} \quad (1.24)$$

$$O_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad O_1 = \begin{bmatrix} L_1 c_1 \\ L_1 s_1 \\ 0 \end{bmatrix}, \quad O_2 = \begin{bmatrix} L_1 c_1 + L_2 c_{12} \\ L_1 s_1 + L_2 s_{12} \\ 0 \end{bmatrix} \quad (1.25)$$

$$z_0 = z_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (1.26)$$

Donc :

$${}^0J = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} & -L_2 s_{12} \\ L_1 c_1 + L_2 c_{12} & L_2 c_{12} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \quad (1.27)$$

Les deux premières lignes de  $J$  indiquent la vitesse linéaire de l'origine  $O_2$  par rapport à la base. La troisième ligne c'est la vitesse linéaire liée à la direction de  $z_0$ , dans ce cas cette vitesse est nulle. Les dernières lignes représentent la vitesse angulaire du dernier repère (l'outil terminal) qui est simplement la rotation par rapport à l'axe verticale avec une vitesse  $\dot{\theta}_1 + \dot{\theta}_2$  [32].

**Remarque :**

Le fameux problème du **Modèle cinématique inverse est la Singularités.**

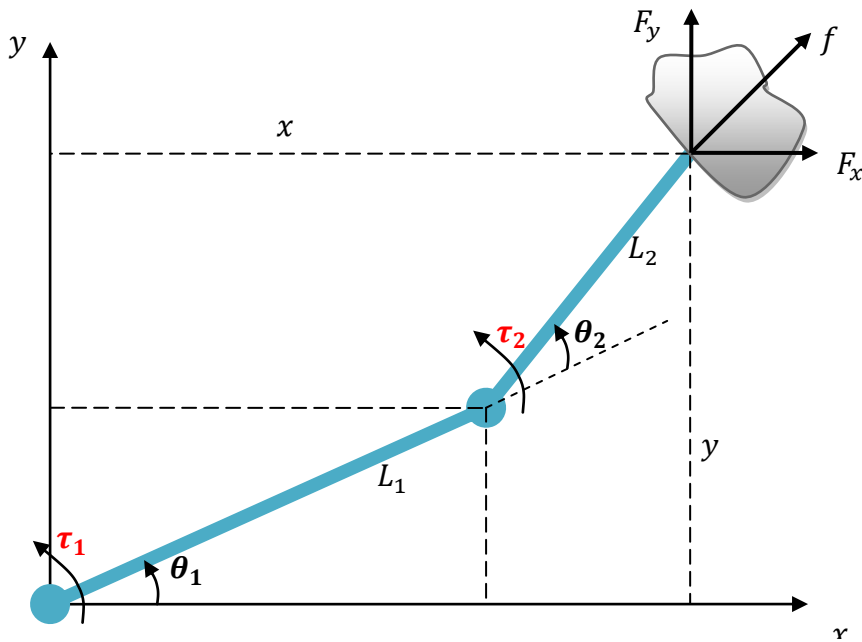
$$\det[J(q)] = 0 \quad (1.28)$$

$$\text{Avec : } J = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} & -L_2 s_{12} \\ L_1 c_1 + L_2 c_{12} & L_2 c_{12} \end{bmatrix}$$

$$\det[J(q)] = L_1 L_2 s_2 \quad (1.29)$$

La singularité est lorsque :  $\theta_2 = k \pi$ .

### 1.6.4 Jacobien de la force :



**Figure 1.10.** Manipulateur en interaction avec l'environnement.

Les torques produits par le contact sont :

$$\tau_1 = -F_x y + F_y x \quad (1.30)$$

Les coordonnées de  $x$  et  $y$  sont donnée par l'équation (1.11), insérées dans l'équation (1.30) donnent:

$$\tau_1 = -F_x (L_1 s_1 + L_2 s_{12}) + F_y (L_1 c_1 + L_2 c_{12}) \quad (1.31)$$

Le torque dans la deuxième articulation est calculé de même façon :

$$\tau_2 = -F_x L_2 s_{12} + F_y L_2 c_{12} \quad (1.32)$$

A partir de l'équation (1.31) et (1.32) on peut écrire la forme matricielle suivante :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} & L_1 c_1 + L_2 c_{12} \\ -L_2 s_{12} & L_2 c_{12} \end{bmatrix} \begin{bmatrix} F_x \\ F_y \end{bmatrix} \quad (1.33)$$

Cette dernière c'est la matrice jabiéenne transposée, donc on peut écrire :

$$\tau = J^T(q)f \quad (1.34)$$



$$\text{Avec : } \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, f = \begin{bmatrix} F_x \\ F_y \end{bmatrix}$$

Cette formule est très utile pour la modélisation des forces d'interaction du robot avec l'environnement.

## 1.7 Le modèle dynamique :

La cinématique décrit les mouvements du robot sans tenir compte des forces et des moments produisant ces mouvements, par contre la dynamique décrit explicitement la relation entre force et mouvement [32].

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

- le formalisme de Lagrange
- le formalisme de Newton-Euler

Dans cette section on va développer la dynamique du manipulateur par la méthode de Lagrange.

Equation de Lagrange :

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = \tau \quad (1.35)$$

Avec :

$$L = K - U \quad (1.36)$$

L : s'appelle le Lagrangien

K : L'énergie cinétique

U : L'énergie potentielle (elle est fonction de q  $U = U(q)$  )

Substituant L dans (1.35) :

$$\frac{d}{dt} \left( \frac{\partial K}{\partial \dot{q}} \right) - \frac{\partial K}{\partial q} + \frac{\partial U}{\partial q} = \tau \quad (1.37)$$

$$\frac{\partial U}{\partial q} = G \quad (1.38)$$

G: Vecteur de gravité

$\frac{d}{dt} \left( \frac{\partial K}{\partial \dot{q}} \right) - \frac{\partial K}{\partial q}$  : Forces d'inertie

### 1.7.1 Le calcul des Forces d'inertie :

$$K = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (1.39)$$

$$\frac{\partial K}{\partial \dot{q}} = \frac{\partial}{\partial \dot{q}} \left[ \frac{1}{2} \dot{q}^T M(q) \dot{q} \right] = M(q) \dot{q}$$

$$\frac{d}{dt} \left( \frac{\partial K}{\partial \dot{q}} \right) = \frac{d}{dt} [M(q) \dot{q}] = M(q) \ddot{q} + \dot{M}(q) \dot{q}$$

$$\frac{d}{dt} \left( \frac{\partial K}{\partial \dot{q}} \right) - \frac{\partial K}{\partial q} = M(q) \ddot{q} + \dot{M}(q) \dot{q} - \frac{1}{2} \begin{bmatrix} \dot{q}^T \frac{\partial M}{\partial q_1} \dot{q} \\ \vdots \\ \dot{q}^T \frac{\partial M}{\partial q_n} \dot{q} \end{bmatrix} = M(q) \ddot{q} + V(q, \dot{q}) \quad (1.40)$$

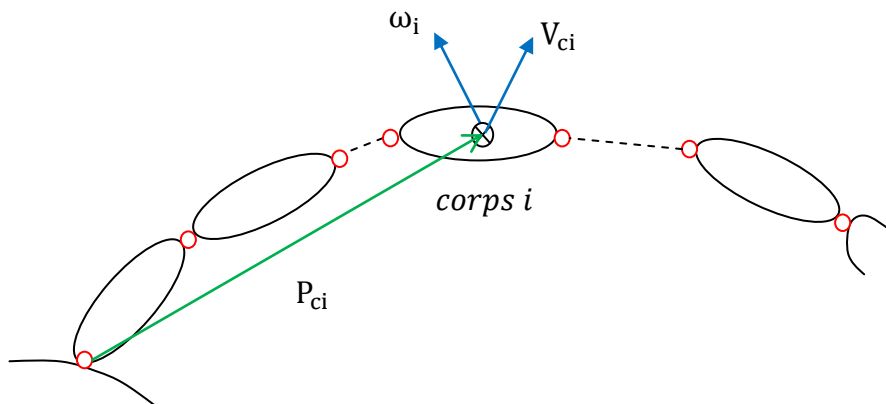
Donc on peut écrire la forme générale :

$$M(q) \ddot{q} + \dot{M}(q) \dot{q} - \frac{1}{2} \dot{q}^T \frac{\partial M}{\partial q} \dot{q} + G = \tau \quad (1.41)$$

$$M(q) \ddot{q} + V(q, \dot{q}) + G = \tau$$

### 1.7.2 Le calcul de la matrice $M(q)$ :

Calculons d'abord l'énergie cinétique globale du robot et on la reformule sous la forme de l'équation (1.39), pour extraire la matrice  $M(q)$ .



**Figure 1.11.** Schéma synoptique d'un manipulateur série de  $n$  DDL.

$$K_i = \frac{1}{2} (m_i v_{c_i}^T v_{c_i} + \omega_i^T I_i \omega_i) \quad (1.42)$$

L'énergie cinétique totale :

$$K = \sum K_i \quad (1.43)$$

Avec :

$$\begin{aligned} v_{c_i} &= J_{v_i} \dot{q} \\ \omega_i &= J_{\omega_i} \dot{q} \end{aligned} \quad (1.44)$$

$J_{v_i}$  : La matrice jacobéenne exprimée dans le référentiel  $i$ , elle décrit la vitesse linéaire  $v_i$  du corps  $i$  en fonction des  $q_0 \dots q_i$ , et par rapport à la base.

$J_{\omega_i}$  : La matrice jacobéenne exprimée dans le référentiel  $i$ , elle décrit la vitesse angulaire  $\omega_i$  du corps  $i$  en fonction des  $q_0 \dots q_i$ , et par rapport à la base.

Par la substitution de (1.44) dans la formule de (1.42) nous obtenons la relation suivante :

$$\begin{aligned} K_i &= \frac{1}{2} (m_i \dot{q}^T J_{v_i}^T J_{v_i} \dot{q} + \dot{q}^T J_{\omega_i}^T I_i J_{\omega_i} \dot{q}) \\ \frac{1}{2} \dot{q}^T \mathbf{M}(q) \dot{q} &= \frac{1}{2} \dot{q}^T \left[ \sum_{i=1}^n (m_i J_{v_i}^T J_{v_i} + J_{\omega_i}^T I_i J_{\omega_i}) \right] \dot{q} \end{aligned} \quad (1.45)$$

Donc :

$$\mathbf{M}(q) = \sum_{i=1}^n (m_i J_{v_i}^T J_{v_i} + J_{\omega_i}^T I_i J_{\omega_i}) \quad (1.46)$$

$I_{c_i}$  : L'inertie du corps  $i$ .

A partir du schéma 1.10 nous obtenons:

$$\begin{aligned} J_{v_i} &= \begin{bmatrix} \frac{\partial P_{c_i}}{\partial q_1} & \dots & \frac{\partial P_{c_i}}{\partial q_i} & 0 & \dots & 0 \end{bmatrix} \\ J_{\omega_i} &= [z_1 \quad \dots \quad z_i \quad 0 \quad \dots \quad 0] \end{aligned} \quad (1.47)$$

On pose  $M(q)$  de la forme suivante :

$$M := \begin{bmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nn} \end{bmatrix} \quad (1.48)$$

$:=$  indice indique ‘*par définition*’

A partir de (1.42) et (1.48) nous avons :

$$M(q)\ddot{q} + \begin{bmatrix} \dot{m}_{11} & \cdots & \dot{m}_{1n} \\ \vdots & \ddots & \vdots \\ \dot{m}_{n1} & \cdots & \dot{m}_{nn} \end{bmatrix} \dot{q} - \frac{1}{2} \begin{bmatrix} \dot{q}^T \begin{pmatrix} m_{111} & m_{121} & \cdots & m_{1n1} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n11} & m_{n21} & \cdots & m_{nn1} \end{pmatrix} \dot{q} \\ \vdots \\ \dot{q}^T \begin{pmatrix} m_{11n} & m_{12n} & \cdots & m_{1nn} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1n} & m_{n2n} & \cdots & m_{n nn} \end{pmatrix} \dot{q} \end{bmatrix} + G = \tau \quad (1.49)$$

$$m_{ijk} = \frac{\partial m_{ij}}{\partial q_k} \quad (1.50)$$

$$\dot{m}_{ij} = \frac{dm_{ij}}{dt} = \frac{dm_{ij}}{dq_k} \frac{dq_k}{dt} = m_{ijk} \dot{q}_k \quad (1.51)$$

### 1.7.3 Les symboles de Christoffel :

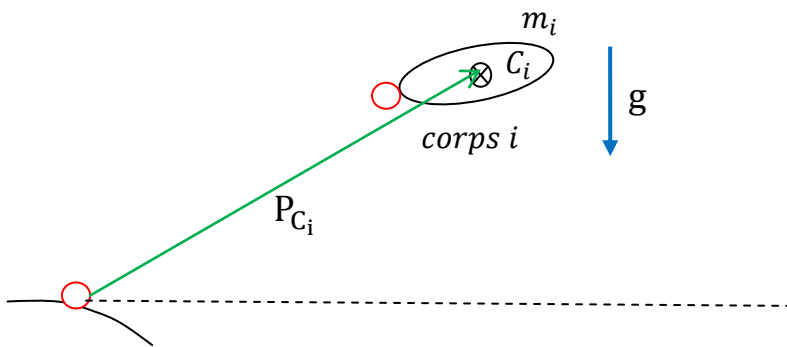
$$b_{ijk} = \frac{1}{2} (m_{ijk} + m_{ikj} - m_{jki}) \quad (1.52)$$

$$V = C(q)[\dot{q}^2] + B(q)[\dot{q}\dot{q}] \quad (1.53)$$

$$C(q)[\dot{q}^2] = \begin{bmatrix} b_{1,11} & b_{1,22} & \cdots & b_{1,nn} \\ b_{2,11} & b_{2,22} & \cdots & b_{2,nn} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n,11} & b_{n,22} & \cdots & b_{n,nn} \end{bmatrix} \begin{bmatrix} \dot{q}_1^2 \\ \dot{q}_2^2 \\ \vdots \\ \dot{q}_n^2 \end{bmatrix} \quad (1.54)$$

$$B(q)[\dot{q}\dot{q}] = \begin{bmatrix} 2b_{1,12} & 2b_{1,23} & \cdots & 2b_{1,(n-1)n} \\ 2b_{2,12} & 2b_{2,23} & \cdots & 2b_{2,(n-1)n} \\ \vdots & \vdots & \vdots & \vdots \\ 2b_{n,12} & 2b_{n,23} & \cdots & 2b_{n,(n-1)n} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \dot{q}_2 \\ \dot{q}_2 \dot{q}_3 \\ \vdots \\ \dot{q}_{(n-1)} \dot{q}_n \end{bmatrix} \quad (1.55)$$

### 1.7.4 Le calcul de la gravité :



**Figure 1.12.** Schéma synoptique de la gravité.

$$U_i = m_i(g^T P_{C_i}) \quad (1.56)$$

$g$  : Vecteur donne la direction de la gravité dans le référentiel de la base.

$P_{C_i}$  : Vecteur donne les coordonnées du centre de masse du corps  $i$ .

$$U = \sum_{i=1}^n U_i = \sum_{i=1}^n m_i(g^T P_{C_i}) \quad (1.57)$$

Le vecteur de gravité est donné par :

$$G = \frac{dU}{dq_j} = \sum_{i=1}^n m_i(g^T \frac{\partial P_{C_i}}{\partial q_j}) \quad (1.58)$$

$$G = \begin{bmatrix} J_{v_1}^T & J_{v_2}^T & \dots & J_{v_n}^T \end{bmatrix} \begin{bmatrix} m_1 g \\ m_2 g \\ \vdots \\ m_n g \end{bmatrix} \quad (1.59)$$

### 1.7.5 La forme explicite du model dynamique :

$$M(q)\ddot{q} + C(q)[\dot{q}^2] + B(q)[\dot{q}] + G = \tau \quad (1.60)$$

Avec :

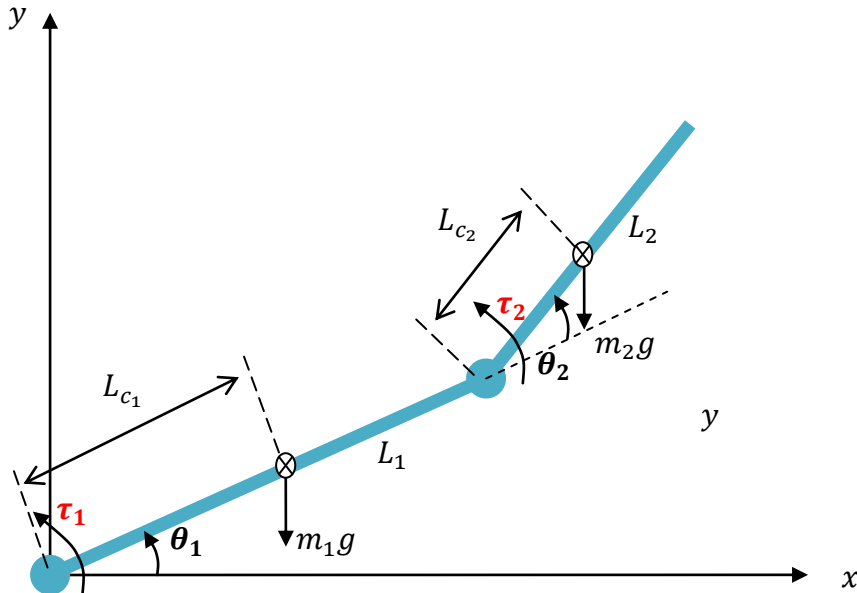
$M(q)$  : La matrice d'inertie (symétrique définie positive).

$C(q)$  : Matrice représente les forces centrifuges.

$B(q)$  : Matrice représente les forces de Coriolis.

$G$  : Le vecteur de gravité.

### 1.7.6 Obtention du modèle dynamique du bras manipulateur de 2 DLL :



**Figure 1.13.** Schéma du bras manipulateur avec les paramètres dynamiques.

#### 1.7.6.1 Le calcul de $J_{v_1}$ et $J_{v_2}$ :

Par l' utilisation du modèle développé dans la section précédente (1.47):

Dans ce cas là  $q = (\theta_1 \theta_2)^T$

$$J_{v_1} = \begin{bmatrix} L_{c_1} \sin(\theta_1) & 0 \\ L_{c_1} \cos(\theta_1) & 0 \\ 0 & 0 \end{bmatrix}$$

$$J_{v_2} = \begin{bmatrix} -L_1 \sin\theta_1 - L_{c_2} \sin(\theta_1 + \theta_2) & -L_{c_2} \sin(\theta_1 + \theta_2) \\ L_1 \cos\theta_1 + L_{c_2} \cos(\theta_1 + \theta_2) & L_{c_2} \cos(\theta_1 + \theta_2) \\ 0 & 0 \end{bmatrix}$$

#### 1.7.6.2 Le calcul de $\omega_1$ et $\omega_2$ :

$$\omega_1 = \dot{\theta}_1 k \quad \text{et} \quad \omega_2 = (\dot{\theta}_1 + \dot{\theta}_2) k$$

$$\text{Donc : } J_{\omega_i}^T I_i J_{\omega_i} = I_1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + I_2 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} I_1 + I_2 & I_2 \\ I_2 & I_2 \end{bmatrix}$$

Avec ces données et l'application de la formule (1.45)

On peut écrire la forme de matrice d'inertie M comme suite:

$$M = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}$$

$$\text{Tel que : } \begin{cases} m_{11} = m_1 L_{c_1}^2 + m_2 (L_1^2 + L_{c_2}^2 + 2L_1 L_{c_2} \cos \theta_2) + I_1 + I_2 \\ m_{12} = m_{21} = m_2 (L_{c_2}^2 + L_1 L_{c_2} \cos \theta_2) + I_2 \\ m_{22} = m_2 L_{c_2}^2 + I_2 \end{cases} \quad (1.61)$$

### 1.7.6.3 Le calcul des symboles de Christoffel :

$$\begin{cases} b_{111} = \frac{\partial m_{11}}{\partial \theta_1} = 0 \\ b_{121} = b_{112} = \frac{1}{2} \frac{\partial m_{11}}{\partial \theta_2} = -m_2 L_1 L_{c_2} \sin \theta_2 \\ b_{122} = \frac{\partial m_{12}}{\partial \theta_2} - \frac{1}{2} \frac{\partial m_{22}}{\partial \theta_1} = -m_2 L_1 L_{c_2} \sin \theta_2 \\ b_{211} = \frac{\partial m_{21}}{\partial \theta_1} - \frac{1}{2} \frac{\partial m_{11}}{\partial \theta_2} = m_2 L_1 L_{c_2} \sin \theta_2 \\ b_{221} = b_{212} = \frac{1}{2} \frac{\partial m_{22}}{\partial \theta_1} = 0 \\ b_{222} = \frac{1}{2} \frac{\partial m_{22}}{\partial \theta_2} = 0 \end{cases} \quad (1.62)$$

$$C(q)[\dot{q}^2] = \begin{bmatrix} 0 & -m_2 L_1 L_{c_2} \sin \theta_2 \\ m_2 L_1 L_{c_2} \sin \theta_2 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix} \quad (1.63)$$

$$B(q)[\dot{q}\dot{q}] = \begin{bmatrix} -2m_2 L_1 L_{c_2} \sin \theta_2 \\ 0 \end{bmatrix} [\dot{\theta}_1 \dot{\theta}_2] \quad (1.64)$$

$$V = C(q)[\dot{q}^2] + B(q)[\dot{q}\dot{q}] = \begin{bmatrix} -m_2 L_1 L_{c_2} \dot{\theta}_2^2 \sin \theta_2 - 2 m_2 L_1 L_{c_2} \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 \\ m_2 L_1 L_{c_2} \dot{\theta}_1^2 \sin \theta_2 \end{bmatrix} \quad (1.65)$$

Avec :  $q = (\theta_1 \theta_2)^T$

#### 1.7.6.4 Le calcul du vecteur de gravité :

D'après l'équation (1.57), on doit d'abord commencer par le calcul des  $P_{C_i}$ .

$$P_{C_1} = \begin{bmatrix} L_{c_1} \cos\theta_1 \\ L_{c_1} \sin\theta_1 \end{bmatrix}, \quad P_{C_2} = \begin{bmatrix} L_1 \cos\theta_1 + L_{c_2} \cos(\theta_1 + \theta_2) \\ L_1 \sin\theta_1 + L_{c_2} \sin(\theta_1 + \theta_2) \end{bmatrix} \quad (1.66)$$

$$g^T = [0 \quad g] \quad (1.67)$$

$g^T$  : Représenté dans le repère de la base.

$$U = U_1 + U_2 = m_1 [0 \quad g] \begin{bmatrix} L_{c_1} \cos\theta_1 \\ L_{c_1} \sin\theta_1 \end{bmatrix} + m_2 [0 \quad g] \begin{bmatrix} L_1 \cos\theta_1 + L_{c_2} \cos(\theta_1 + \theta_2) \\ L_1 \sin\theta_1 + L_{c_2} \sin(\theta_1 + \theta_2) \end{bmatrix}$$

$$U = m_1 L_{c_1} g \sin\theta_1 + m_2 L_1 g \sin\theta_1 + m_2 L_{c_2} g \sin(\theta_1 + \theta_2) \quad (1.68)$$

$$G_1 = \frac{\partial U}{\partial \theta_1} = m_1 L_{c_1} g \cos\theta_1 + m_2 L_1 g \cos\theta_1 + m_2 L_{c_2} g \cos(\theta_1 + \theta_2)$$

$$G_2 = \frac{\partial U}{\partial \theta_2} = m_2 L_{c_2} g \cos(\theta_1 + \theta_2)$$

$$G = \begin{bmatrix} m_1 L_{c_1} g \cos\theta_1 + m_2 L_1 g \cos\theta_1 + m_2 L_{c_2} g \cos(\theta_1 + \theta_2) \\ m_2 L_{c_2} g \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (1.69)$$

## 1.8 Conclusion :

Nous avons montré, dans ce chapitre, comment obtenir le modèle géométrique et cinématique directe et inverse ainsi que le modèle dynamique.

Le modèle géométrique donne une image du déplacement de l'effecteur sur le plan articulaire, on a vu comment obtenir ce modèle d'une manière systématique.

Le modèle géométrique inverse offre au contrôleur à chaque instant les angles des articulations à suivre par l'outil terminal du robot. Le calcul de ce modèle présente des difficultés tel que l'inexistence d'une méthode systématique d'inversion, plusieurs méthodes de calcul sont présentées comme celle de Paul ou la méthode de géométrie.

Ensuite on a abordé le problème de la cinématique, ce qui est traduit par le déplacement de l'organe terminal par rapport aux variations des articulations et inversement, la matrice



Le jacobien est la base de ce modèle, la cinématique inverse présente un problème de singularité qui correspond au déterminant nul de la matrice Jacobienne, les singularités doivent être évitées lors de la conception du contrôleur.

Par conséquent on a utilisé la matrice jacobienne pour calculer le jacobien des forces (jacobien transposé) qui relie les forces cartésiennes avec les couples.

Enfin, nous avons développé le modèle dynamique par le formalisme EULER-LAGRANGE .

# *Chapitre II*

## *Commande en Impédance*

### **2.1 Introduction :**

Si le robot est asservit à dessiner un cercle sur un tableau attaché au mur, l'outil terminal du robot manipulateur est perpendiculaire par rapport au tableau, dans ce cas là, la loi de commande en position réalise cette tâche avec efficacité. L'autre cas si le tableau est incliné de  $30^\circ$  par rapport au mur (ou l'outil terminal), ce robot et avec la même loi de commande interagissent avec le tableau durant le suivi de sa trajectoire (le cercle), générant ainsi une force plus au moins importante dépend de la vitesse du robot et/ou rigidité du tableau, ces forces provoquent un risque pour l'environnement et/ou robot.

Avec cet exemple l'insuffisance de la commande en position est claire en espace contraint, où la commande en effort s'avère intéressante, donc les lois de commande doivent être élaborées de façon qu'elles puissent bien gérer les forces produites du contact.

On ne peut jamais considérer les forces générées du contact comme source de perturbation, puisque ces forces d'interaction sont en relation directe avec les états du robot (vitesse...), alors que les perturbations sont traditionnellement indépendantes des états du robot [16].

La commande en impédance est abordée premièrement par HOGAN [15] c'est une généralisation de la commande par raideur développé par Whitney en 1977 [36] et la commande en amortissement, étudiée par Salisbury en 1980 [Sal 80], ce type de commande rentre dans la catégorie de la commande en force indirecte où la force est implicitement contrôlée.

L'autre catégorie c'est la commande en force directe où la force est explicitement contrôlée, plusieurs types de cette commande sont exposés dans la littérature, la plus intéressante c'est la commande hybride force/position de Raibert et Craig en 1981 [28].

Les applications de la commande en impédance ne cesse d'augmenter, tel que : l'assemblage l'ébavurage, polissage ainsi que le domaine médical et de la chirurgie...etc.

Pour bien aborder le problème de la commande en impédance, on va exposer dans ce chapitre les deux grandes familles de la commande en force commençons par la commande en force directe puis dans la deuxième section la commande en force indirecte et on met l'accent sur celle en impédance.

## 2.2 Repère de complianc (de contrainte) :

Les contraintes affectent généralement certaines directions de l'espace opérationnel sans en affecter les autres. Cette association entre l'espace opérationnel et les degrés de liberté du système est réalisée à l'aide d'un repère dit de complianc. Ce dernier peut être lié, selon la tâche à accomplir, à l'organe terminal, à l'environnement ou à l'objet manipulé.

La correction peut être effectuée dans le repère de la base ou dans le repère de complianc. Dans ce dernier cas, quelques transformations doivent intervenir pour en décrire les vecteurs de positions et vitesses cartésiennes. L'opérateur doit choisir intelligemment Les directions de repère des contraintes (complianc) afin de faciliter plus tard la mise en œuvre de la commande [12].

Les trajectoires désirées étant générées dans le repère de la base, quelques transformations sont à effectuer pour les décrire dans le repère de complianc.

Les positions cartésiennes actuelles et désirées dans le repère de complianc sont obtenues en utilisant la transformation suivante :

$${}^cX = \begin{bmatrix} {}^cR & 0_3 \\ 0_3 & {}^cR \end{bmatrix} {}^0X, \text{ S'il s'agit uniquement d'une rotation}$$

La matrice  ${}^cR = {}^cR^T$  est la matrice de rotation du repère de la base par rapport au repère de complianc.

- Le cas général :

$${}^cX = {}^0T {}^0X \tag{2.1}$$

${}^0T$  : Matrice de transformation du repère de la base par rapport au repère de complianc.

Le jacobien défini dans le repère de compliance est obtenu par la relation suivante:

$${}^c J = \begin{bmatrix} {}^c R & 0_3 \\ 0_3 & {}^c R \end{bmatrix} {}^0 J, \text{ S'il s'agit uniquement d'une rotation}$$

- Le cas général :

$${}^c J = {}^c \mathbf{T} {}^0 J \quad (2.2)$$

Les vitesses cartésiennes sont déterminées par la relation suivante:

$${}^c \dot{X} = {}^c J \dot{q} \quad (2.3)$$

Pour décrire (l'effort) dans le repère de la base on effectue une simple projection dans le repère de la base selon la relation suivante :

$${}^0 F = {}^0 \mathbf{T} {}^c F \quad (2.4)$$

La correction peut être effectuée dans le repère de la base ou dans le repère de compliance., quelques transformations doivent intervenir pour décrire les vecteurs de positions et vitesses cartésiennes dans le même repère, les résultats obtenus sont identiques néanmoins doit choisir un repère sur lequel on développe une loi de commande qui demande un nombre **minimal** des opérations de transformation entre repère.

### 2.3 Commande en force directe :

Chaque mission de manipulation peut être décomposé en tâches qui sont définies par une situation de contact particulière entre l'effecteur du manipulateur (ou l'outil terminal) et l'environnement. On peut associer pour chaque tache, un ensemble de contraintes, appelée les contraintes naturelles.

Par exemple, la main en contact d'une surface fixe et rigide elle n'est pas libre de s'y déplacer en sens perpendiculaire; par conséquent, une contrainte de position naturelle existe.

Si la surface présente des frottements, la main n'est pas libre d'y appliquer des forces arbitraires tangentes; donc, une contrainte de force naturelle existe.

Ces deux types de contrainte, de la force et de la position, partagent les degrés de liberté de mouvements possibles de l'effecteur en deux ensembles orthogonaux qui doivent être commandées selon des critères différents [11].

La Commande en force directe, c'est une commande avec consigne en force, elle contrôle les degrés contraintes en position par une Loi de Commande en Force (LCF), ainsi les directions contraintes en force par une Loi de Commande en Position (LCP), dans la littérature existent des Structures parallèles et Séries

### **2.3.1 Commande Hybride position/force (Structures parallèle) :**

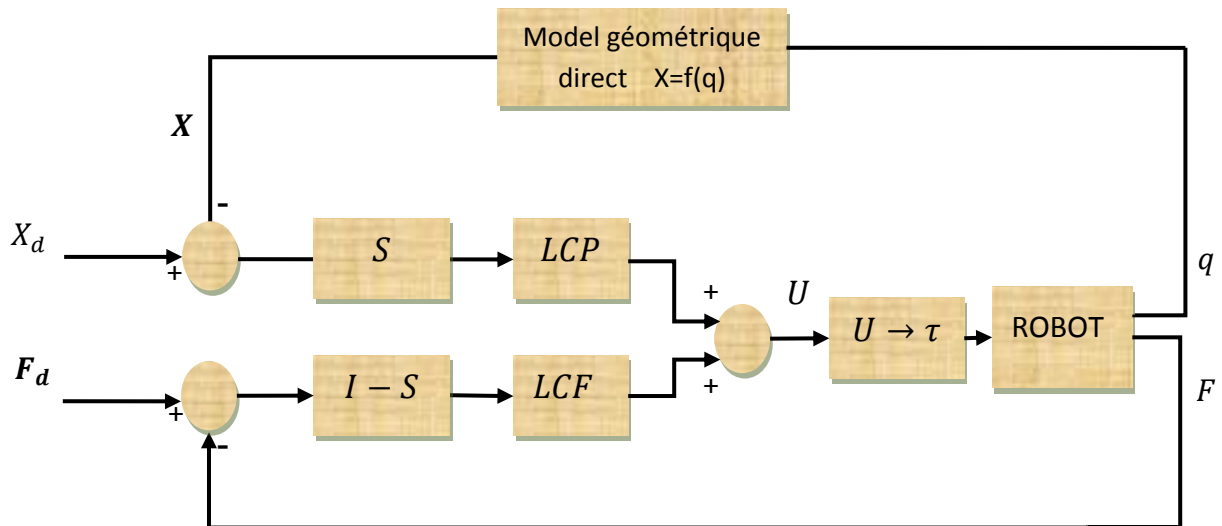
Son principe est d'appliquer le découplage de degrés de liberté du manipulateur de façon qu'un certains nombre de degrés de liberté agissent en force et les autres en position.

Ce type de méthode a l'avantage d'être facilement réalisable et très efficace pour certaines tâches comme le suivie de trajectoire et la commande en force.

Le contrôleur position hybride / force doit résoudre trois problèmes:

1. Commande de la position d'un manipulateur selon des directions dans laquelle une contrainte de force naturelle existe.
2. Commande en force d'un manipulateur selon des directions dans laquelle une contrainte de position naturelle existe.
3. La mise en œuvre du mélange arbitraire de ces modes tout au long des degrés de liberté orthogonaux d'un repère arbitraire,  $\{C\}$ .

Une structure générale de cette commande peut être représentée par le schéma fonctionnel ci-après :



**Figure 2.1.** Principe de la commande hybride position / force.

*LCP: Loi de commande en position;*

*LCF : Loi de commande en Force.*

La matrice  $S$  est définie comme une matrice diagonale, sa diagonale composée de 1 et 0, pour activer la commande en position dans le repère de contrainte, nous allons définir un 1 sur l'élément  $(i, i)$  de  $S$ . Et afin de désactiver la commande en force un 0 sur la même ligne  $i$  sera choisi, le rôle de la matrice  $S$  et leur complément  $I-S$  est d'activer une seule commande que ce soit en position ou en force.

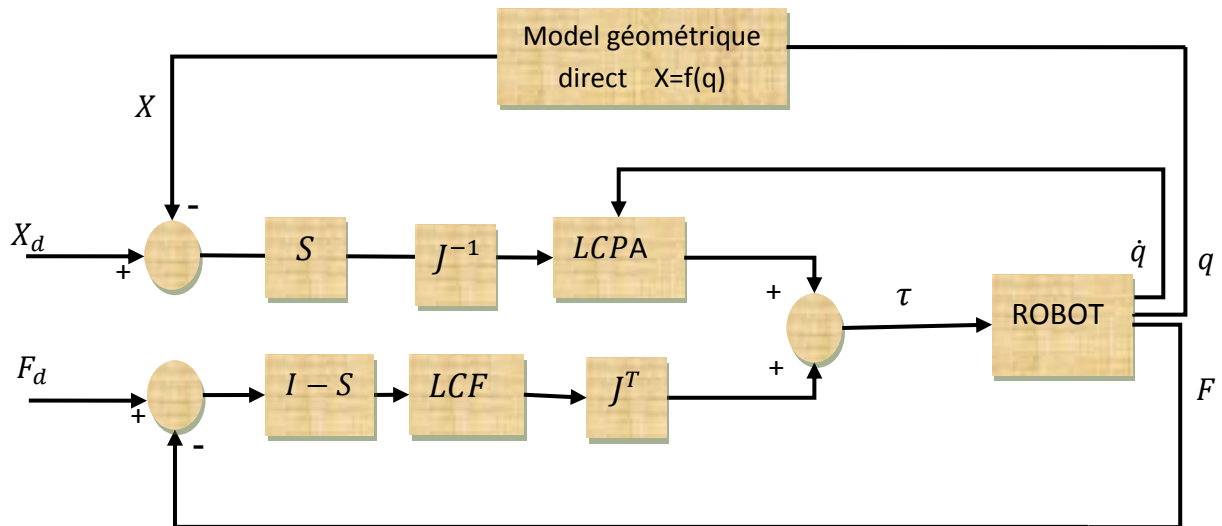
Comme montre le schéma ci-dessus, les lois de commande des deux boucles ne sont pas définies, donc Nous pouvons choisir des lois de commande en position ou en force qui sont différentes, notamment adaptative ou robuste [12].

On distingue trois types de commande hybride force/position, Selon la nature des grandeurs de  $U$  (signale de commande résultant) qui est la somme des sorties des deux boucles :

- a. *Commande avec somme des couples articulaires.*
- b. *Commande avec somme des déplacements ou vitesses.*
- c. *Commande avec somme des efforts cartésiens.*

### 2.3.1.1 Commande hybride force/position avec somme des couples articulaires :

Raibert et Craig ont proposé cette architecture en 1981, dans ce cas la loi de commande en position est élaborée dans l'espace articulaire LCPA, et une loi de commande en effort LCF. La figure 2.2 montre son principe.



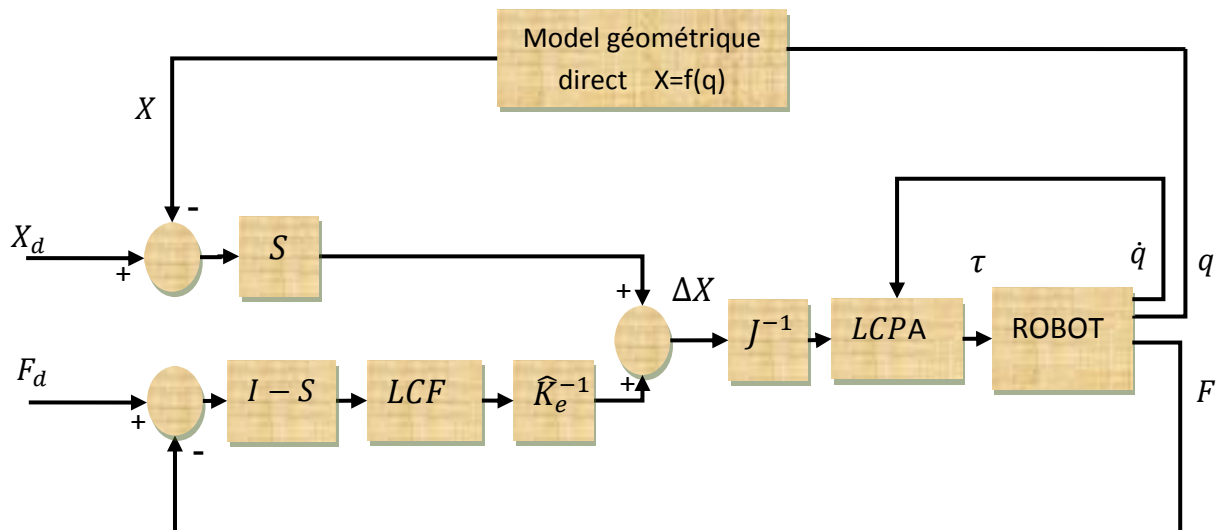
**Figure 2.2.** Commande hybride force/position avec somme des couples.

*LCPA*: loi de commande en position dans l'espace articulaire.

*LCF* : Loi de commande en Force.

**2.3.1.2 Commande avec somme des déplacements ou vitesses :**

Reboulet and Robert, ont proposé cette architecture en 1985, La sortie du sommateur est un déplacement  $\Delta X$ , qui sera transformée via la matrice Jacobienne inverse à l'espace articulaire, une loi de commande de position articulaire est calculée dans cet espace (figure 2.3).

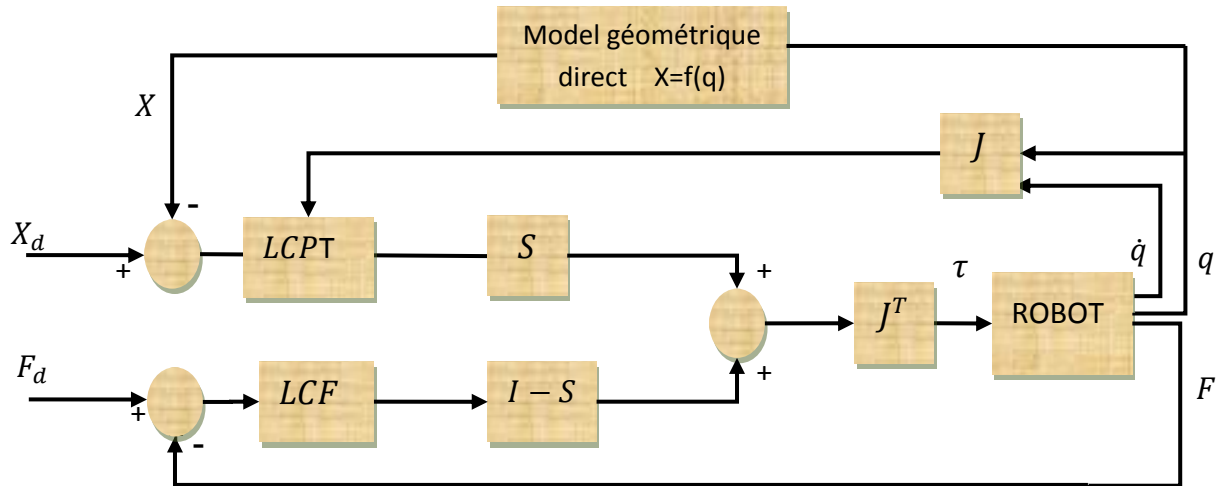


**Figure 2.3.** Commande hybride force/position avec somme des déplacements cartésiens.

$\hat{K}_e$  : Estimation de la raideur de l'environnement.

**2.3.1.3 Commande hybride force/position avec somme des forces cartésiennes :**

Khatib.O a proposé cette architecture en 1985, Dans ce cas les couples actionneurs sont obtenus en transformant la somme des forces cartésiennes via la transposée de la matrice Jacobienne du robot (figure 2.4) :



**Figure 2.4.** Commande hybride force/position avec somme des forces.

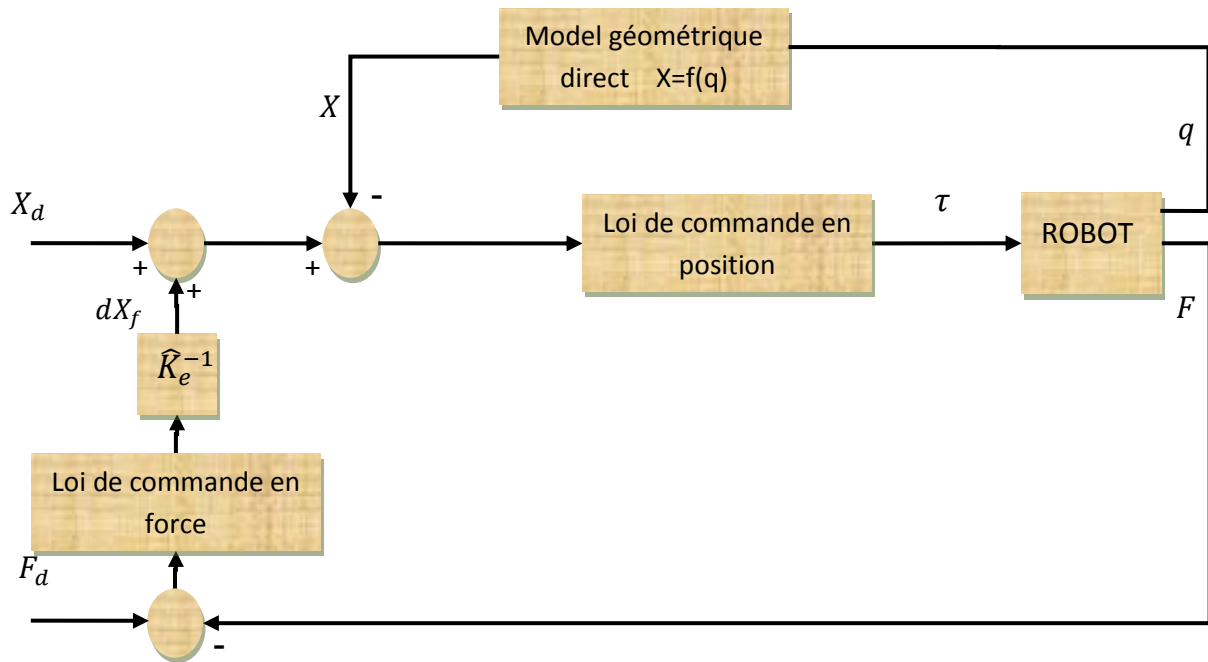
*LCPT : loi de commande en position dans l'espace des taches.*

**2.3.2 Commande Hybride Externe (Structures série) :**

Le nom de La structure externe provient du fait qu'elle est constituée d'une **boucle externe** de la force avec une boucle de position interne. L'idée principale de cette solution c'est que l'erreur de la force peut être transformée en un déplacement au niveau de l'outil terminal. Pour ce faire, à partir de l'erreur de la force, nous calculons un incrément de position qui modifie une consigne de position initiale. En d'autres termes, la position opérationnelle du robot (dans l'espace des taches) est asservit pour suivre une trajectoire désirée, et des qu'il existe un contact avec l'environnement, cette dernière est corrigé automatiquement par la commande en force [12].

La Figure ci-après illustre ce principe :





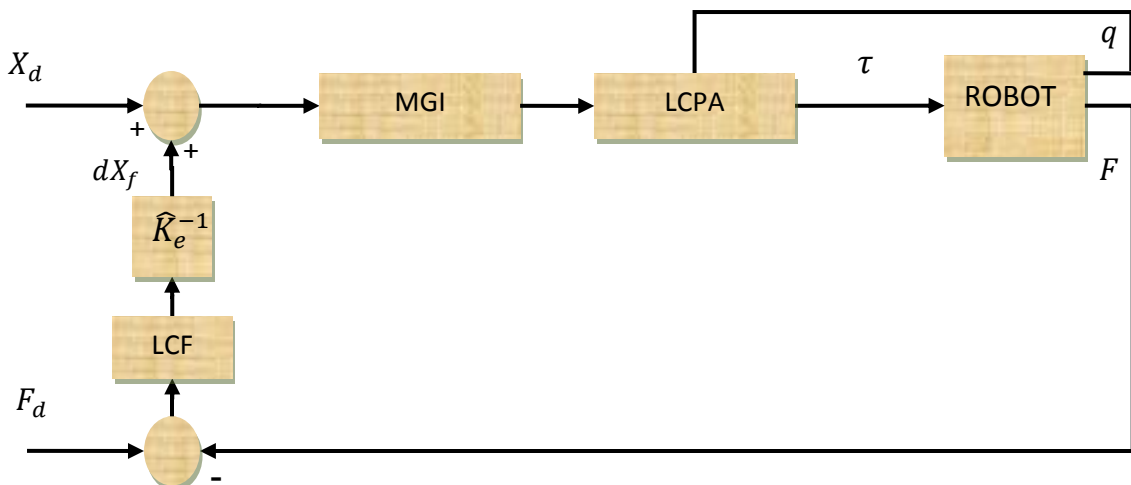
**Figure 2.5.** Schéma de principe de la commande hybride externe.

$\hat{K}_e^{-1}$  : Estimation de l'inverse de la matrice de rigidité.

Selon l'espace où la correction des positions s'effectue, on distingue deux types de commande hybride externe [27]

**2.3.2.1 Commande Hybride Externe avec Correction Articulaire de Position :**

La consigne de la position et la correction issue de la boucle externe sont additionnées dans l'espace des tâches puis transformées vers l'espace articulaire à l'aide du modèle géométrique inverse, le principe est illustré dans le schéma suivant :

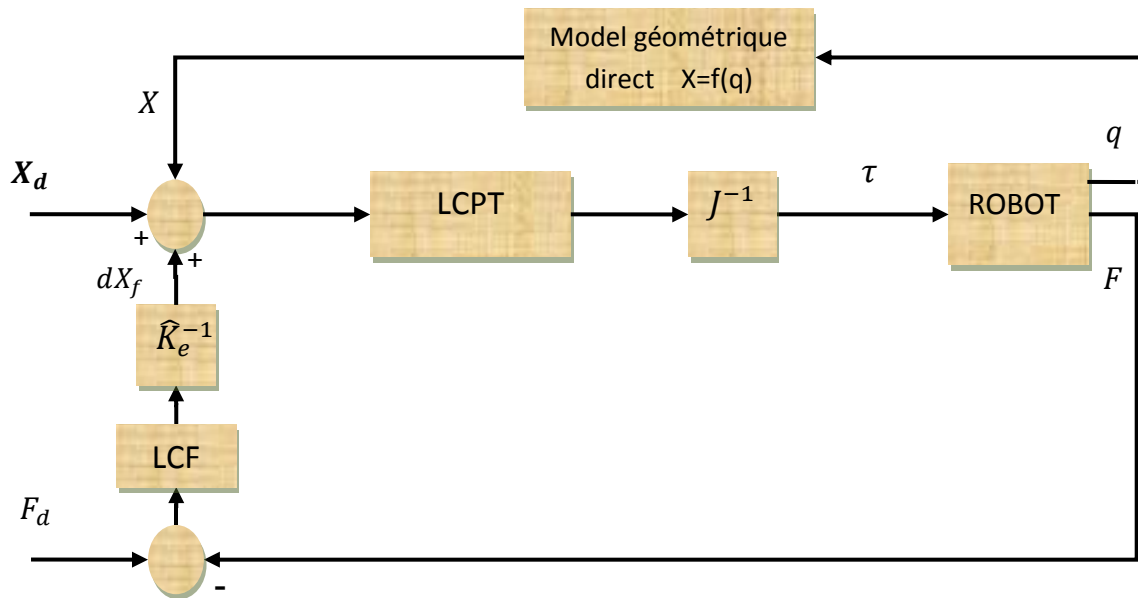


**Figure 2.6.** Commande hybride externe avec correction de la position articulaire.

*MGI* : model géométrique inverse.

**2.3.2.2 Commande Hybride Externe avec Correction Cartésienne de Position :**

Dans ce cas, la correction de la position est effectuée dans l'espace cartésien en transformant la position articulaire en position cartésienne à l'aide du modèle géométrique direct. Les consignes seront sommées dans l'espace cartésien. (Figure 2.7).



**Figure 2.7.** Commande hybride externe avec correction de la position cartésienne.

**2.4 Commande en force indirecte :**

C'est une commande sans consigne en force, la force est commandée implicitement, citant la Commande par raideur active, commande par matrice d'amortissement et commande en impédance.

**2.4.1 Commande passive :**

La commande passive est simple, l'installation d'un organe mécanique souple sur l'effecteur du manipulateur pour répondre aux contraintes de l'environnement, c'est un type purement mécanique.

**2.4.2 Commande par raideur active :**

La commande par raideur active c'est une commande qui permet au manipulateur d'avoir une raideur apparente vis à vis l'environnement, son équivalence en mécanique est un ressort

linéaire avec raideur prédéfinie installées dans les différentes directions de l'effecteur pour permettre au bras manipulateur de bien gérer le contact avec l'environnement.

Cette méthode consiste à définir en statique une fonction (une matrice de raideur) qui relie la force d'interaction au déplacement  $y$  correspond, dans l'espace cartésien.

Cette matrice est diagonale avec des valeurs élevées attribuées aux directions qui sont contraintes en force (commandées en position), tandis que des faibles gains sont affectés aux directions contraintes en position (commandé en force).

Donc simplement les gains de position dans l'espace articulaire sont modifiés de telle sorte que l'effecteur semble avoir une certaine raideur le long des degrés de liberté cartésiens.

La formulation de la rigidité de base est donnée par :

$$F = K dX \quad (2.5)$$

$K$ : La matrice de rigidité désirée (diagonal dans le repère de compliance), qui relie les forces mesurées à l'effecteur aux déplacements de la position nominale

$dX$ : Déplacement généralisés de la position nominal commandée  $x_0$  de l'effecteur.

$F$ : La force appliquée par le bras sur l'environnement

Donc nous voulons restaurer les couples correspondants à ces déplacements de la commande nominale (sans contrainte) pour que la formule linéaire (2.5) soit maintenue [30]

En utilisant la jacobienne on obtient :

$$dX = J q \quad (2.6)$$

Supposant maintenant que les forces de frottement et la dynamique du bras sont compensés ou sont suffisamment faibles qu'on peut les négligés on obtient :

$$\tau = J^T F \quad (2.7)$$

A partir des équations (2.5) (2.6) (2.7)

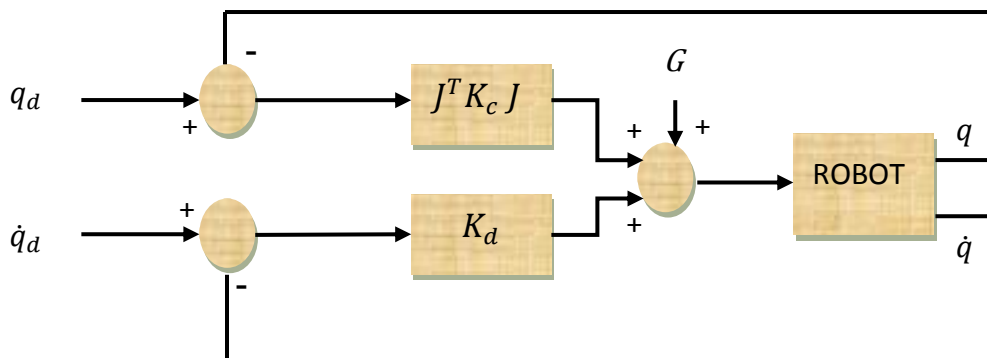
$$\tau = J^T K J dq = K_q dq \quad (2.8)$$

Le terme  $K_q$  s'appelle matrice de raideur articulaire [30].

Cette formule est centrale dans la commande par raideur elle a l'avantage qu'elle ne demande pas la matrice Jacobienne inverse,  $K_q$  est non diagonale mais symétrique [20] Cela signifie qu'une simple erreur dans une seule articulation affecte la commande dans toutes les autres articulations [30].

Donc La loi de commande par raideur active est formulée comme suite :

$$\tau = K_q(q_d - q) + K_d(\dot{q}_d - \dot{q}) + G \quad (2.9)$$



**Figure 2.8.** Commande par raideur active.

La matrice de rigidité peut être modifiée en ligne afin d'adapter le comportement du robot à diverses contraintes de travail.

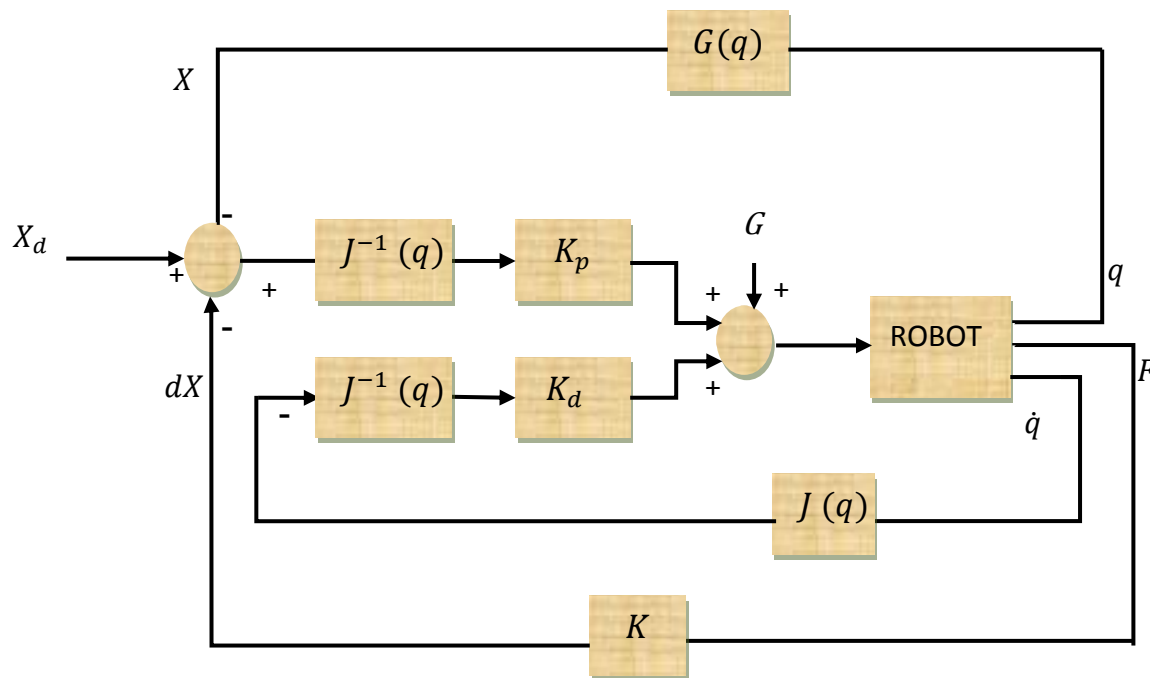
### 2.4.3 Commande par Raideur avec retour de force :

Un réglage indirect de l'effort grâce à la matrice  $K$  qui correspond la matrice de raideur désirée du robot [37]. Les éléments de cette matrice sont prédéfinis en imposant des valeurs faibles pour les directions contraintes afin d'assurer une faible force d'interaction et des valeurs élevées pour les directions non contraintes afin d'assurer un suivi correcte de la position désirée.

La correction de la position prend en considération le déplacement  $dX$  dû à l'effort à travers la Matrice de raideur souhaitée du robot selon la relation :

$$dX = K F \quad (2.10)$$

La figure suivante présente le principe de cette commande, elle présente en plus une correction proportionnelle-dérivée classique.



**Figure 2.9.** Schéma de principe de la commande en raideur avec retour de la force.

#### 2.4.4 Commande par Matrice d'Amortissement :

Cette méthode est semblable en forme à la précédente mais ici la commande en vitesse au lieu de la position, Cette méthode peut être modélisée par la relation suivante [24]

$$F = B(V - V_0) \tag{2.11}$$

Avec :  $F$  : La force de l'effecteur exercée sur l'environnement.

$B$  : La matrice d'amortissement, qui relie la force de l'effecteur à la déviation de la vitesse par rapport à sa vitesse nominale.

$V$  : La vitesse.

$V_0$  : La vitesse nominale.

La commande de la force est aboutit via la différence de la vitesse actuelle du robot par rapport à celle nominale (figure 2.4) [37]. La force mesurée est multiplié par la matrice  $K_{vf}$ , ses éléments sont choisis en imposant des valeurs faibles pour les directions non

contraintes et des valeurs élevées dans les directions où le contact est attendu. L'objectif est d'atteindre l'état où la force de l'effecteur soit fonction de la déviation de la vitesse par rapport à sa vitesse nominale.

Quand il n'y a pas de contact  $F = 0 \Rightarrow \dot{X}_d$  reste inchangé.

Quand il y a un contact  $F \neq 0 \Rightarrow \dot{X}_d$  est changé suivant la relation  $\dot{X}_d = B F$ .

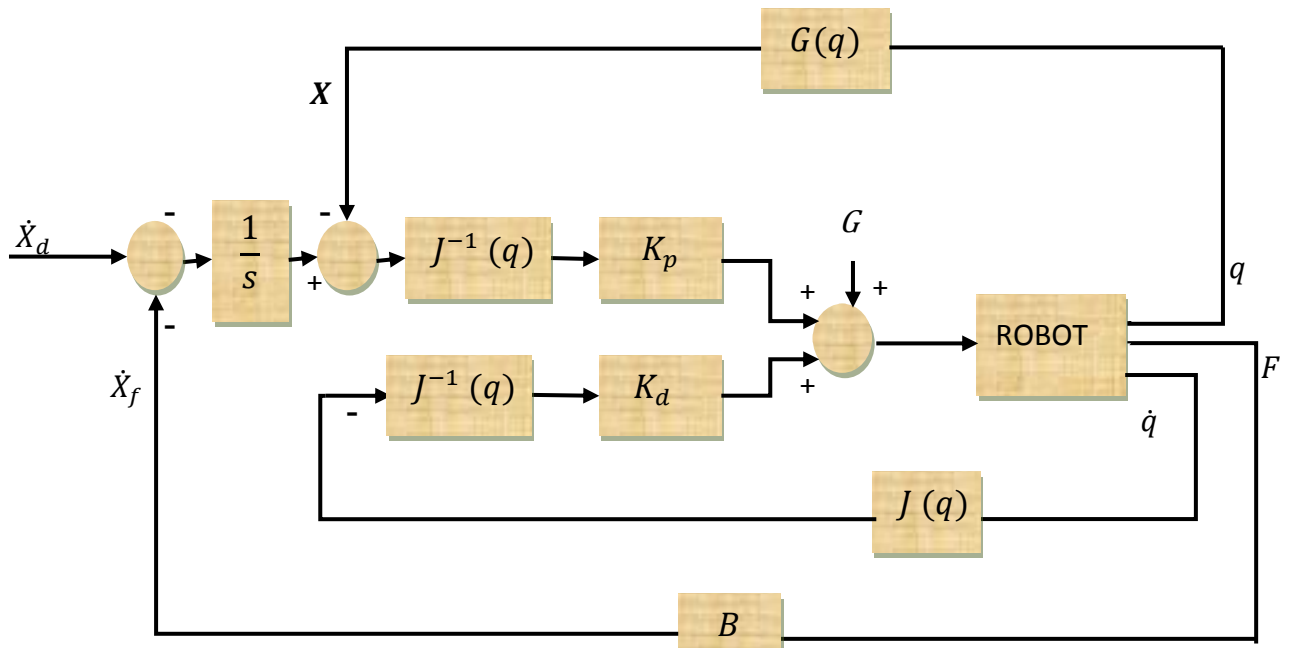


Figure 2.10. Schéma de principe de la commande en amortissement avec retour de la force.

## 2.4.5 Commande en impédance :

### 2.4.5.1 Principe :

Aucune stratégie de commande physiquement réalisable ne peut éliminer l'inertie du manipulateur, mais l'inertie apparente vue au niveau de l'effecteur peut être modifiée. [15]. Sur cette base qu'on peut masquer l'inertie non linéaire du manipulateur et imposer une dynamique plus simple, c'est l'impédance désirée, que ce n'est qu'une extension et une combinaison entre la commande en raideur et la commande en amortissement.

L'impédance décrit la relation entre La vitesse de L'effecteur terminal  $\dot{X}$  et la force appliquée, Dans le domaine de fréquence, ceci est représenté par:

$$f(s) = Z(s) \dot{X}(s) \quad (2.12)$$

En termes de position  $X(s)$ , nous pouvons écrire:

$$f(s) = s Z(s) X(s) \quad (2.13)$$

Le réglage d'impédance assure que le manipulateur, soumis à une force externe, a le comportement d'une impédance constituée d'un ensemble de: l'inertie ( $A$ ), ressort avec raideur ( $K$ ) et amortisseur ( $B$ ), d'une manière générale le robot est considéré équivalent à un système de second ordre masse-ressort-amortisseur [20], dont la fonction de transfert est:

$$sZ(s) = As^2 + Bs + K = \frac{F(s)}{X(s)} \quad (2.14)$$

Où  $A$ ,  $B$  et  $K$  représentent respectivement les matrices d'inertie souhaitée, d'amortissement et de rigidité. Les valeurs de ces matrices sont choisies de manière à obtenir la performance souhaitée.

La forme la plus simple d'un dispositif de commande d'impédance est l'une impédance d'ordre zéro. Dans ce cas,  $Z$  est une constante et :

$$F = K X \quad (2.15)$$

Une forme plus typique d'un contrôleur d'impédance est une impédance du premier ordre. Dans ce cas,

$$F = (Bs + K)X \quad (2.16)$$

Le paramètre ajouté  $B$  est l'amortissement souhaité du manipulateur, égal à la somme de l'amortissement actif et naturel. En effet, l'amortissement actif négatif peut être utilisé pour éliminer l'apparition d'une perturbation dans le bras (due à la friction et roue dentée) [35]. Enfin, une forme plus complète de contrôle d'impédance est fournie par le second type d'ordre,

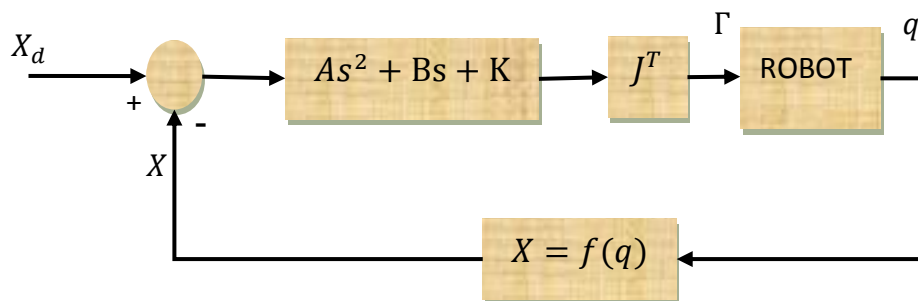
$$F = (As^2 + Bs + K)X \quad (2.17)$$

Le paramètre  $A$  est l'inertie désirée du manipulateur. Alors que l'inertie intrinsèque du bras est due à sa masse, il peut être modifié par une rétroaction active.

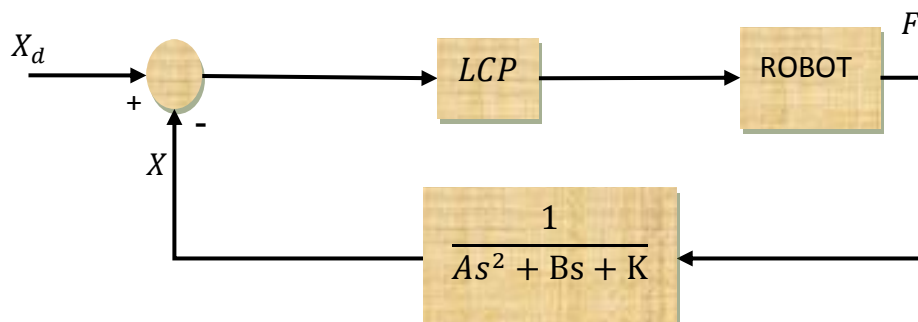
**REMARQUE :**

- Des valeurs élevées sont donnés à  $M$  dans les directions où un contact est prévu afin de limiter la dynamique du robot;
- Des valeurs élevées de  $B$  dans les directions où il est nécessaire de dissiper l'énergie cinétique et donc d'amortir la réponse;
- La rigidité  $K$  affecte la précision de la régulation de position: selon les directions de la force contrôlée, la rigidité doit être suffisamment faible pour limiter les forces de contact; à l'inverse, le long des directions contrôlées en de position, l'utilisateur doit définir une grande rigidité pour obtenir un positionnement précis de l'effecteur [20].

Deux formulations peuvent être présentées selon l'existence d'une mesure de la force ou non :



**Figure 2.11.** Schéma de Principe de la Commande d'impédance **sans** retour de force.



**Figure 2.12.** Schéma de Principe de la Commande d'impédance **avec** retour de force.

*LCP : loi de commande en position.*

Cette deuxième méthode présente deux approches selon l'expression du modèle de robot en espace articulaire ou opérationnel [12]



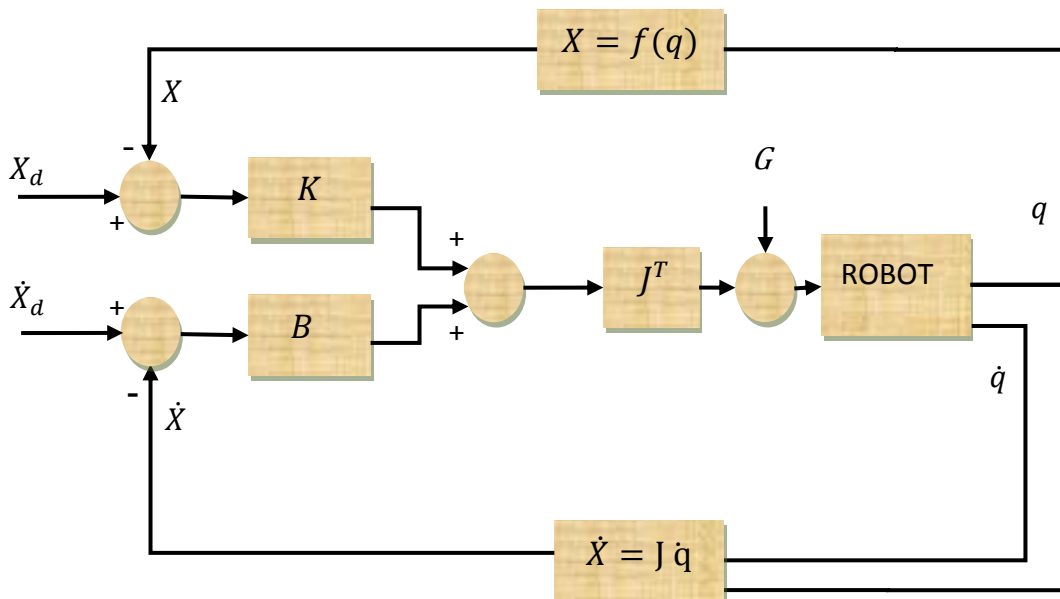
**2.4.5.2 Commande en Impédance sans retour de force :**

Cette première approche est une commande implicite de la force qui n'a pas besoin de mesure de la force extérieure et ne prend pas en compte le modèle dynamique du robot (sauf pour la compensation de gravité), donc la dynamique du robot est négligée. Dans ce cas, le vecteur de commande est choisi de la manière suivante:

$$\tau = J^T (B_d \dot{E} + K_d E) + G \tag{2.18}$$

$E = X_d - X$  : L'erreur entre la position  $X$  actuelle et la position désirée  $X_d$ .

- Les éléments de la matrice  $B_d$  représentent l'amortissement désiré dans chaque degré de liberté de l'effecteur cela a été identifié selon la tâche (environnement).
- Les éléments de la matrice  $K_d$  représentent la raideur désirés dans chaque degré de liberté de l'effecteur.
- Le terme d'inertie est nul puisque on a déjà négligé la dynamique du robot.
- Le terme de gravité  $G$  dans la loi de commande pour compenser la gravité du robot.
- Cette commande est l'équivalent d'un asservissement par action proportionnelle-dérivée dans l'espace opérationnel [20].



**Figure 2.13.** Schéma de la Commande d'impédance **sans** retour de force.

### 2.4.5.3 Commande en Impédance avec retour de force :

Dans cette seconde famille, existe deux formes de commande en impédance, selon l'expression du modèle dynamique du robot dans l'espace articulaire, ou dans l'espace des tâches.

Notons que le modèle dynamique d'un robot exerçant une force  $f$  sur son environnement est écrit comme suite :

$$M\ddot{q} + V\dot{q} + G + J^T F_e = \tau \quad (2.19)$$

Nous posons :  $H = V\dot{q} + G$  , donc l'équation (2.19) devient :

$$M\ddot{q} + H + J^T F = \tau \quad (2.20)$$

$M$  : La matrice d'inertie du manipulateur.

$V$  : Le vecteur des forces centrifuges et de Coriolis.

$G$  : Le vecteur de gravité.

$F_e$  : La force exercée par le manipulateur sur l'environnement.

$\tau$  : Le vecteur des couples appliqués aux moteurs installés aux différentes articulations.

La dynamique désirée est formulée comme suite :

$$A(\ddot{X}_d - \ddot{X}) + B(\dot{X}_d - \dot{X}) + K(X_d - X) = F \quad (2.21)$$

Donc

$$\ddot{X} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F ] \quad (2.22)$$

Pour obtenir  $\ddot{q}$  en fonction de  $\ddot{X}$  nous utilisons le jacobien

$$\dot{X} = J \dot{q} \Rightarrow \ddot{X} = \dot{J} \dot{q} + J \ddot{q} \Rightarrow$$

$$\ddot{q} = J^{-1}(\ddot{X} - \dot{J} \dot{q}) \quad (2.23)$$

De (2-23) et (2-20) on peut écrire :

$$\tau = M J^{-1}(\ddot{X} - \dot{J} \dot{q}) + H + J^T F \quad (2.24)$$

Par l'application de la méthode de découplage non linéaire, on doit choisir la commande  $\tau$  de telle façon la dynamique désirée (2-21) soit réalisée :

$$\tau = \widehat{M} J^{-1}(w - j \dot{q}) + \widehat{H} + J^T F \quad (2.25)$$

Avec :

$$w = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F ] \quad (2.26)$$

Et :  $\widehat{M}, \widehat{H}$  L'estimation de  $M, H$  respectivement.

Donc la loi de commande se présente comme suite :

$$\tau = \widehat{M} J^{-1} [ \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F ] - j \dot{q} ] + \widehat{H} + J^T F \quad (2.27)$$

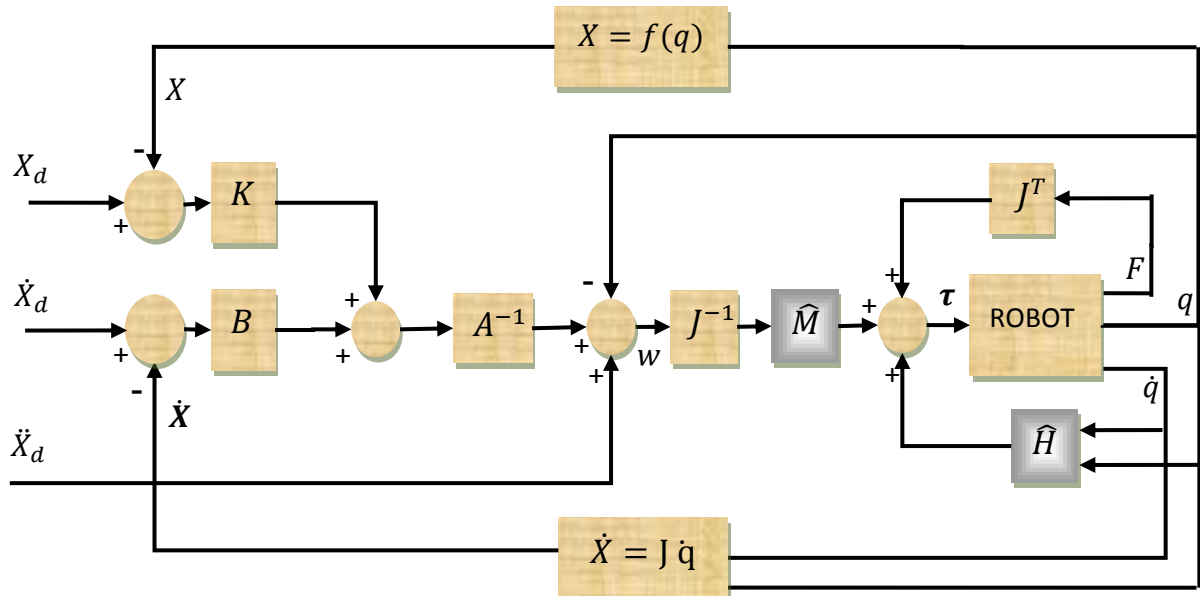
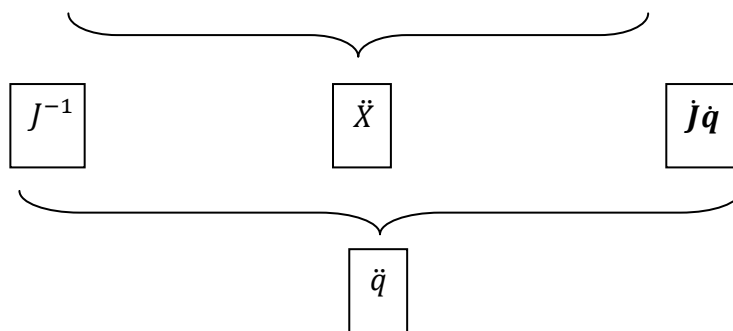


Figure 2.14. Schéma de la Commande d'impédance avec retour de force.

L'autre formulation peut être achevée par le modèle du robot en espace des taches, tout en utilisant les différentes transformations géométriques et cinématique, on arrive au modèle suivant :

$$M_x \ddot{X} + V_x \dot{X} + G_x + F = J^{-T} \tau \quad (2.28)$$

$M_x = J^{-T} M J^{-1}$  : Représente l'inertie du manipulateur à l'espace des taches

$V_x = J^{-T} V J^{-1} - M_x \dot{J} J^{-1}$  : Représente les forces centrifuges et de Coriolis du manipulateur dans l'espace des taches.

$G_x = J^{-T} G$  : Représente la gravité dans l'espace des taches

On fait la même démarche pour trouver la loi de commande par la méthode de découplage non linéaire :

$$\tau = J^T (\hat{M}_x w + \hat{V}_x \dot{X} + \hat{G}_x + F) \quad (2.29)$$

Avec :

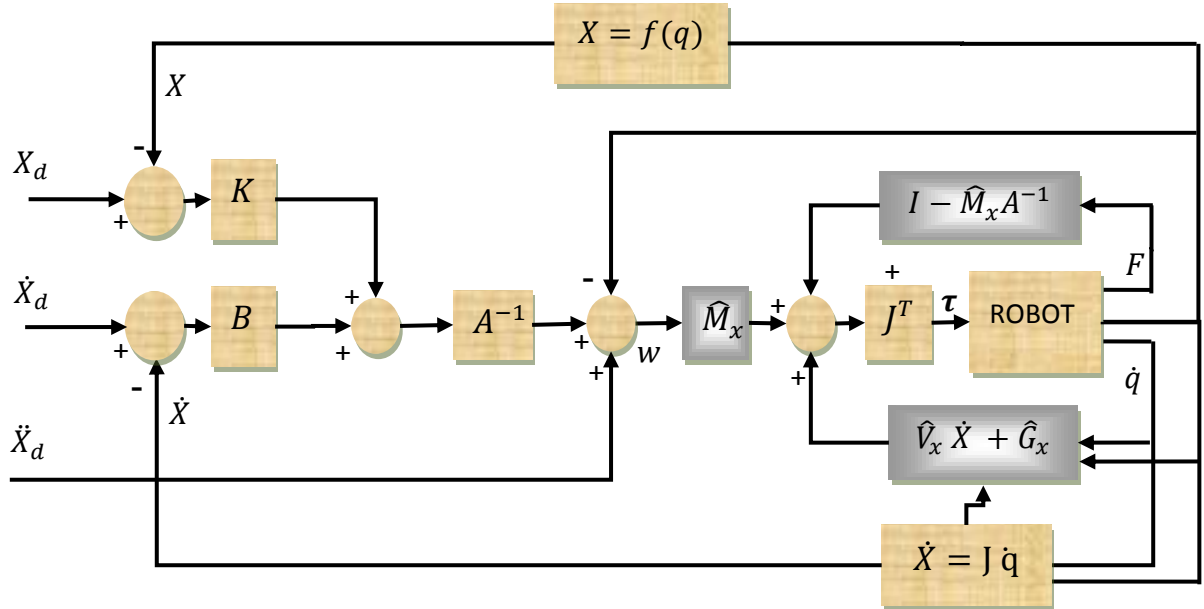
$$w = \ddot{X} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F ] \quad (2.30)$$

$\hat{M}_x, \hat{V}_x, \hat{G}_x$  : L'estimation de  $M_x, V_x, G_x$  respectivement.

En remplaçant  $w$  par son expression dans (2-29) on obtient :

$$\tau = J^T \hat{M}_x \underbrace{\left[ \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F ] \right]}_w + J^T (\hat{V}_x \dot{X} + \hat{G}_x) + J^T F \quad (2.31)$$

$$\tau = J^T \hat{M}_x \left[ \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) ] \right] + J^T [ (\hat{V}_x \dot{X} + \hat{G}_x) + (I - \hat{M}_x A^{-1}) F ] \quad (2.32)$$



**Figure 2.15.** Schéma de la Commande en impédance par découplage non linéaire avec retour de force

**Simplification :**

- Dans les cas quasi-statique le terme  $\hat{V}_x \dot{X}$  est considéré nulle [19], et l'équation (2.31) devient:

$$\tau = J^T \hat{M}_x A^{-1} [A \ddot{X}_d + B(\dot{X}_d - \dot{X}) + K(X_d - X) - F] + G + J^T F \quad (2.33)$$

- Dans le cas ou  $\hat{V}_x \dot{X} \neq 0$ , et avec Le choix de  $A = \hat{M}_x$ , donc la force dans l'équation (2.32) va s'annuler, on peut écrire :

$$\tau = J^T \hat{M}_x \left[ \ddot{X}_d + A^{-1} [B(\dot{X}_d - \dot{X}) + K(X_d - X)] \right] + J^T (\hat{V}_x \dot{X} + \hat{G}_x) \quad (2.34)$$

Avec  $\hat{M}_x = J^{-T} \hat{M} J^{-1}$

$$\hat{V}_x = J^{-T} \hat{V} J^{-1} - \hat{M}_x j J^{-1} \quad \text{Et} \quad \dot{X} = J \dot{q}$$

Donc

$$\hat{V}_x \dot{X} = (J^{-T} \hat{V} - \hat{M}_x j) \dot{q} \quad (2.35)$$

On substitue l'équation (2.35) dans l'équation (2.34) on trouve :

$$\tau = J^T \hat{M}_x [\ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) ] - j \dot{q}] + \hat{V} \dot{q} + \hat{G} \quad (2.36)$$

#### 2.4.5.4 Commande en Impédance Généralisée :

La commande en impédance conventionnelle n'est pas applicable dans des situations de compliance qui exigent une commande explicite en force (consigne en force) [9], avec la combinaison entre la commande hybride externe et celle en impédance Nous pouvons arriver à commander le manipulateur en impédance avec une consigne en force, c'est ce qui a été rapporté dans la littérature sous le nom de la Commande en Impédance Généralisée.

Notons que la force désirée n'a pas de place dans l'équation de l'impédance conventionnelle, l'impédance étendue prend en charge cette inconvénient, donc l'impédance généralisée est donnée aussi par une équation de second ordre :

$$A(\ddot{X}_d - \ddot{X}) + B(\dot{X}_d - \dot{X}) + K(X_d - X) = K_f(F_d - F) \quad (2.37)$$

À travers cette nouvelle équation d'impédance, le robot est capable d'agir et aboutir aux positions, vitesses et accélérations spécifiques, selon l'erreur entre la force désirée et la force mesurée pondérée avec une matrice diagonale  $K_f$ .

Si  $K_f = I_n$  (matrice identité  $n \times n$ ) et  $F_d = 0$ , donc l'impédance généralisée devient l'originale. [9]

La force d'interaction désirée  $F_d$  est introduite pour que la force de contact soit directement contrôlée lors de contact. Une loi de commande basée sur le modèle dynamique du robot est adoptée pour réaliser l'objectif de l'impédance généralisée.

Pour le robot manipulateur décrit par (2.19), il peut réaliser l'impédance cible généralisée (2.37) si la loi de commande suivante est mise en œuvre:

$$\tau = \hat{M} J^{-1} (w - \dot{j} \dot{q}) + \hat{H} + J^T F \quad (2.38)$$

Avec :

$$\mathbf{w} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - K_f(F_d - F) ] \quad (2.39)$$

$\hat{M}, \hat{H}$  L'estimation de  $M, H$  respectivement.

**Démonstration :** par la substitution de la commande (2.38) dans l'équation (2.19) (modèle robot), avec des estimations parfaites de  $\hat{H}$  et  $\hat{M}$ , nous aboutissons à :

$$\mathbf{w} = \ddot{X} \quad \text{Ce qui implique } \ddot{X} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - K_f(F_d - F) ]$$

Nous arrivons à l'impédance désirée :

$$A(\ddot{X}_d - \ddot{X}) + B(\dot{X}_d - \dot{X}) + K(X_d - X) = \mathbf{K}_f(\mathbf{F}_d - \mathbf{F})$$

Ce qui représente l'impédance désirée.

L'impédance cible combinée avec la dynamique de l'environnement doit être définie d'une manière que, la stabilité du système en boucle fermée soit préservée et la force de contact désirée obtenue. Supposons que, dans l'espace de tâches, l'environnement est modélisé comme suite :

$$A_e \ddot{X} + B_e \dot{X} + K_e(X_e - X) = F_e \quad (2.40)$$

Où :  $X_e$  est la position d'équilibre de l'environnement dans l'absence de forces d'interaction.  $A_e, B_e$  et  $K_e$  sont des matrices SDP (semi définie positive) qui représentent respectivement l'inertie, amortissement, et raideur de l'environnement. La dynamique d'ensemble (manipulateur et de l'environnement) est obtenue en combinant (2.37) et (2.40):

$$\begin{aligned} (A_d + K_f A_e) \ddot{X} + (B_d + K_f B_e) \dot{X} + (K_d + K_f K_e) X \\ = A_d \ddot{X}_d + B_d \dot{X}_d + K_d X_d + K_f F_d + K_f K_e X_e \end{aligned} \quad (2.41)$$

Qui est asymptotiquement stable si  $M, B, K$  et  $K_f$  sont choisies diagonales définies positives [19] et [9].

## 2.5 Conclusion:

Nous avons vu dans ce chapitre les différents types de la commande en force, ces techniques de commande sont divisées globalement en deux grandes familles celle qui contrôle directement la force elles s'appellent la commande en force directe **ou explicite** son objectif principale est de commander la force et la position de l'effecteur séparément une loi de commande en force est destinée à contrôler les degrés de libertés qui sont affectés par les contraintes physique devant sa trajectoire nominale (contrainte de la position), et une autre loi de commande en position pour commander les degrés de liberté qui sont en espace libre (contrainte en force), dans cette catégorie de commande existe plusieurs mode de contrôle à l'instar de la **commande hybride position/force (structure parallèle dans la littérature)** ce type de control contient deux boucles de commande en position et en force, en parallèle la sélection d'une loi de commande se fait par une matrice de sélection  $S$  et leur complément  $I-S$  pour permettre l'activation d'une seule loi de commande au même temps.

Les autre types de commande sont fréquemment connus sous le nom de **la commande hybride externe (structure série)**, qui contient deux boucles de commande imbriquées une boucle d'asservissement de la position interne gouvernée par une autre boucle externe de force.

La deuxième catégorie, qui s'appelle **la commande en force indirecte ou implicite**, c'est une autre stratégie de commande elle ne s'intéresse pas à la commande de la force seule ou la position seule, mais elle se focalise sur la dynamique d'interaction entre l'environnement et l'effecteur du manipulateur, cette dynamique est dirigée par une dynamique désirée définie préalablement par l'opérateur pour donner au robot un comportement linéaire découplé, à partir de laquelle le robot détecte les force sur son effecteur venant de l'environnement et les traduit au biais de l'impédance (dynamique désirée) en un déplacement ou une vitesse modifiant ainsi la position et/ou la vitesse désirée.

On distingue trois types de commande en impédance, commande par raideur, par amortissement ou par impédance.



Chaque type de commande en force contient des inconvénients -comme bien sure possède beaucoup d'avantage-, la commande hybride exige une connaissance parfaite de l'environnement.

La commande en impédance conventionnelle ne permet pas de contrôler la force avec une consigne désirée, **la commande en impédance généralisée** [29] [9] a été développée pour surmonter cet inconvénient.

# Chapitre III

## La Commande non Linéaire

### 3.1 Introduction :

Un système obéit au principe de superposition si sa réponse à deux entrées différentes donne deux sorties alors sa réponse à la somme de ces deux entrées est égale à la somme de ces deux sorties, Les systèmes obéissent à ce principe sont des systèmes linéaires.

Les systèmes non linéaires sont des systèmes basés sur le non respect de la notion de superposition, les algorithmes de commandes classiques ne sont applicables aux systèmes non linéaire qu'autour du point de fonctionnement où les systèmes sont considérés linéaires et les méthodes linéaires sont ainsi adéquats et donnent des bons résultats, Cependant, certaines applications industrielles, exigent la prise en compte des non linéarités des processus dans la synthèse de la commande.

Dans ce chapitre on commence par la commande par découplage non linéaires où l'objectif principal de cette méthode est la linéarisation totale du système par bouclage, donc après la compensation de toutes les non-linéarités, les algorithmes de commande linéaires peuvent être appliqués.

Dans la deuxième section on aborde la commande par backstepping, cette méthode de commande non linéaire est basé sur la stabilité par Lyapunov, la commande est synthétisée étape à étape, assurant ainsi la stabilité de tous les états du système.

Finalement on développe la méthode de commande par mode glissant, cette commande robuste est basée sur la projection du système à commander sur une surface généralement linéaire, cette dernière est choisit pour répondre aux exigences de la stabilité et précision

### 3.2 Commande par découplage non linéaire :

Ce type de commande est aussi connu sous le nom de *commande dynamique* (ou "couple calculé", *computed torque* dans la littérature anglo-saxonne) parce qu'il est fondé sur l'utilisation du modèle dynamique. Théoriquement, il assure **le découplage et la linéarisation** des équations du modèle, ayant pour effet une réponse uniforme quelle que soit la configuration du robot.

La commande par découplage non linéaire consiste à transformer par retour d'état le problème de commande d'un système non linéaire en un problème de commande d'un système linéaire. Dans le cas général, le problème de linéarisation par retour d'état d'un système non linéaire n'est pas facile à résoudre.

Cependant, grâce aux méthodes de modélisation et grâce aux évolutions technologiques en micro-informatique. Le développement des techniques d'identification permet une bonne évaluation des paramètres dynamiques [21].

#### 3.2.1 Principe de la Commande par découplage non linéaire :

Dans cette section nous utilisons l'équation dynamique du robot pour développer la commande par découplage non linéaire

$$M\ddot{q} + H = \tau \quad (3.1)$$

**Avec :**  $H = V\dot{q} + G$

La commande  $\tau$  est choisit comme suite [20] :

$$\tau = \hat{M}(q) w(t) + \hat{H}(q, \dot{q}) \quad (3.2)$$

Soit  $\hat{M}$  ,  $\hat{H}$  les estimations de  $M$  ,  $H$  respectivement,

On distingue deux cas de commande Suivant le modèle du robot.

##### 3.2.1.1 Modèle certain :

Dans ce cas les positions et les vitesses articulaires sont mesurables et que les mesures ne sont pas bruitées, donc le modèle est exacte sans incertitudes alors :

$$\hat{M} = M \text{ et } \hat{H} = H .$$

Le système en boucle fermée est régi par l'équation :

$$\ddot{q} = w(t) \quad (3.3)$$

$w(t)$  peut être considéré comme un nouveau vecteur de commande. On se ramène donc à un problème de commande de  $n$  sous-systèmes linéaires, invariants, découplés et du second ordre (doubles intégrateurs).

Deux cas sont envisageables selon la trajectoire est définie avec une vitesse et accélération désirées ( $q_d, \dot{q}_d, \ddot{q}_d$ ) ou bien uniquement une position désiré à atteindre  $q_d$

### 3.2.1.1.1 Trajectoire définie complètement ( $q_d, \dot{q}_d, \ddot{q}_d$ ) :

On désigne respectivement par  $q_d(t), \dot{q}_d(t), \ddot{q}_d(t)$  la position, la vitesse et l'accélération désirées dans l'espace articulaire. On calcule  $w(t)$  selon la relation suivante :

$$w = \ddot{q}_d + k_d(\dot{q}_d - \dot{q}) + k_p(q_d - q) \quad (3.4)$$

Où  $k_d$  et  $k_p$  sont des matrices diagonales définies positives de dimension  $(n \times n)$  alors, d'après l'équation (3.3), la réponse du système en boucle fermée est décrite par l'équation linéaire découplée suivante :

$$\ddot{e} + k_d \dot{e} + k_p e = 0 \quad (3.5)$$

Avec :  $e = q_d - q$

Les gains  $k_{d_i}$  et  $k_{p_i}$  sont choisis pour imposer à l'erreur de l'axe  $i$  la dynamique désirée d'amortissement  $\xi_i$  et de pulsation  $\omega_i$  quelle que soit la configuration du robot [21] :

$$\begin{cases} k_{d_i} = 2 \xi_i \omega_i \\ k_{p_i} = \omega_i^2 \end{cases} \quad (3.6)$$

### 3.2.1.1.2 Trajectoire définie par la position ( $q_d$ ) :

Dans ce cas, le but à atteindre est la position  $q_d$ . Un choix possible pour  $w(t)$  est de prendre :

$$w = -k_d \dot{q} + k_p(q_d - q) \quad (3.7)$$

On substitue  $w$  dans l'équation (3.2) puis (3.1), on trouve :

$$\ddot{q} + k_d \dot{q} + k_p q = k_p q_d \quad (3.8)$$

Qui est une équation linéaire découplée du deuxième ordre dont la solution  $q(t)$  est globalement exponentiellement stable, si les gains  $k_d$  et  $k_p$  sont bien choisis.

**3.2.1.2 Modèle incertain :**

Si le modèle du robot n'est pas exact et présente des incertitudes de modélisation, l'équation de la boucle fermée est obtenue en utilisant les relations (3.1) et (3.2) :

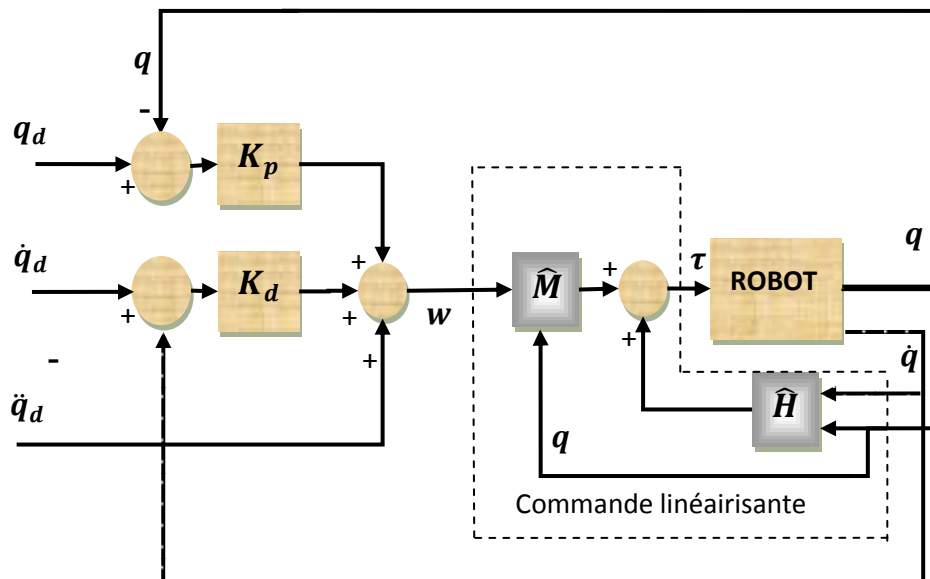
$$\widehat{M} [\ddot{q}_d + k_d \dot{e} + k_p e] + \widehat{H} = M\ddot{q} + H \tag{3.9}$$

Et on en déduit que :

$$\ddot{e} + k_d \dot{e} + k_p e = \widehat{M}^{-1} [(M - \widehat{M})\ddot{q} + H - \widehat{H}] \tag{3.10}$$

On constate dans cette dernière équation que l'erreur due à la modélisation excite la dynamique des erreurs, donc on doit ajuster les paramètres  $k_p$  et  $k_d$  (aux limites de la stabilité du système) de telle façon l'incertitude sera surmontée,

Cette commande est illustrée dans **la figure suivante 3.1:**



**Figure 3.1.** Le principe de la commande par découplage non linéaire

### 3.3 La commande par backstepping :

La technique de backstepping a été développée au début des années 90. L'arrivée de la commande par backstepping a donné un nouveau souffle à la commande des systèmes non linéaires, qui malgré les grands progrès réalisés, il manquait des approches générales.

Cette technique est une méthode systématique et récursive de synthèse de lois de commande non linéaires qui utilise le principe de stabilité par Lyapunov et qui peut s'appliquer à un grand nombre de systèmes non linéaires [14].

Le backstepping est une méthode multi-étapes ; à chaque étape du processus, une commande virtuelle est générée pour assurer la convergence du système vers son état d'équilibre. Cela peut être atteint à partir des fonctions de Lyapunov qui assurent pas à pas la stabilisation de chaque étape de synthèse

la technique du backstepping donne plus de flexibilité au concepteur en exploitant les " bonnes " non linéarités, tandis que les " mauvaises " non linéarités ou les non linéarités déstabilisantes sont dominées en ajoutant des amortissements non linéaires. Ainsi, une robustesse additionnelle est obtenue [26].

#### 3.3.1 La conception de la commande par backstepping :

Pour bien exposer l'algorithme de la commande par backstepping, on considère le système non linéaire de 3<sup>ème</sup> degré de la forme suivante :

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 = f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \\ \dot{x}_3 = f_3(x_1, x_2, x_3) + g_3(x_1, x_2, x_3) u \end{cases} \quad (3.11)$$

D'abord pour appliquer la méthode backstepping il faut formuler, le système sous la forme dite forme paramétrique de type boucle de retour stricte comme dans l'équation précédente.

$g_i(.) \neq 0, f_i(.)$   $i = 1,2$  : sont des fonctions non linéaires continues et dérivables.

$g_3(.) \neq 0, f_3(.)$  : sont des fonctions non linéaires continues.

$u$  : est l'entrée du système.

L'idée principale du backstepping est illustrée par le principe suivant :

$$\left\{ \begin{array}{l} \dot{x}_1 = f_1 + g_1 x_2 \rightarrow \text{Sous systeme1 : trouver } x_2^r \text{ pour que } x_1 - x_{1d} \text{ soit nulle (par Lyapunov)} \\ \dot{x}_2 = f_2 + g_2 x_3 \rightarrow \text{Sous systeme2 : trouver } x_3^r \text{ pour que } x_2 - x_2^r \text{ soit nulle (par Lyapunov)} \\ \dot{x}_3 = f_3 + g_3 u \rightarrow \text{Sous systeme2 : trouver } u \text{ pour que } x_3 - x_3^r \text{ soit nulle (par Lyapunov)} \end{array} \right.$$

**Remarque :**

$$x_2^r \neq x_{2d}$$

$$x_3^r \neq x_{3d}$$

Pour alléger l'algorithme, On procède au changement des variables suivants.

$$\left\{ \begin{array}{l} a_0 = x_{1d} \\ a_1 = x_2^r \\ a_2 = x_3^r \end{array} \right. \quad (3.12)$$

$$\left\{ \begin{array}{l} z_1 = x_1 - a_0 \\ z_2 = x_2 - x_2^r \\ z_3 = x_3 - x_3^r \end{array} \right. \quad (3.13)$$

**Première étape :**

Objectif : trouver  $a_1$  pour que  $z_1$  soit stable.

$z_1$  comme sortie virtuelle du premier sous système  $z_1 = x_1 - a_0$ . Sa dérivée dans le temps est :

$$\dot{z}_1 = \dot{x}_1 - \dot{a}_0$$

$$\dot{z}_1 = f_1 + g_1 x_2 - \dot{a}_0 \quad (3.14)$$

La fonction de Lyapunov partielle de type quadratique :

$$V_1 = \frac{1}{2} z_1^2 \quad (3.15)$$

Sa dérivée par rapport au temps est :

$$\begin{aligned} \dot{V}_1 &= z_1 \dot{z}_1 \\ &= z_1 (f_1 + g_1 x_2 - \dot{a}_0) \end{aligned}$$

Un bon choix de  $x_2$  peut assurer la stabilité ( $\dot{V}_1 < 0$ ) du système (3.11), prenons donc  $x_2^r$  ou  $a_1$  pour que  $\dot{V}_1 = -c z_1^2 < 0$ , avec  $c_1 > 0$  :

Donc nous avons :

$$z_1 (f_1 + g_1 a_1 - \dot{a}_0) = -c z_1^2 \quad (3.16)$$

$$a_1 = \frac{1}{g_1} (-c z_1 - f_1 + \dot{a}_0) \quad (3.17)$$

### Deuxième étape :

Objectif : trouver  $a_2$  pour que  $z_1, z_2$  soit stable.

Nous avons de l'expression (3.13)  $\Rightarrow x_2 = a_1 + z_2$

Dans cette étape nous utilisons la troisième équation de l'expression (3.13) comme une sortie pour le second sous système avec  $x_3$  comme entrée, et réécrivant la seconde équation du système :

La dynamique du système dans L'espace  $z_1, z_2$  :

D'abord :

$$\begin{cases} \dot{z}_1 = f_1 - \dot{a}_0 + g_1 (a_1 + z_2) \\ \dot{z}_2 = f_2 - \dot{a}_1 + g_2 x_3 \end{cases} \quad (3.18)$$

Sur la base de ces équations, on choisit la fonction de Lyapunov comme suite :

$$V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 \quad (3.19)$$

Sa dérivée par rapport au temps est donnée par :

$$\begin{aligned} \dot{V}_2 &= z_1 \dot{z}_1 + z_2 \dot{z}_2 \\ &= z_1 (f_1 + g_1 (a_1 + z_2) - \dot{a}_0) + z_2 (f_2 - \dot{a}_1 + g_2 x_3) \\ &= z_1 (f_1 + g_1 a_1 - \dot{a}_0) + z_2 (f_2 + g_1 z_1 + g_2 x_3 - \dot{a}_1) \\ &= -c_1 z_1^2 + z_2 (f_2 + g_1 z_1 + g_2 x_3 - \dot{a}_1) \end{aligned}$$

Un bon choix de  $x_3$  peut assurer la stabilité ( $\dot{V}_2 < 0$ ) du système (3.11), donc prenons  $x_3^r$  ou  $a_2$  pour que  $\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 < 0$ , avec  $c_1, c_2 > 0$  :



$$\mathbf{a}_2 = \frac{\mathbf{1}}{\mathbf{g}_2} (-\mathbf{c}_2 \mathbf{z}_2 - \mathbf{f}_2 - \mathbf{g}_1 \mathbf{z}_1 + \dot{\mathbf{a}}_1) \quad (3.20)$$

Puisque  $a_1$  est une fonction stabilisante et fonction de  $x_1, x_{1d}, \dot{x}_{1d}$  alors :

$$\dot{a}_1 = \frac{\partial a_1}{\partial x_1} \dot{x}_1 + \frac{\partial a_1}{\partial x_{1d}} \dot{x}_{d1} + \frac{\partial a_1}{\partial \dot{x}_{d1}} \ddot{x}_{d1} \quad (3.21)$$

**Troisième étape :**

Objectif : trouver la commande  $u$  pour que  $z_1, z_2, z_3$  soit stable.

Nous avons de l'expression (3.13)  $\Rightarrow x_3 = a_2 + z_3$

D'abord Construisons La dynamique du système dans L'espace  $z_1, z_2, z_3$  :

$$\begin{cases} \dot{z}_1 = f_1 - \dot{a}_0 + g_1 (a_1 + z_2) \\ \dot{z}_2 = f_2 - \dot{a}_1 + g_2 (a_2 + z_3) \\ \dot{z}_3 = f_3 - \dot{a}_2 + g_3 u \end{cases} \quad (3.22)$$

Sur la base de ces équations on formule la fonction de Lyapunov comme suite :

$$V_3 = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 + \frac{1}{2} z_3^2 \quad (3.23)$$

Sa dérivée par rapport au temps est donnée par :

$$\dot{V}_3 = -c_1 z_1^2 - c_2 z_2^2 + z_3 (g_3 u + g_2 z_2 + f_3 - \dot{a}_2) \quad (3.24)$$

Donc la loi de commande finale stabilisante est donnée par :

$$\mathbf{u} = \frac{\mathbf{1}}{\mathbf{g}_3} (-\mathbf{c}_3 \mathbf{z}_3 - \mathbf{g}_2 \mathbf{z}_2 - \mathbf{f}_3 + \dot{\mathbf{a}}_2) \quad (3.25)$$

Puisque  $a_1$  est une fonction stabilisante en fonction de  $a_1(x_1, x_{1d}, \dot{x}_{1d}, \ddot{x}_{1d})$  alors :

$$\dot{a}_2 = \frac{\partial a_2}{\partial x_1} \dot{x}_1 + \frac{\partial a_2}{\partial x_{1d}} \dot{x}_{d1} + \frac{\partial a_2}{\partial \dot{x}_{1d}} \ddot{x}_{d1} + \frac{\partial a_2}{\partial \ddot{x}_{1d}} x_{1d}^{(3)} \quad (3.26)$$

Avec tous ces étapes on a assuré que :

$$\dot{V}_2 \leq -c_1 z_1^2 - c_2 z_2^2 - c_3 z_3^2 < 0, \text{ avec } c_1, c_2, c_3 > 0$$

D'où la stabilité asymptotique du système (3.22) est assurée. Ceci se traduit par la stabilité, en boucle fermée, du système (3.11) et la régulation à zéro de l'erreur de poursuite.

### 3.3.2 La conception de la commande backstepping pour le manipulateur:

Le modèle dynamique du robot en espace articulaire est donné par :

$$M\ddot{q} + V\dot{q} + G = \tau$$

Le modèle direct est donné par :

$$\ddot{q} = M^{-1}(\tau - V\dot{q} - G)$$

Donc avant d'appliquer la méthode de backstepping on doit reformuler cette dernière équation dans l'espace d'état et en forme paramétrique comme suivant:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \ddot{q} = M^{-1}(\tau - V\dot{q} - G) \end{cases} \quad (3.27)$$

On procède d'abord au changement des variables suivants.

$$\begin{cases} a_0 = x_{1d} \\ a_1 = x_2^r \end{cases} \quad (3.28)$$

$$\begin{cases} z_1 = x_1 - a_0 \\ z_2 = x_2 - x_2^r \end{cases} \quad (3.29)$$

Dans ce cas la synthèse de la loi de commande backstepping se fait en deux étapes :

#### Première étape :

Objectif de cet étape est de trouver  $x_2^r$  pour que l'erreur du 1<sup>er</sup> sous système de l'expression (3.29), soit nulle (stable par Lyapunov)

Nous avons de l'expression (3.29)  $\Rightarrow \dot{z}_1 = \dot{x}_1 - \dot{a}_0 \Rightarrow$

$$\dot{z}_1 = x_2 - \dot{a}_0 \quad (3.30)$$

On définit la 1<sup>ère</sup> fonction de Lyapunov telle que :

$$V_1 = \frac{1}{2} z_1^T z_1 \quad (3.31)$$

Sa dérivée par rapport au temps :

$$\begin{aligned}\dot{V}_1 &= z_1^T \dot{z}_1 \\ \dot{V}_1 &= z_1^T (x_2 - \dot{a}_0)\end{aligned}$$

Nous voulons que :  $\dot{V}_1 = -c_1 z_1^T z_1$  avec  $c_1 > 0$

$$x_2^r = a_1 = \dot{a}_0 - c_1 z_1 \quad (3.32)$$

**Deuxième étape :**

Objectif de cet étape est de trouver le couple  $\tau$  pour que l'erreur de 2<sup>ème</sup> sous système de l'expression (3.29), soit nulle (stable par Lyapunov)

Nous avons de l'expression (3.28) et (3.29):

$$x_2 = z_2 + a_1 \quad (3.33)$$

**On forme la dynamique des erreurs dans l'espace  $z_1$  ,  $z_2$  comme suite :**

De l'équation (3.30), (3.33) et (3.32)

$$\dot{z}_1 = z_2 + \dot{a}_0 - c_1 z_1 - \dot{a}_0$$

Alors :

$$\dot{z}_1 = z_2 - c_1 z_1 \quad (3.34)$$

La dérivée de 2<sup>ème</sup> sous système de l'expression (3.29) donne :

$$\dot{z}_2 = \dot{x}_2 - \dot{a}_1 \quad (3.35)$$

La substitution de (3.27) dans (3.35) et l'utilisation de l'équation (3.34) donne :

$$\begin{aligned}\dot{z}_2 &= \ddot{q} - \dot{a}_1 \\ \dot{z}_2 &= M^{-1}(\tau - V\dot{q} - G) - \ddot{q}_d + c_1 (z_2 - c_1 z_1) \quad (3.36)\end{aligned}$$

**Avec :  $\ddot{q}_d = \ddot{a}_0$**

L'espace  $z_1, z_2$  est composé des équations (3.35) et (3.36)

$$\begin{cases} \dot{z}_1 = z_2 - c_1 z_1 \\ \dot{z}_2 = M^{-1}(\tau - V\dot{q} - G) - \ddot{q}_d + c_1 (z_2 - c_1 z_1) \end{cases} \quad (3.37)$$

Sur la base de cet espace des erreurs, on définit la 2<sup>ème</sup> fonction de Lyapunov telle que :

$$V_2 = \frac{1}{2} z_1^T z_1 + \frac{1}{2} z_2^T z_2 \quad (3.38)$$

Sa dérivée par rapport au temps :

$$\dot{V}_2 = -c_1 z_1^T z_1 + z_2^T (M^{-1}(\tau - V\dot{q} - G) - \ddot{q}_d + z_1 + c_1 (z_2 - c_1 z_1))$$

Pour vérifier  $\dot{V}_2 < -c_1 z_1^T z_1 - c_2 z_2^T z_2$ , avec  $c_1, c_2 > 0$

Donc la commande est donnée par :

$$\tau = M \left[ \ddot{q}_d - z_1 - c_1 (z_2 - c_1 z_1) \right] + V\dot{q} + G \quad (3.39)$$

### 3.4 La commande par mode glissant :

Les systèmes à structure variables sont évolués suite aux travaux d'Emel'yano et Barbashin au début de années soixante en Russie, cette technique est devenue publique en dehors de la Russie au milieu des années soixante dix, par Utkin et Itkis Angleterre [10].

La commande par mode glissant est un type particulier de la commande à structure variable dont la commande change sa structure pour répondre à des exigences prédéfinies de contrôle, robustesse et précision en présence des perturbations. Les incertitudes entre le modèle mathématique et le système réel vont fortement affecter les commandes classiques qui peuvent même déstabiliser le système.

La commande par mode glissant est conçue pour faire face à ce problème, cette commande est destinée principalement aux systèmes non linéaires à dynamique incertain.

Sa robustesse vient du fait que la dynamique du système est projetée sur une surface (S) insensible aux perturbations et qui englobe une dynamique désirée définie a priori, l'objectif

de la loi de commande ici est **d'assurer la convergence du système à la surface en temps fini et d'y rester**, une fois la surface de glissement (S) est atteinte, la dynamique du système reste **insensible aux perturbations**, c'est pourquoi cette commande est connue par sa robustesse vis-à-vis des incertitudes paramétriques ainsi que des perturbations internes ou externes. La commande par mode de glissement a été appliquée avec succès aux robots manipulateurs, véhicules sous-marins, transmissions automobiles, des moteurs électriques et des systèmes de puissance [31].

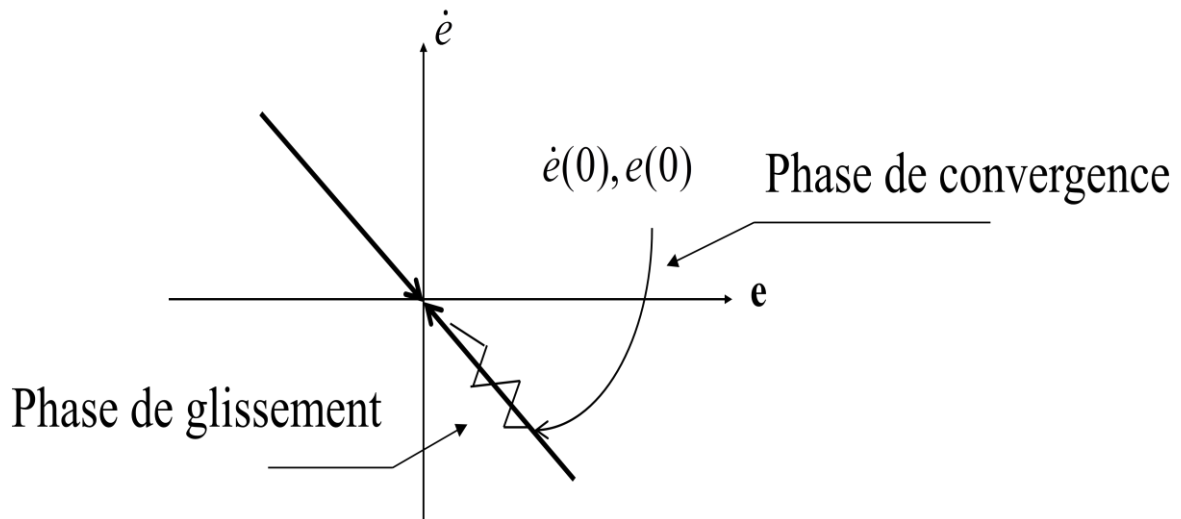
L'inconvénient principal de cette commande est le chattering (réticence), il existe différentes méthodes pour l'atténuer à l'instar du changement de surface de glissement par autre quasi continue où le glissement se fait d'une manière douce, une autre méthode c'est le mode glissant d'ordre 2 SOSM basée sur le déplacement des commutations haute fréquence derrière la 1<sup>ère</sup> dérivée de la surface.

### 3.4.1 Principe de fonctionnement :

Le principe de la commande par mode glissant est basé sur la création d'une dynamique autonome en fonction de l'erreur (entre les états du système  $x$  à commander et leurs états désirés  $x_d$  qui répond aux exigences de robustesse, de stabilité et de précision  $f(e, \dot{e} \dots \rho_1 \dots) = 0$ , la solution de cette équation différentielle conduit l'erreur à s'annuler ( $e = 0$ ), pour un choix adéquat des paramètres.

On fait ensuite évaluer cette dynamique à une autre variable (fonction) qui s'appelle une surface de glissement  $s(t)$  donc quand  $s = 0$  on revient à la dynamique désirée  $f = 0$  et donc l'erreur tend vers ( $e = 0$ ),

A cet étape, on fait appel à la fonction de Lyapunov pour trouver la commande conduisant à la convergence des états du système vers cette surface de glissement  $s = 0$ , La figure 1-I ci-après schématise ce processus dans le plan de phase,



**Figure 3.2.** Schématisation du mode de glissement dans le plan de phase.

En résumé :

L'approche de La commande par mode glissant comprend deux étapes [10]:

**First.** Concevoir une surface de glissement en sorte que sa dynamique satisfait certaine spécifications.

**Second.** La sélection d'une loi de commande pour que la surface soit attractive.

**Remarque 1.III** la loi de commande n'est pas nécessairement discontinue.

### 3.4.2 Choix de la surface de glissement :

Soit le système non linéaire et affine (linéaire par rapport l'entrée u) suivant :

$$\begin{aligned} \dot{x} &= f(x) + g(x)u & (3.40) \\ u &\in \mathcal{R} \quad x = (x_1, x_2 \dots x_n) \in \mathcal{R}^n \end{aligned}$$

$x$  : Vecteur d'état

$u$  : L'entrée du système (la commande).

$f(x), g(x)$  : Des champs de vecteurs suffisamment différentiables, ne sont pas bien connait mais limités et bornés [31] [38].

Le problème de la commande est d'obliger le vecteur d'état de suivre un vecteur d'état spécifique (désiré) varié dans le temps  $x_d = (x_{1d}, x_{2d} \dots x_{nd})$  en présence des incertitudes dans  $f(x), g(x)$ .

La surface de glissement est formulée comme suite selon Slotine [31] :

$$s(t) = \left( \frac{d}{dt} + \lambda_i \right)^{r-1} e(t) \quad (3.41)$$

$e(t) = x_d - x$  ,  $x \in \mathcal{R}^n$  : Vecteur d'état ,  $x_d$  : Valeur désirée de  $x$

$s(t)$  : Surface de glissement.

$\lambda_i$  : Constance d'ajustement strictement positive

$r$  : Degrés relative du système

### Remarque 2.III

Cette reformulation de la surface est en fonction du degré relatif du système, c'est pour faire apparaître **la commande u** dans la formule de la **1<sup>ère</sup> dérivée de la surface**, quelque soit le degré relatif du système à commander.

Pour un système de degré relatif 2:

$$\dot{e} + \lambda e = 0$$

Pour un système de degré relatif 3:

$$\ddot{e} + 2\lambda \dot{e} + \lambda^2 e = 0$$

Ces équations différentielles sont stable pour  $\lambda > 0$ ,

La solution générale de la 1<sup>ère</sup> équation ( $\dot{e} + \lambda e = 0$ ) est :

$$e = e(0)e^{-\lambda t}$$

Sa dérivée est donnée par

$$\dot{e} = -\lambda e(0)e^{-\lambda t}$$

Donc  $e$  et  $\dot{e}$  Convergent asymptotiquement vers 0

### 3.4.3 Condition de convergence :

Donc pour que les erreurs de poursuite  $e$  et  $\dot{e}$  s'annulent asymptotiquement en cas de la présence des perturbations, on fait conduire la surface  $s(t)$  vers 0 en temps fini par les moyens du contrôle et de la commande [38].

En définissant la fonction de Lyapunov (scalaire défini positif) comme suite :

$$V = \frac{1}{2}s^2 \quad (3.42)$$

Et on cherche une commande qui décroît la dérivée de la fonction de Lyapunov et on conséquence garantit la stabilité de la fonction (la dynamique de  $s$ ), Donc la dérivée de  $V$  est donnée par :

$$\frac{dV}{dt} = s\dot{s} \quad (3.43)$$

Pour que le carré de la distance  $s^2$  par rapport la surface de commutation associée diminue Continuellement dans le temps en se rapprochant vers  $S=0$ , et contraindre les états du système à commuter autour de cette surface [31], il faut que :

$$s\dot{s} < 0 \quad (3.44)$$

Cette condition Est appelée **une condition d'attractivité (reaching condition)**, Dans le but d'accomplir la convergence en temps fini (stabilité global à temps fini), la condition (3.44) est modifiée [38] pour être :

$$s\dot{s} \leq -\eta|s| \quad , \quad \eta > 0 \quad (3.45)$$

Ou bien  $\dot{V} \leq -\eta V^{1/2}$

Telque :  $V^{1/2} = |s|$

Généralement, pour que la condition  $s\dot{s} < -\eta|s|$  soit vérifiée en quelque soit le temps, le choix évident de  $\dot{s}$  se fait comme suit :

$$\dot{s} = -k \operatorname{sign}(s) \quad k > 0 \quad (3.46)$$

Cette loi est appelée **une loi d'attractivité** (reaching law)



### 3.4.4 Calcul du temps d'attractivité :

Supposant que  $s(t = 0) > 0$  et  $t_{reach}$  le temps nécessaire pour aboutir à la surface  $s = 0$

En intégrant l'équation  $s\dot{s} \leq -\eta|s|$  entre l'intervalle  $0 < \tau < t_r$  ,  $t_r$  le temps d'atteindre ;

On trouve :

$$s(t_r) - s(0) \leq -\eta t_r \quad \text{À } t = t_r \quad s(t_r) = 0$$

Donc Le temps est limité à  $t_r \leq \frac{s(0)}{\eta}$

Donc la commande calculée pour satisfaire  $s\dot{s} \leq -\eta|s|$  , va conduire la surface à s'annuler en temps finit et la maintenir [38].

#### Régime glissant :

$$\dot{x} = f(x) + g(x)u \quad (3.47)$$

$$u \in \mathcal{R} \quad x = (x_1, x_2 \dots x_n) \in \mathcal{R}^n$$

$s\dot{s} < 0$  Représente l'attractivité de la surface de glissement, ce qui peut être traduit mathématiquement par

$$\begin{cases} \lim_{s \rightarrow 0^+} \dot{s} < 0 \\ \lim_{s \rightarrow 0^-} \dot{s} > 0 \end{cases} \Rightarrow \begin{cases} \lim_{s \rightarrow 0^+} \frac{\partial s}{\partial x} \dot{x} < 0 \\ \lim_{s \rightarrow 0^-} \frac{\partial s}{\partial x} \dot{x} > 0 \end{cases}$$

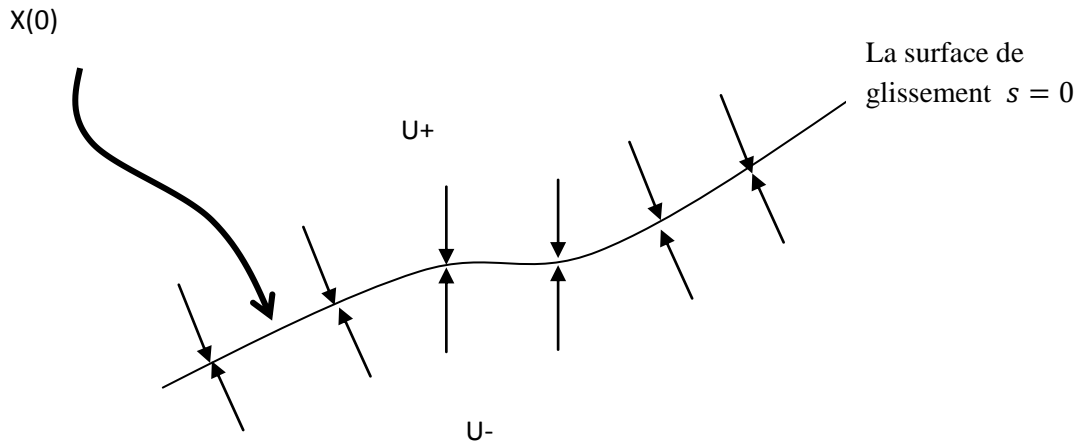
Avec la commutation de la commande à structure variable u est donnée par :

$$u = \begin{cases} u^+ & \text{si } s < 0 \\ u^- & \text{si } s > 0 \end{cases} \quad u^+ \neq u^- \quad (3.48)$$

Nous obtenient :

$$\begin{cases} \lim_{s \rightarrow 0^+} \frac{\partial s}{\partial x} (f + gu^-) < 0 \\ \lim_{s \rightarrow 0^-} \frac{\partial s}{\partial x} (f + gu^+) > 0 \end{cases} \quad (3.49)$$

Ces conditions expriment les projections des champs de vecteurs  $(f + gu^+)$  et  $(f + gu^-)$  sur le gradient de la surface  $s$ .



**Figure 3.3.** L'attraction de la surface de glissement.

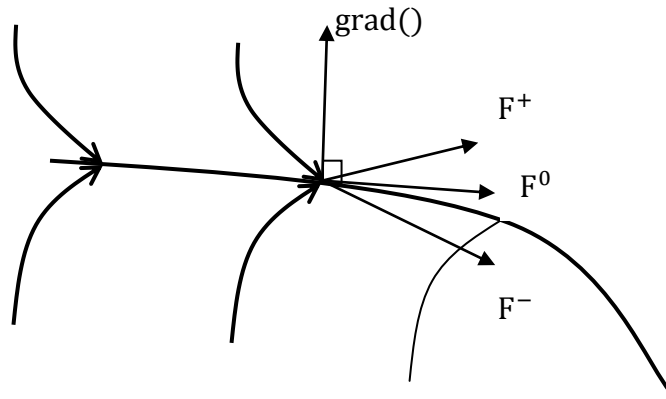
### 3.4.5 La commande équivalente dans le sens de Filippov :

VSS engendre des discontinuités due à la commutation de la commande, la théorie conventionnelle des équations différentielles est insuffisante, et ne permet pas de décrire le comportement des solutions dans ce cas là, l'un des premières méthodes adresser à ce type de solution est la méthode de Filippov,

Filippov a montré que la trajectoire d'état de l'équation (3.47) Avec la commande (3.48) et dans la surface de glissement  $s=0$  résout l'équation suivante :

$$\dot{x} = a F^+ + (1 - a)F^- = F^0 \quad 0 \leq a \leq 1 \quad (3.50)$$

La figure ci-après illustre cette méthode :



**Figure 3.4.** Illustration de la méthode de Filippov.

Où  $F^+ = F(x, t, u^+)$  ,  $F^- = F(x, t, u^-)$  et  $F^0$  le vecteur vitesse résultant de la trajectoire d'état dans le régime glissant

Le problème est de trouver  $a$  qui vérifie  $\langle grad(s), F^0 \rangle = 0$  i.e

$$a = \frac{\langle grad(s), F^- \rangle}{\langle grad(s), (F^- - F^+) \rangle} \quad (3.51)$$

**Le calcul de  $F^0$**  représente la vitesse moyenne  $\dot{x}$  , de la trajectoire d'état contraint à la surface de glissement.

### 3.4.6 La commande équivalente :

Considérons le système suivant :

$$\dot{x} = f(x) + g(x) u + d = f(x, u, d) \quad (3.52)$$

$x \in \mathcal{R}^n$  : Vecteur d'état

$u \in \mathcal{R}$  : La commande

$d \in \mathcal{R}^q$  : La perturbation

$f(x), g(x)$  : Champs de vecteur

Considérant aussi S dans l'espace d'état

$$\mathbf{s} = \{x : s(x) = 0\}$$

**Définition III.2 :** on dit régime glissant idéal si les états du système évoluent dans le temps avec  $s(x(t_r)) = 0$  à un temps fini  $t_r \in \mathbb{R}^+$  et  $s(x(t)) = 0$  pour tout  $t > t_r$ .

La méthode de la commande équivalente est attribuée à Utkin [38], c'est la commande nécessaire pour maintenir un mouvement de glissement idéal sur  $\mathbf{s}$ , l'expression de  $\dot{\mathbf{s}}$  peut s'écrire :

$$\dot{s} = \frac{\partial s}{\partial x} \dot{x} = \frac{\partial s}{\partial x} f(x, u, d) = 0$$

Donc  $u_{eq}$  est la solution de cette dernière équation algébrique :

$$\begin{aligned} \frac{\partial s}{\partial x} f(x, u_{eq}, d) &= 0 \\ \frac{\partial s}{\partial x} f(x) + \frac{\partial s}{\partial x} g(x) u_{eq} + \frac{\partial s}{\partial x} d &= 0 \end{aligned}$$

$\frac{\partial s}{\partial x} g(x) \neq 0$  donc :

$$u_{eq} = - \left( \frac{\partial s}{\partial x} g(x) \right)^{-1} \frac{\partial s}{\partial x} f(x) - \left( \frac{\partial s}{\partial x} g(x) \right)^{-1} \frac{\partial s}{\partial x} d \quad (3.53)$$

La réponse en boucle fermée est obtenue en substituant cette dernière expression dans notre system dynamique, on trouve :

$$\dot{x} = \left( I - g(x) \left( \frac{\partial s}{\partial x} g(x) \right)^{-1} \frac{\partial s}{\partial x} \right) f(x) + \left( I - g(x) \left( \frac{\partial s}{\partial x} g(x) \right)^{-1} \frac{\partial s}{\partial x} \right) d$$

### Remarque III.3

L'expression de  $u_{eq}$  c'est simplement la solution algébrique de l'équation suivante :

$$\frac{\partial s}{\partial x} f(x, u_{eq}, d) = 0 \quad (3.54)$$

Donc pratiquement on est loin de loi de commande réelle.

### Remarque III.4

L'expression de  $u_{eq}$  dépend de l'expression de la perturbation  $d$  qui est pratiquement inaccessible.

### 3.4.7 Constitution de la loi de commande :

On considère pour cette section le système d'ordre 2 suivant :

$$\ddot{x} = f(x, t) + u \quad (3.55)$$

La commande par mode glissant est constituée de deux éléments principaux l'un, appelé la commande discontinue, qui assure la convergence vers la surface de glissement en temps fini, cette partie est très utile pour compenser les incertitudes du système, tandis que la deuxième c'est la commande équivalente qui sert à maintenir le système (leur états) dans la surface de glissement donc on peut écrire :

$$u = u_{eq} + u_{dis} \quad (3.56)$$

$$\dot{s} = \ddot{e} + \lambda \dot{e} = \ddot{x} - \ddot{x}_d + \lambda \dot{e} = f(x, t) + u - \ddot{x}_d + \lambda \dot{e} \quad (3.57)$$

Et d'après l'expression (3.46) nous obtenons la commande comme suite :

$$-k \operatorname{sign}(s) = f(x, t) + u - \ddot{x}_d + \lambda \dot{e} \quad (3.58)$$

$$\Rightarrow u = \ddot{x}_d - f(x, t) - \lambda \dot{e} - k \operatorname{sign}(s) = u_{eq} + u_{dis} \quad (3.59)$$

$$\begin{cases} u_{eq} = \ddot{x}_d - f(x, t) - \lambda \dot{e} \\ u_{dis} = -k \operatorname{sign}(s) \end{cases} \quad (3.60)$$

L'interprétation du résultat lorsque on est dans la phase transitoire (attractivité  $t < t_r$ ) la commande est constituée de la commande équivalente dont son rôle ici est de prépositionner les états du système à leurs états désirés, et une autre partie de nature discontinue dont son rôle dans cette phase est primordial qui permet d'accéder à la surface en temps fini tout en surmontant les incertitudes et les perturbations,

Et lorsque la surface est atteinte donc le terme  $-k \text{sign}(s)$  s'annule ou proche de zéro (ne s'annule jamais réellement) dans cette phase la commande équivalente qui persiste dont son rôle ici est de garder le vecteur de la dynamique du système (le vecteur d'états) sur la surface de glissement, le rôle de la commande équivalente peut percer comme compensateur des non linéarités introduites par le terme  $f$ .

### 3.4.8 La robustesse :

Quoi qu'il en soit, la robustesse de cette commande reste un atout majeur et c'est également  $u_{dis}$  qui assure cette propriété.

En effet, imaginons à titre d'exemple, que la fonction non linéaire  $f$  du système (3.55) n'est pas rigoureusement connue. Soit alors  $\hat{f}$  estimée de  $f$ . Nous supposons que  $f$  ainsi que son estimée sont bornées dans le temps. La loi de commande (3.59) devient alors :

$$\mathbf{u} = \ddot{x}_d - \hat{f}(x, t) - \lambda \dot{e} - k \text{sign}(s) \quad (3.61)$$

En substituant cette loi de commande dans l'expression (3.57), nous obtenons, la loi d'attractivité suivante:

$$\dot{s} = (f - \hat{f}) - k \text{sign}(s) \quad (3.62)$$

Ainsi, pour que la condition d'attractivité (3.46) soit encore vérifiée, il suffirait, d'après (3.62) que la condition suivante soit respectée :

$$k > |f - \hat{f}| \quad (3.63)$$

D'après (3.63), nous constatons qu'en choisissant le gain discontinu  $k$  suffisamment grand, L'incertitude sur  $f$  sera absorbée par le terme discontinu de la loi de commande. Ainsi, nous constatons qu'une augmentation du gain discontinu assure une plus grande robustesse du contrôleur, mais provoque le phénomène de commutations sur la commande.

### 3.4.9 Le phénomène de réticence (Chattering) :

Un régime glissant idéal nécessite une commande de fréquence de commutation infinie. Evidemment, pour une utilisation pratique, seule une commutation à une fréquence finie est possible.

Les phénomènes de réticence peuvent être si pénalisants que l'utilisation d'une loi de commande par modes glissants peut, dans certaines applications être à proscrire, vu que son

utilisation peut dégrader les performances, voire conduire à l'instabilité à cause de la réticence sur la sortie. La réticence de la commande, peut entraîner une usure prématurée des actionneurs ou de certaines parties du système à cause de fortes sollicitations. En excitant les modes propres des dynamiques non modélisées ou des fréquences de résonance du système correspondant aux retards de commutation, cette commande peut provoquer sur les systèmes mécaniques un bruit et des oscillations préjudiciables à leur structure. Sur des systèmes autres que mécaniques, les oscillations engendrées peuvent poser d'autres problèmes (réduction de précision, créations d'ondes électromagnétiques néfastes ou autres ondes amplifiées par le système, . . .).

### 3.4.10 Solutions pour la réticence:

Afin de réduire ou d'éliminer le phénomène de la réticence, de nombreuses techniques ont été proposées. Parmi les techniques les plus utilisées on cite :

#### 3.4.10.1 La couche limite (boundary layer) :

Elle consiste à substituer la fonction *signe* par une approximation à gain élevé au voisinage de  $S$ , et faire saturer la fonction à l'extérieur [34] [31]. Dans ce cas le régime glissant n'est plus dans  $S$ , mais au voisinage de celui-ci. Ce qui entraîne un régime pseudo-glissant. Cependant, cette méthode engendre un compromis entre le niveau de réduction de chattering et la conservation de robustesse. Parmi les fonctions utilisées on cite : la fonction saturation, la fonction *signe* + saturation et tangente hyperbolique.

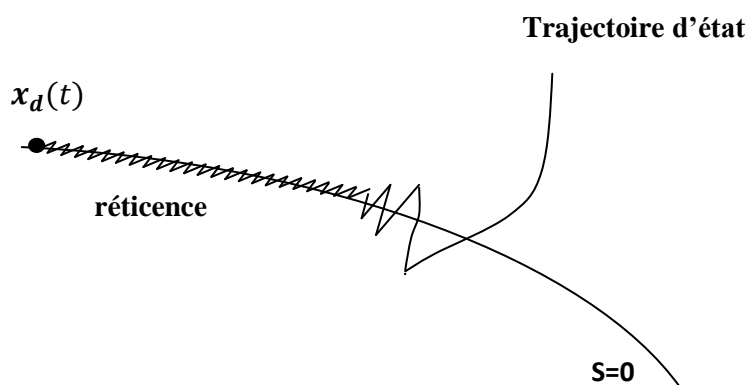


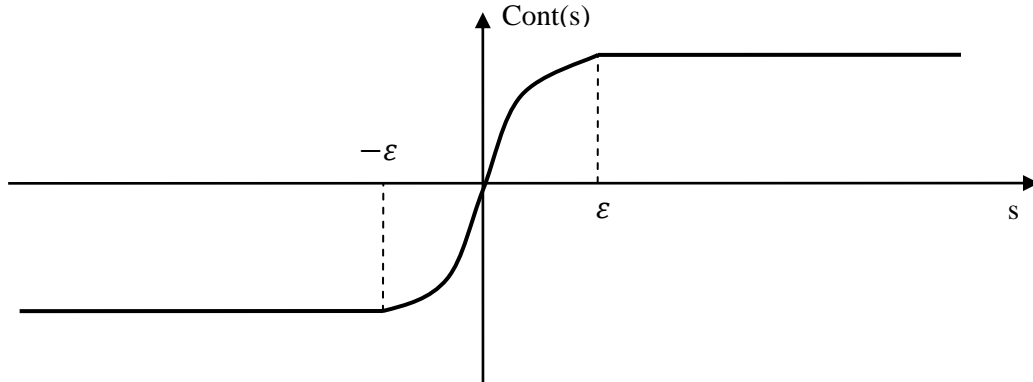
Figure 3.5. Le phénomène de réticence.

#### 3.4.10.2 Pseudo mode glissant (quasi sliding mode) :

La fonction *signe*, dans le voisinage de la surface de glissement, est remplacée par une fonction appelée *cont(.)*, dont:

$$\text{sign}(s) \approx \frac{s}{|s| + \varepsilon}$$

Ou  $\varepsilon$  est scalaire positif, on peut remarquer :  $\lim_{\varepsilon \rightarrow 0} \frac{s}{|s| + \varepsilon} = \text{sign}(s)$

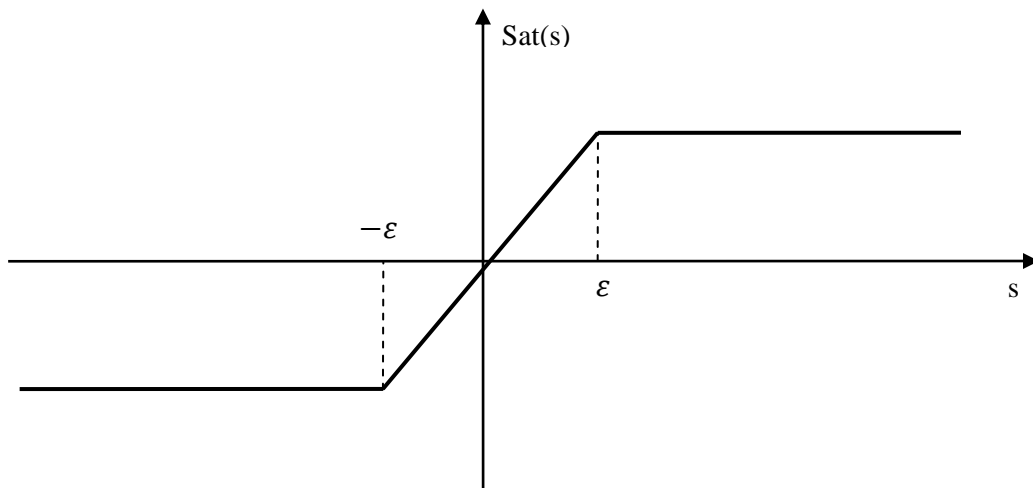


**Figure 3.6.** La fonction Cont(.)

### 3.4.10.3 La fonction de saturation :

Cela consiste à remplacer la fonction  $\text{sign}(s)$  par la droite de pente  $\frac{1}{\varepsilon}$  à l'intérieur d'une bande de largeur  $2\varepsilon$  située de part et d'autre de la surface de glissement, la discontinuité étant conservée à l'extérieur de cette bande. Son expression est donnée par :

$$\text{sat}(s, \varepsilon) = \begin{cases} \text{sign}(s) & \text{si } |s| > \varepsilon \\ \frac{s}{\varepsilon} & \text{si } |s| \leq \varepsilon \end{cases} \quad (3.64)$$



**Figure 3.7.** La fonction Sat(.)



#### 3.4.10.4 *Solution par des lois d'approche :*

Cette approche intéressante consiste à concevoir des lois de convergence non linéaires qui permettent une adaptation dynamique d'une composante de la commande en fonction de la variation de la fonction de glissement. Ainsi, plus le vecteur d'état est loin de la surface de glissement, plus cette l'attraction est importante et tend à ramener le vecteur vers la surface.

Par conséquent, il est théoriquement possible de réduire les commutations haute fréquence en régime établi, sans toutefois affecter le temps de convergence ni l'erreur de poursuite du système. Gao et Hung ont proposé notamment deux lois d'atteinte possibles pour réduire les commutations [38].

La première loi d'atteinte contient un terme proportionnel à la fonction de glissement, cette loi d'atteinte est donnée par :

$$\dot{s} = -a \operatorname{sign}(s) - Q s$$

Dans la deuxième loi de convergence est donnée, comme suite :

$$\dot{s} = -a |s|^\rho \operatorname{sign}(s) \quad 0 < \rho < 1 \quad (3.65)$$

Il est démontré par Gao et Hung que la loi de convergence (3.65) provoque un temps d'atteinte fini de la surface de glissement.

#### 3.4.10.5 *Mode glissant asymptotique :*

L'idée est de construire une commande mode glissant en terme de la dérivée de la fonction de contrôle, dans ce cas là, la commande actuelle est devenue continue qui représente l'intégrale de la fonction haute fréquence, cette approche est appelée ‘ ’ d'atténuation de chattering ‘ ’ vu que certain périodique résiduelle est observé après l'intégration [38].

Pour simplification on propose le système de 2<sup>ème</sup> ordre suivant :

$$\begin{cases} \ddot{x} = f(x, t) + u \\ \dot{u} = v \end{cases} \quad (3.66)$$

Supposant que  $f(x, t) \leq L$  et  $\dot{f}(x, t) \leq \bar{L}$

Pour arriver à atténuer le chattering , on propose la surface auxiliaire  $\sigma$  telque :

$$\sigma = \dot{s} + \bar{c} s \quad (3.67)$$

Avec

$$s = \dot{x} + cx \quad (3.68)$$

Donc on doit assurer la convergence de  $\sigma$ , et donc  $s$ , et par conséquent  $x, \dot{x} \rightarrow 0$

Si on construit une loi de commande  $v$  qui assure une convergence dans un temps fini  $s \rightarrow 0$

Donc un mode glissant idéal est installée dans la surface de glissement :

$$\sigma = \dot{s} + \bar{c}s = 0$$

Par l'utilisation de la condition de convergence (3.45) pour construire une commande mode glissant

$$\sigma \dot{\sigma} = \sigma(\ddot{s} + \bar{c}\dot{s})$$

$$\sigma \dot{\sigma} = \sigma(\ddot{x} + c\dot{x} + \bar{c}\dot{x} + c\bar{c}\dot{x})$$

$$\sigma \dot{\sigma} = \sigma(\dot{u} + \dot{f} + cu + cf + \bar{c}u + \bar{c}f + c\bar{c}\dot{x})$$

$$\sigma \dot{\sigma} = \sigma(v + \dot{f} + (c + \bar{c})f + (c + \bar{c})u + c\bar{c}\dot{x})$$

On choisi :  $v = -c\bar{c}\dot{x} - (c + \bar{c})u + v_1$  par ce choix on compense les éléments observables et un choix adéquat de  $v_1$ , notre système devient stable selon Lyapunov.

$$\sigma \dot{\sigma} \leq \sigma(v_1 + \dot{f} + (c + \bar{c})f) \leq \sigma v_1 + |\sigma|(\bar{L} + (c + \bar{c})L)$$

Pour vérifier cette inégalité il suffit que  $v_1$  doit être de signe opposé à  $\sigma$ , on admet le choix suivant :

$$v_1 = -\rho \text{sign}(\sigma) \quad \rho \geq 0 \quad \text{Donc}$$

$$\sigma \dot{\sigma} \leq |\sigma|(-\rho + (\bar{L} + (c + \bar{c})L)) = \frac{-\alpha}{\sqrt{2}} |\sigma| \quad \sigma \text{sign}(\sigma) = |\sigma|$$

$$\rho = (\bar{L} + (c + \bar{c})L) + \frac{\alpha}{\sqrt{2}} \quad (3.69)$$

La loi de commande  $v$  qui conduit  $\sigma$  à zero dans un temps fini  $t_r \leq \frac{\sqrt{2} \sigma(0)}{\alpha}$  est donnée par:

$$v = -c\bar{c}\dot{x} - (c + \bar{c})u + -\rho \text{sign}(\sigma) \quad (3.70)$$

#### **3.4.10.6 Mode glissant d'ordre 2 :**

Le but de ce type de contrôleurs est de générer un régime glissant du second ordre sur une surface  $\sigma$  par l'annulation de  $\sigma$  ainsi que de sa dérivée  $\dot{\sigma}$  en un temps fini  $\sigma = \dot{\sigma} = 0$ . La caractéristique principale de cette stratégie est que la partie discontinue apparaît sur la dérivée de la commande  $\dot{\mathbf{u}}$ , finalement lorsque l'on fait le calcul de la commande du système  $\mathbf{v} = \int \dot{\mathbf{u}}$ , elle devient continue limitant ainsi le phénomène de réticence.

Il existe aussi le mode glissant d'ordre supérieur, plusieurs types sont exposés dans la littérature qui sont tous cernés l'effet de chattering, ainsi que l'arrivée en temps fini à la surface de glissement [38].

Voir annexe A

### **3.5 Conclusion :**

Nous avons traité dans ce chapitre les commandes non linéaires qui sont très répandues à l'instar de la commande par découplage non linéaire où la dynamique du système est primordiale pour la synthèse de cette commande qui est largement utilisée dans la littérature, en deuxième partie on a abordé la commande par backstepping, et nous l'avons remarquée que c'est une commande systématique dans sa mise en œuvre par rapport à autre commande qui nécessite une large expérience dans le domaine de contrôle, le backstepping est basé sur la fonction de Lyapunov dans la préparation de la commande stabilisante finale, dans chaque étape une fonction de Lyapunov est formulée pour stabiliser un sous système et après plusieurs étapes on arrive à la commande finale stabilisante pour tous les sous systèmes. Enfin dans la dernière partie on a exposé le concept de base de la commande mode glissant ainsi que le choix de la surface de glissement, tout en décrivant les différentes étapes pour la conception de la commande par mode glissant. Cette dernière méthode est caractérisée par sa robustesse vis-à-vis les incertitudes paramétriques où due aux perturbations externes.

Ensuite nous avons souligné le problème majeur de cette commande qui est la réticence (chattering), pour l'éliminer, ou l'atténuer, nous avons présenté quelques solutions pour y remédier.

# Chapitre-IV-

## Application des commandes non linéaires au robot manipulateur en espace contraint.

### 4.1 Le scénario envisagé :

Le schéma ci-dessous illustre le scénario proposé pour appliquer la commande en impédance, le robot manipulateur avant l'interaction avec l'environnement (espace non contraint) est asservi en position afin de suivre la trajectoire désirée, et au point de contact le manipulateur doit garder la force  $F_e$  (force exercée par l'effecteur terminal du robot sur l'environnement) acceptable et la position du robot stable. Tout ça est assuré par la commande en force que ce soit en impédance ou hybride

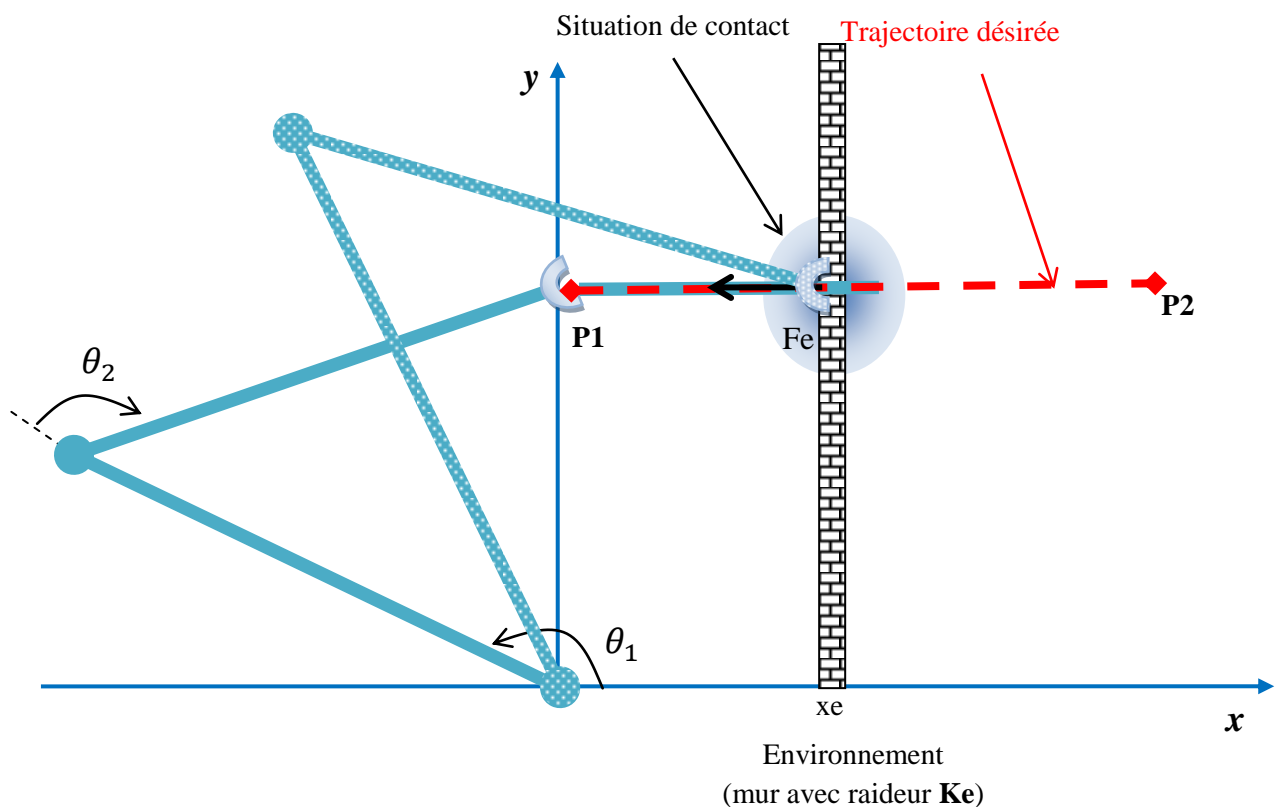


Figure 4.1. Scénario d'interaction du robot avec son environnement.

La trajectoire désirée est générée dans l'espace cartésien, puisque l'impédance désirée est décrite dans cet espace.

Afin de générer la trajectoire dans tel espace on doit s'assurer d'abord que le chemin retenu soit à l'intérieur de l'espace de travail du robot manipulateur,

L'espace de travail est lié principalement aux dimensions des liaisons (links) ( $L_1, L_2$ ) et aux butées angulaires du robot ( $\theta_1, \theta_2$ ), il est maximal lorsque  $L_1 = L_2$  et  $0^\circ < \theta_1, \theta_2 < 360^\circ$ . [33].

Notre cas consiste à générer une ligne droite du point  $P_1(0, 1)$  au  $P_2(0, 0.5)$ , on utilise un polynôme cubique de degrés **4** pour décrire l'évolution du robot dans l'axe  $x$  par rapport au temps (et  $y=1$ , constante).

$$x(t) = a_0 + a_1t + a_2t^2 + a_3t^3 \quad (4.1)$$

Avec :

$$\left\{ \begin{array}{l} x(0) = 0 \quad , \text{ Position initiale} \\ x(10) = 1 \quad , \text{ Position finale} \\ \dot{x}(0) = 0 \quad , \text{ Vitesse initiale} \\ \dot{x}(10) = 0 \quad , \text{ Vitesse finale} \end{array} \right.$$

Les coefficients de polynôme sont :

$$a_0 = 0$$

$$a_1 = 0$$

$$a_2 = 15/1000$$

$$a_3 = -1/1000$$

Donc la trajectoire dans l'espace cartésien est défini par :

$$\left\{ \begin{array}{l} x(t) = \left(\frac{15}{1000}\right)t^2 - \left(\frac{1}{1000}\right)t^3 \\ y(t) = 1 \end{array} \right. \quad (4.2)$$

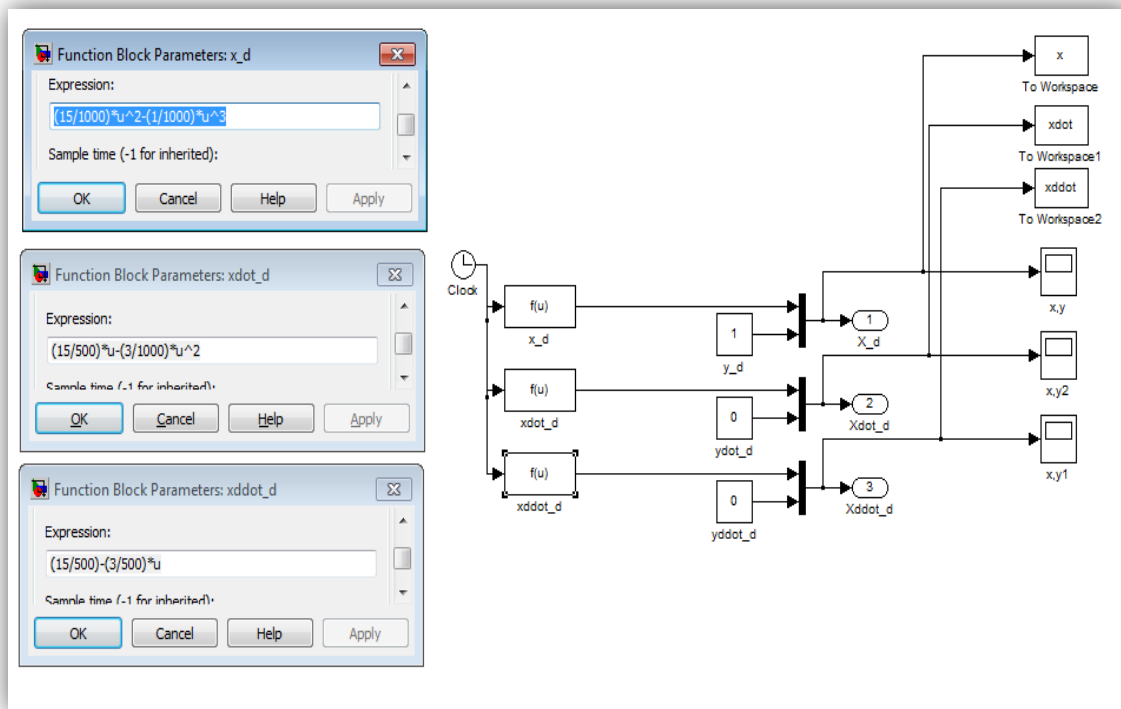


Figure 4.2. Block Simulink des trajectoires désirées.

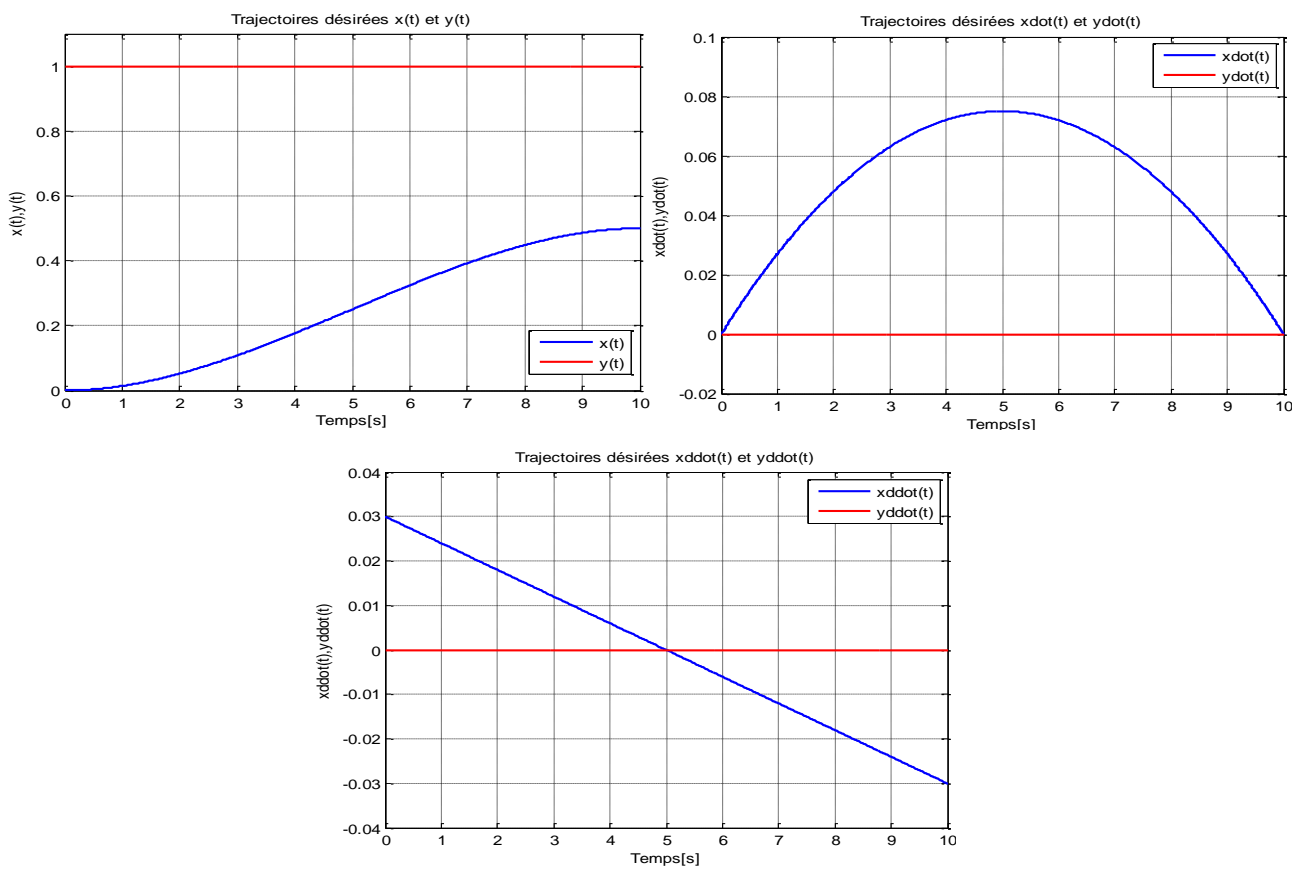


Figure 4. 3. Les trajectoires désirées dans l'espace cartésien.

## 4.2 La commande hybride Force/Position :

Dans cette section on adopte la stratégie de commande de la section §2.3.1.1 qui est une Commande Hybride Force/Position avec Somme des Couples Articulaires, et avec suppression de la matrice de sélection de la boucle de commande en position, c'est-à-dire la commande en position est activée dans toutes les directions, ainsi que la commande en force n'est activée qu'après la détection du contact dans les directions contraintes, donc les directions commandées en effort vont aussi commander en position, ceci est réalisé par une simple superposition des commandes issues des deux boucles grâce au complément de la matrice de sélection ( $I - S$ ).

Le modèle dynamique du robot en contact avec l'environnement est donné par l'équation (2.24) du chapitre II, l'accélération est décrite dans l'espace cartésien.

$$M J^{-1}(\ddot{X} - j \dot{q}) + V_m \dot{q} + G + J^T F_e = \tau \quad (4.3)$$

En utilisant la dynamique du robot on peut conclure La loi de commande en position de la forme suivante :

$$\tau_p = \hat{M} J^{-1}(Wp - j \dot{q}) + \hat{V}_m \dot{q} + \hat{G} + J^T F_e \quad (4.4)$$

Avec :

$$Wp = \ddot{X}_d + K_v (\dot{X}_d - \dot{X}) + K_p (X_d - X) \quad (4.5)$$

Quant à la loi de commande en effort, elle est alors la suivante :

$$\tau_f = J^T (I - S) \left[ K_f (F_d - F) - K_{fv} \dot{X} + K_{fl} \int (F_d - F) \right] \quad (4.6)$$

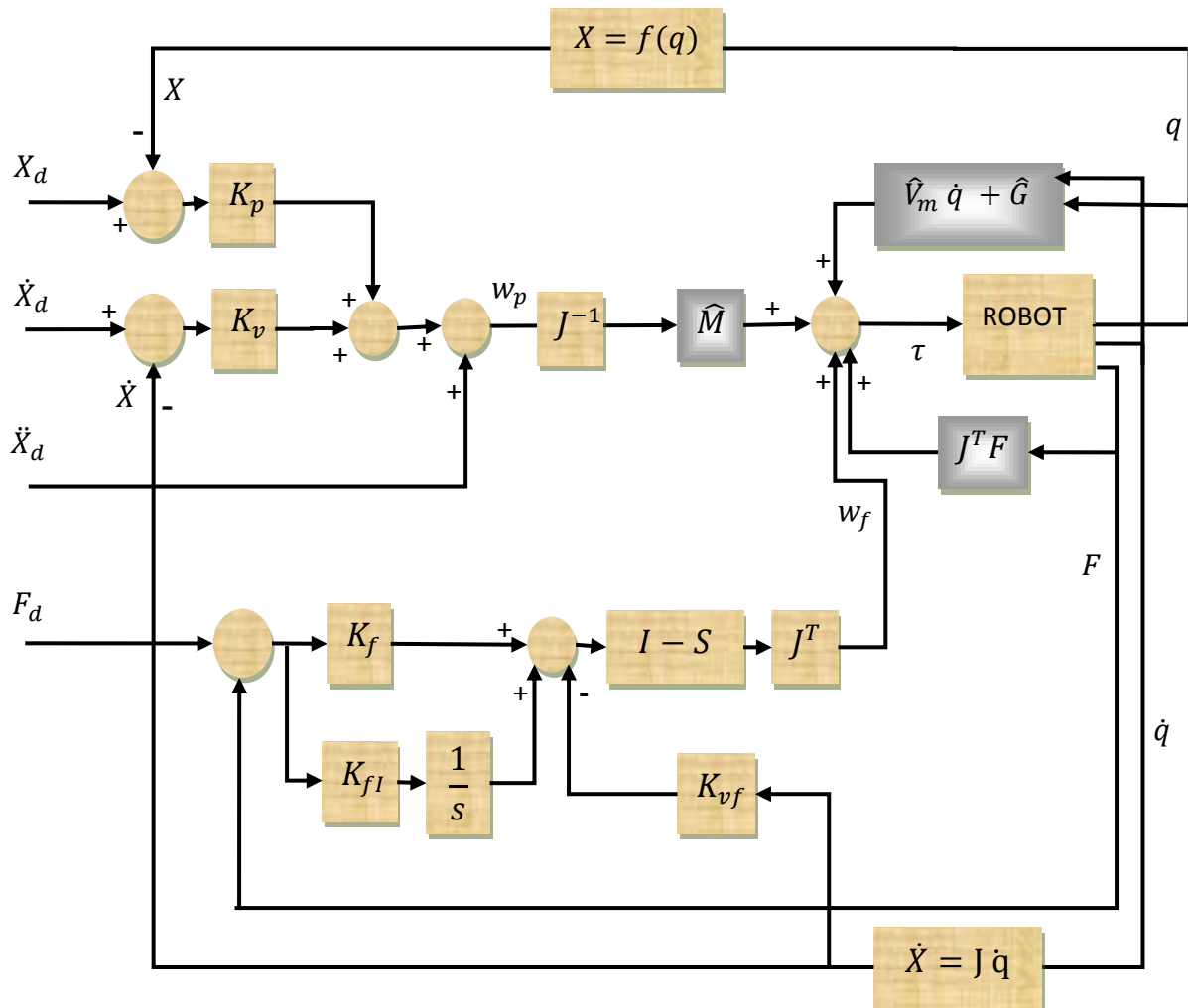
Cette dernière loi de commande n'affecte que les directions commandées en effort.

### **Remarque :**

Dans les applications de compliance (espace contraint), toutes les grandeurs cartésiennes Doivent être exprimées dans le repère de compliance {C} (repère lié au point de contact) ou dans le repère de la base {0}, et dans notre cas le repère de compliance est en parallèle à celui de la base avec seulement d'un décalage de  $x_e$  dans l'axe de l'abscisse  $x$ , (voir fig.4.1) donc

les grandeurs cartésiennes sont maintenues dans le repère de la base {0} ainsi que la force générée lors de l'interaction est décrite aussi dans le repère de la base {0}, alors au lieu d'écrire  ${}^0J, {}^0\ddot{x}_d, {}^0\dot{x}, {}^0\ddot{x}, {}^0k_p, {}^0k_d \dots$  pour les grandeurs exprimées dans le repère de la base, on écrit dans la suite  $J, \ddot{x}_d, \dot{x}, \ddot{x}, k_p, k_d \dots$  pour alléger l'écriture.

Le schéma de cette commande est représenté dans la figure (4.4).



**Figure 4.4.** Commande hybride force/position par découplage non linéaire avec somme des couples articulaires.



**Résultat de simulation :**

Les gains de commande en position sont fixés comme suit :

$$K_p = [200 \ 0; \ 0 \ 8000] \ ; \ K_v = [50 \ 0; \ 0 \ 200]$$

Les gains de la boucle de commande en effort sont fixés tels que:

$$K_{fI} = 500 \ , \ K_f = 150 \ , \ K_{fv} = 200$$

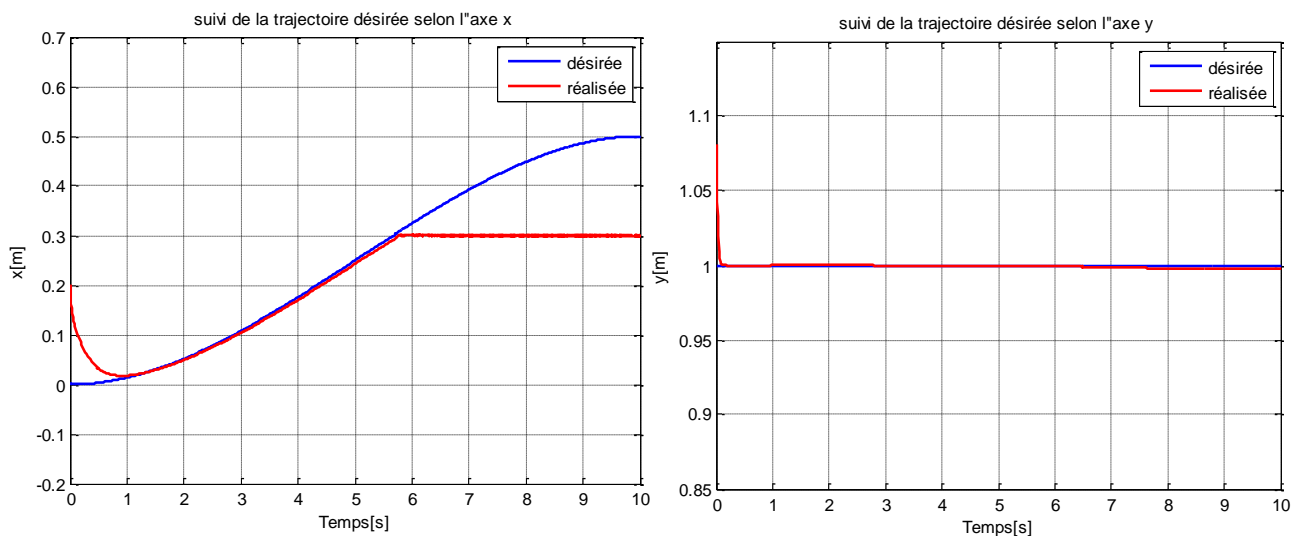
$$\text{La matrice de sélection } \mathbf{s} = [0 \ 0; \ 0 \ 1] \ ; \ \mathbf{I} - \mathbf{s} = [1 \ 0; \ 0 \ 0]$$

La raideur de l'environnement est fixée à :  $6 \times 10^4 \text{ N/m}$

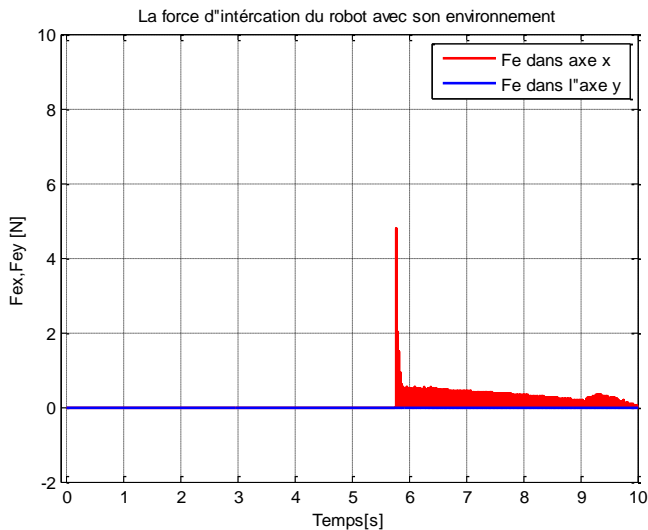
Pour les paramètres du robot, voir l'annexe B

Les résultats obtenus par cette méthode sont illustrés par les figures (4.5, 4.6, 4.7, 4.8).

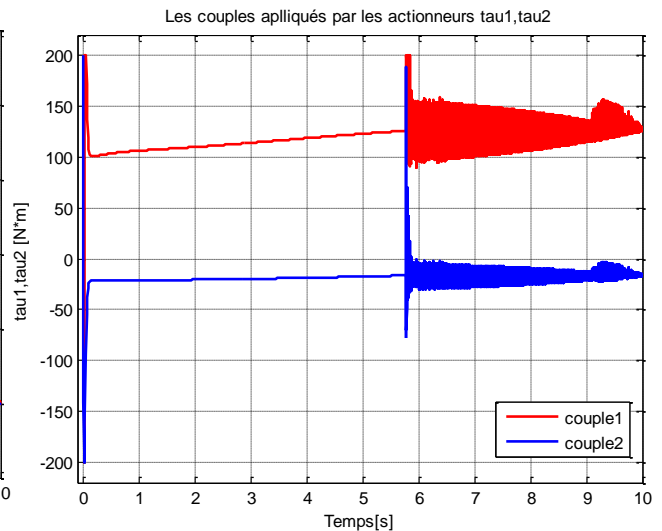
Le modèle du robot est considéré certain donc:  $\hat{M} = M \ , \ \hat{V}_m = V_m \ , \ \hat{G} = G$



**Figure 4.5.** Poursuite de la trajectoire suivant l'axe x. **Figure 4.6.** Poursuite de la trajectoire suivant l'axe y



**Figure 4.7.** Les forces exercées sur l'environnement.



**Figure 4.8.** Couples actionneurs fournis.

### Discussion :

Nous constatons que la trajectoire dans le sens X **Figure 4.5** contient deux parties : celle où l'espace est libre et on remarque une bonne poursuite, et celle où le contact avec l'environnement est imminent, dans ce cas nous observons que le robot maintient sa position et ne dépasse pas l'obstacle, avec la poursuite normale de la trajectoire désirée dans le sens Y **Figure 4.6** là où l'espace dégagé de contrainte.

La force d'interaction du robot avec son environnement converge vers la force désirée  $F_d = \mathbf{0}$ .

Les couples appliqués aux moteurs sont acceptables et ne dépassent pas leurs limites maximales.

Les inconvénients de cette méthode sont :

- L'utilisation de deux lois de commande.
- Besoin d'une connaissance parfaite de l'environnement

## 4.3 La Commande en impédance :

### 4.3.1 La Commande Par raideur active :

Rappelons la loi de commande (2.9) du chapitre II donnée par la formule suivante :

$$\tau = J^T K J (q_d - q) + K_v(\dot{q}_d - \dot{q}) + \hat{G} \quad (4.7)$$

Remarquons de cette dernière expression que cette méthode ne compense que la gravité.

Le schéma de cette commande est identique à celui donné en figure (2.8).

La matrice  $K$  est diagonale par rapport au repère de compliance qui coïncide avec le repère de la base.

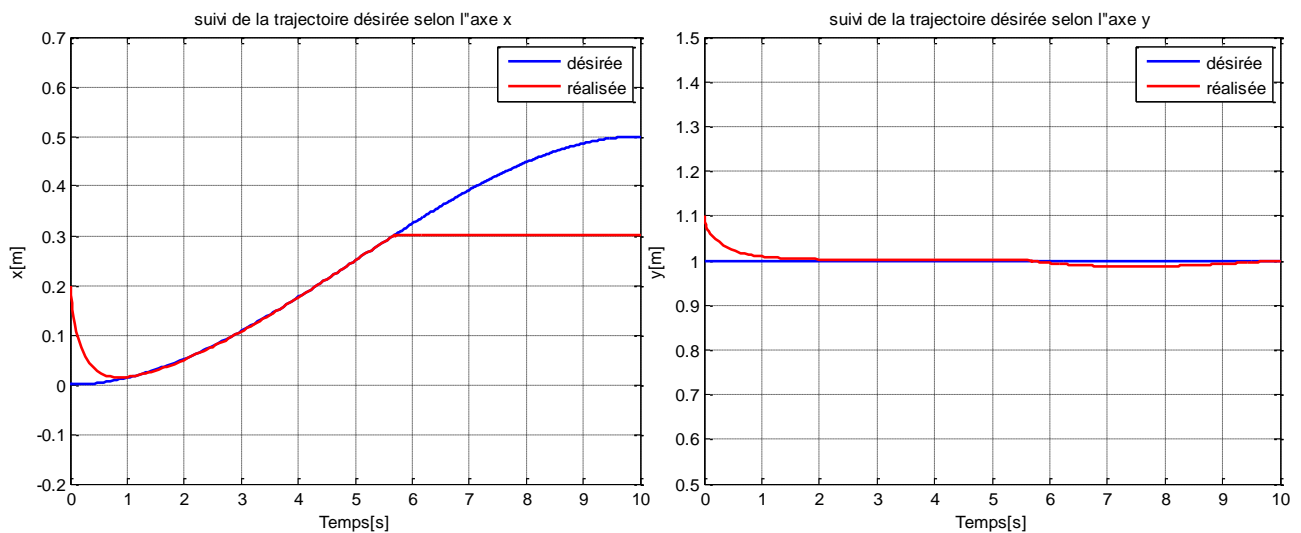
### Résultat de simulation :

Les gains de commande en position sont fixés comme suit :

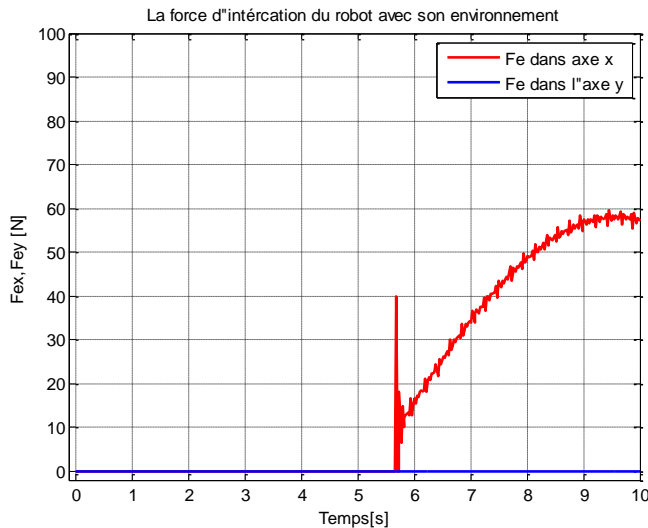
$$K = [300 \ 0; 0 \ 280] \ ; \ K_v = [200 \ 0; 0 \ 150]$$

La raideur de l'environnement est fixée à :  $6 \times 10^4 \text{ N/m}$

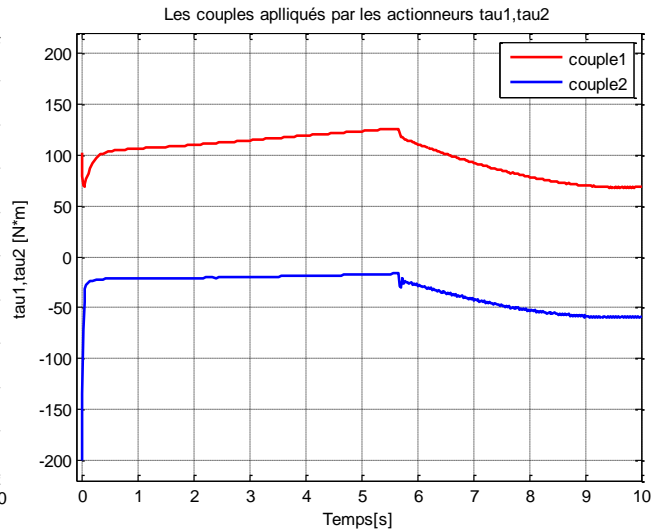
Les résultats obtenus par cette méthode sont illustrés par les figures (4.9, 4.10, 4.11, 4.12).



**Figure 4.9.** Poursuite de la trajectoire suivant l'axe x **Figure 4.10.** Poursuite de la trajectoire suivant l'axe y



**Figure 4.11.** Les forces exercées sur l'environnement.



**Figure 4.12.** Couples actionneurs fournis.

### Discussion :

Cette méthode est équivalente à la commande PD classique, simplement le gain de position dans l'espace articulaire est modifié de telle sorte que l'effecteur semble avoir une certaine raideur le long des degrés de liberté cartésiens.

La **Figure 4.9** représente la trajectoire du robot par rapport l'axe x, ce schéma montre une bonne poursuite de la trajectoire désirée, régime transitoire très court.

Tandis que la trajectoire du robot par rapport l'axe y, présente une petite déviation de son chemin juste après le contact avec l'environnement, cette déviation ne persiste pas et le robot revient à sa trajectoire désirée.

La force d'interaction du robot avec son environnement est environ 50 N, cette valeur par rapport la raideur de l'environnement est très faible, mais tout dépend de l'application du robot.

Les couples appliqués aux moteurs sont très douces et ne dépassent pas leurs limites maximales.

L'avantage de cette méthode est :

- Relativement simple à mettre en œuvre.
- Pas besoin d'une connaissance parfaite de l'environnement.

### 4.3.2 La Commande Par matrice de raideur :

Cette commande nécessite dans sa mise en œuvre le modèle dynamique du robot décrit par l'équation (4.3), la correction est effectuée dans l'espace cartésien précisément dans le repère de la base, la loi de commande est donnée par la formule suivante :

$$\tau = \hat{M} J^{-1}(w - \dot{J} \dot{q}) + \hat{V}_m \dot{q} + \hat{G} + J^T F_e \quad (4.8)$$

Où  $w$  est une entrée auxiliaire.

Si le modèle du robot est considéré certain alors :

$$\hat{M} = M, \quad \hat{V}_m = V_m, \quad \hat{G} = G \quad (4.9)$$

Donc en boucle fermée on aura :

$$\ddot{X} = w \quad (4.10)$$

$w$  est choisie telle que :

$$w = \ddot{X}_d + K_p (X_d - X - dX) + K_v (\dot{X}_d - \dot{X}) \quad (4.11)$$

Où  $K_p, K_v$  sont les matrices de gains proportionnels et dérivés.

D'après l'équation (2.6) du chapitre II On a :

$$dX = K_f F_e \quad (4.12)$$

Le schéma de cette commande est identique à celui donné en figure (2.9).

#### Résultat de simulation :

Les gains de commande proportionnels et dérivés sont fixés comme suit :

$$K_p = [5 \ 0; 0 \ 150] ; K_v = [10 \ 0; 0 \ 8]$$

L'inverse de la raideur est choisi telle que :  $K_f = [20 \ 0; 0 \ 15]$

La raideur de l'environnement est fixée à :  $6 \times 10^4 \text{ N/m}$

Les résultats obtenus par cette méthode sont illustrés par les figures (4.13, 4.14, 4.15, 4.16).

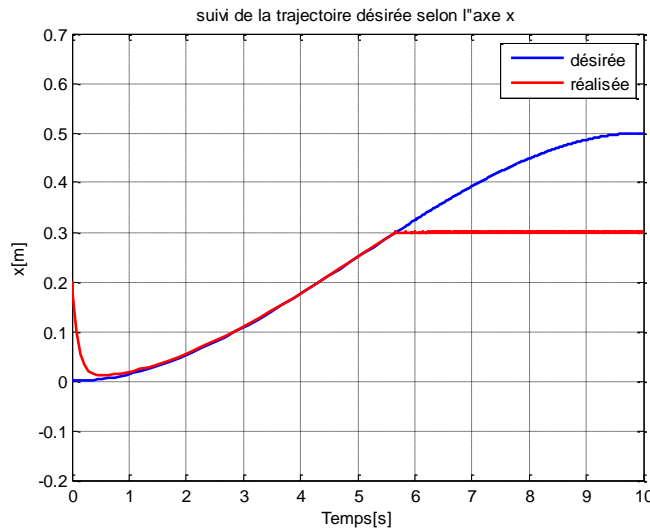


Figure 4.13. Poursuite de la trajectoire suivant l'axe x

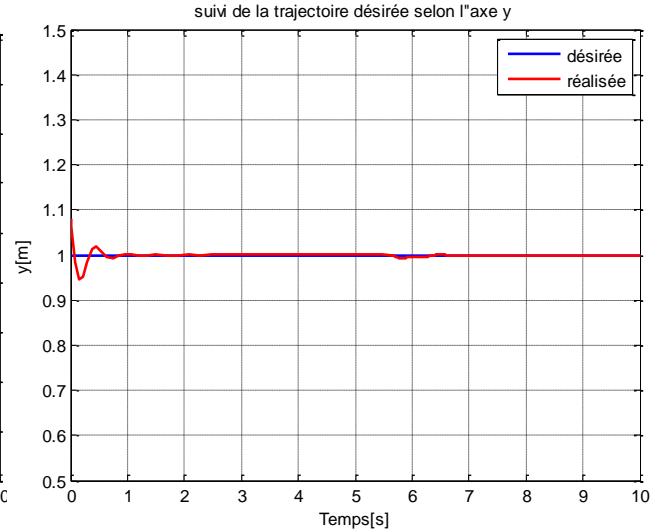


Figure 4.14. Poursuite de la trajectoire suivant l'axe y

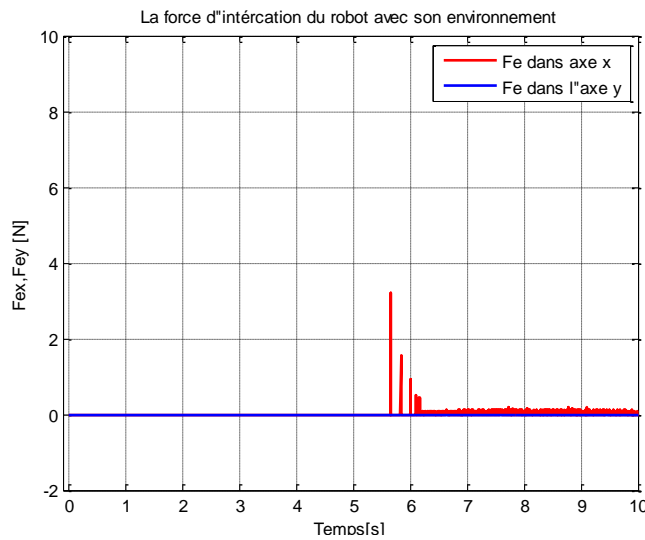


Figure 4.15. Les forces exercées sur l'environnement.

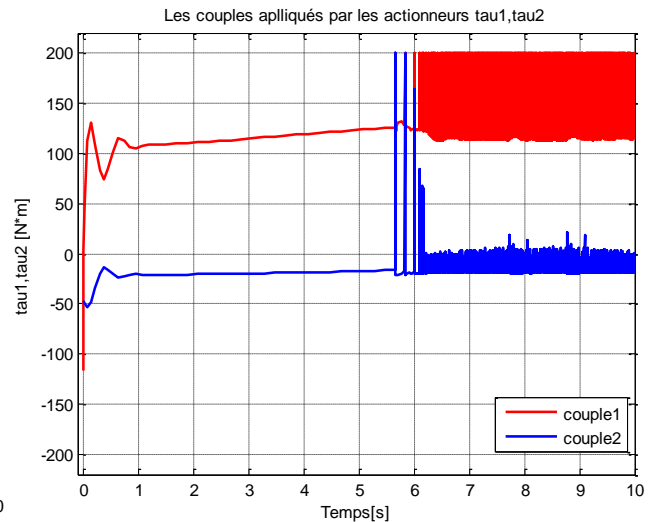


Figure 4.16. Couples actionneurs fournis.

### Discussion :

Cette méthode est équivalente à la précédente sauf pour la force d'interaction qui est dans ce cas là, très acceptable avec des pics ne dépassants pas le 4N.

Les couples appliqués aux moteurs ne sont pas douces néanmoins, ces pics limités dans l'amplitude sont aussi limités dans la fréquence.

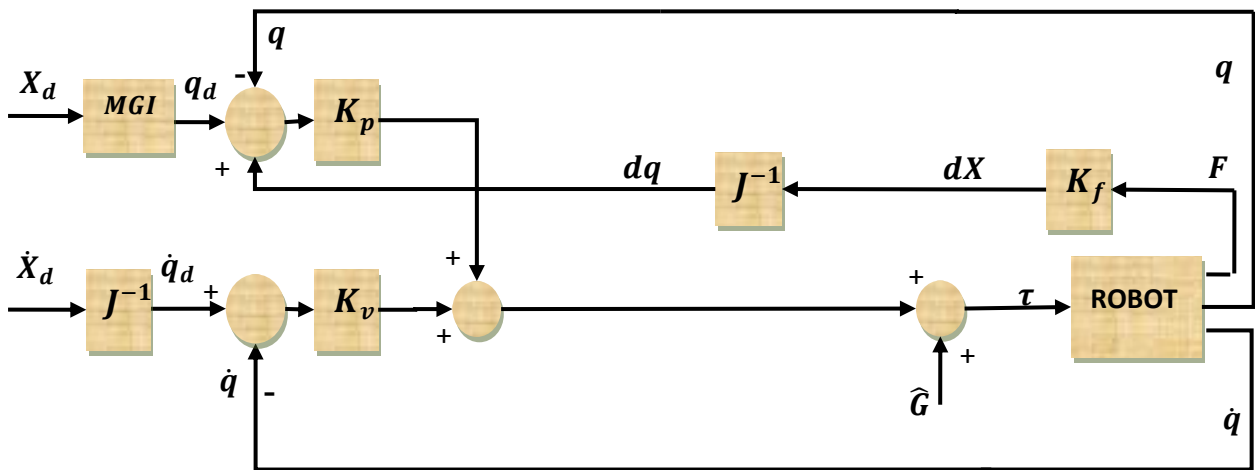
➤ On peut agir autrement avec la transformation du déplacement - due au contact- de l'espace cartésien vers l'espace articulaire et faire la correction dans l'espace articulaire, Nous avons :

$$dq = J^{-1} dX \quad (4.13)$$

La loi de commande est formulée comme suite :

$$\tau = K_p (q_d - q - dq) + K_v (\dot{q}_d - \dot{q}) + \hat{G} \quad (4.14)$$

Le schéma de principe de cette commande est présenté à la figure suivante (4.17)



**Figure 4.17.** Principe de la commande par matrice de raideur en espace articulaire.

**Résultat de simulation :**

Les gains de commande proportionnels et dérivés sont fixés comme suit :

$$K_p = [200 \ 0; 0 \ 900] \ ; \ K_v = [60 \ 0; 0 \ 180]$$

L'inverse de la raideur est choisi telle que :  $K_f = 15$

La raideur de l'environnement est fixée à :  $6 \times 10^4$  N/m

Les résultats obtenus par cette méthode sont illustrés par les figures (4.18, 4.19, 4.20, 4.21).

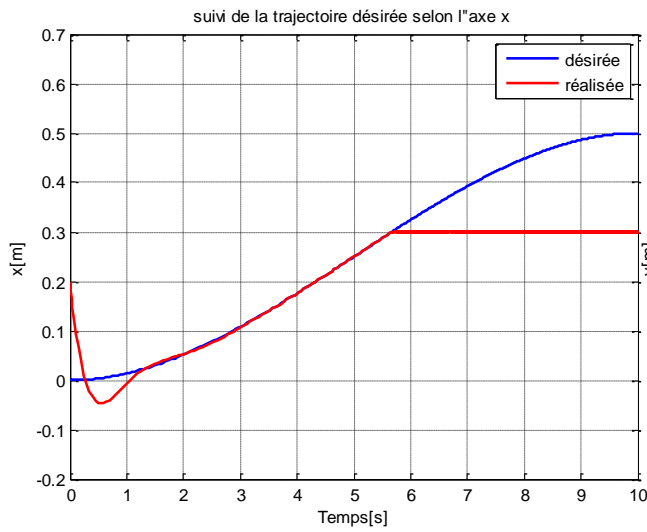


Figure 4.18. Poursuite de la trajectoire suivant l'axe

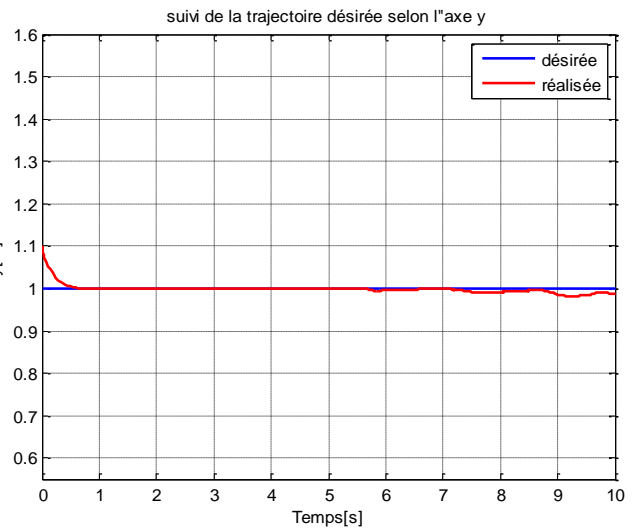


Figure 4.19. Poursuite de la trajectoire suivant l'axe y

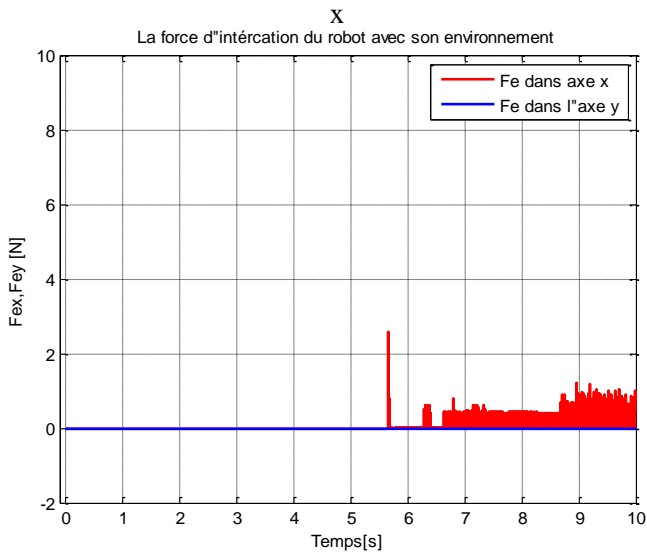


Figure 4.20. Les forces exercées sur l'environnement.

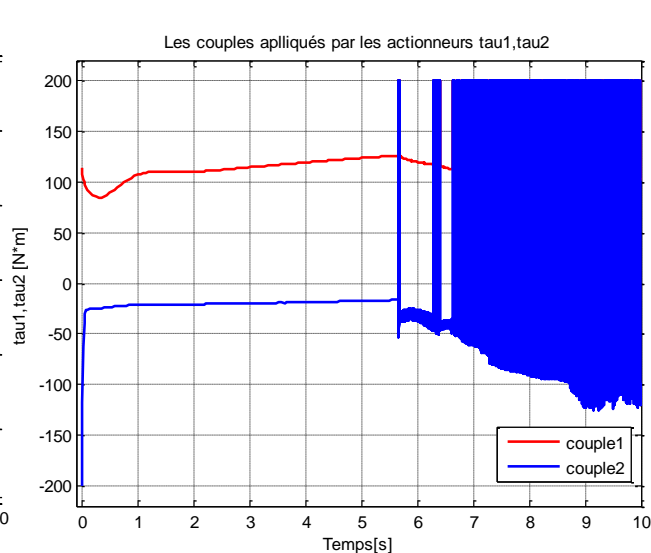


Figure 4.21. Couples actionneurs fournis.

### Discussion :

Cette méthode est équivalente à la précédente avec un petit dépassement au niveau de la trajectoire  $x(t)$ .

Dans ce cas Les couples appliqués aux moteurs ne sont pas douces, mais ils sont limités dans l'amplitude.



### 4.3.3 La Commande en impédance par découplage non linéaire :

Dans cette loi de commande, on utilise la dynamique inverse du robot pour compenser et découpler la dynamique du système, afin d'arriver à un système linéaire découplé, appliquant ainsi la commande PD simple ;

Dans cette section on aborde les deux approches de la commande en impédance, celle basée sur la position et celle qui n'affecte pas la position.

➤ **La commande en impédance basée sur la position :**

Ce type de commande est basé sur l'obtention d'une trajectoire référence à partir des trajectoires désirées qui sont affectées par la force d'interaction avec l'environnement via l'impédance désirée.

Le modèle dynamique du robot en contact avec son environnement, et avec la description de l'accélération dans l'espace cartésien est donné par l'équation (4.3).

On reformule l'impédance désirée donnée par l'équation (2.21) du chapitre II, comme suite :

$$\ddot{X}_r = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}_r) + K(X_d - X_r) - F_e ] \quad (4.15)$$

$X_r$  : C'est une déviation de la trajectoire désirée  $X_d$  , en présence d'une force d'interaction  $F_e$

Si  $F_e = 0$       Donc       $X_r = X_d$

Si  $F_e \neq 0$       Donc       $X_r \neq X_d$

Donc la loi commande est donnée par:

$$\tau = M J^{-1}(\mathbf{w} - \dot{J} \dot{q}) + V_m \dot{q} + G + J^T F_e \quad (4.16)$$

Avec :

$$\mathbf{w} = \ddot{x}_r + k_d(\dot{x}_r - \dot{x}) + k_p(x_r - x) \quad (4.17)$$

En boucle fermée on arrive à :

$$\mathbf{w} = \ddot{x} \quad (4.18)$$

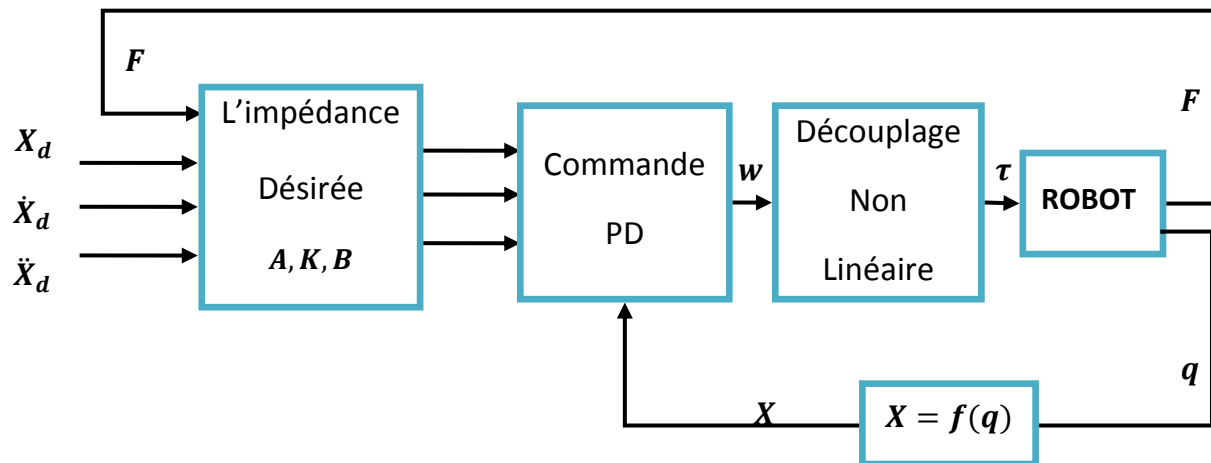
Alors la dynamique des erreurs s'écrit comme suite:

$$\ddot{e}_r + k_d \dot{e}_r + k_p e_r = 0 \quad (4.19)$$

Avec :

$$e_r = x_r - x \quad (4.20)$$

Le schéma de principe de cette commande est représenté dans la figure suivante (4.22)



**Figure 4.22.** Principe de la commande en impédance par découplage basée sur la position.

### Résultat de simulation :

Les éléments des matrices utilisées pour la commande PD sont fixés comme suit :

$$k_p = \text{diag}(200, 200) ; k_d = \text{diag}(20, 20)$$

Les éléments des matrices d'inertie, d'amortissement et de raideur désirées sont choisies comme suit :

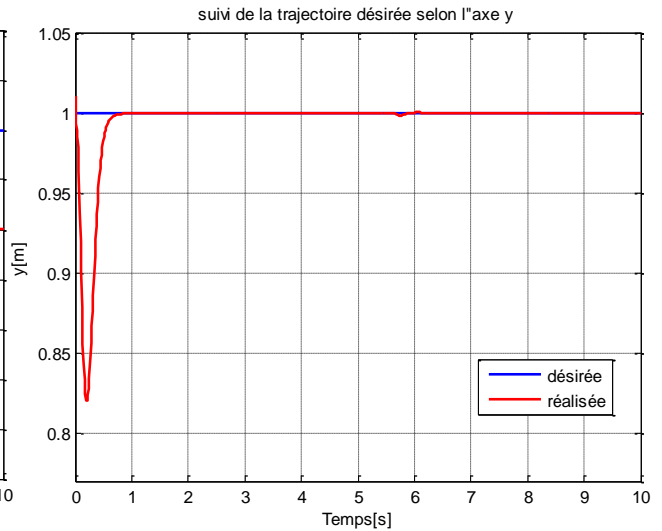
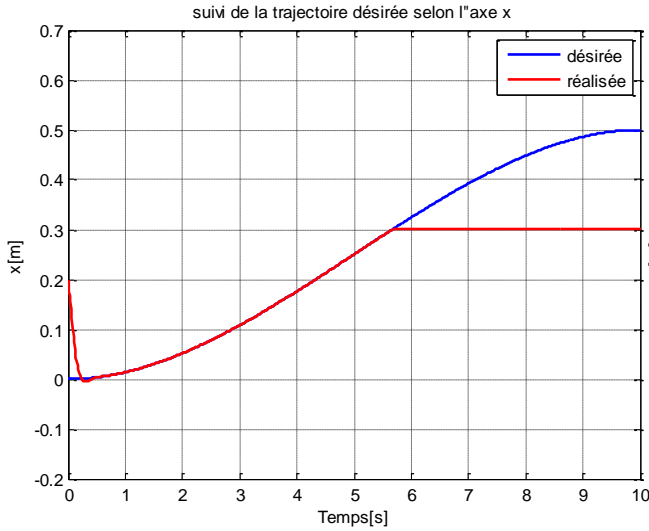
Pour la matrice d'inertie sont attribuées des valeurs élevées aux directions contraintes pour diminuer la dynamique du robot, tandis qu'aux directions libres sont imposées des valeurs faibles.

Dans le cas de la matrice d'amortissement sont imposées des valeurs élevées aux directions contraintes pour amortir le contact, et des valeurs faibles aux directions libres.

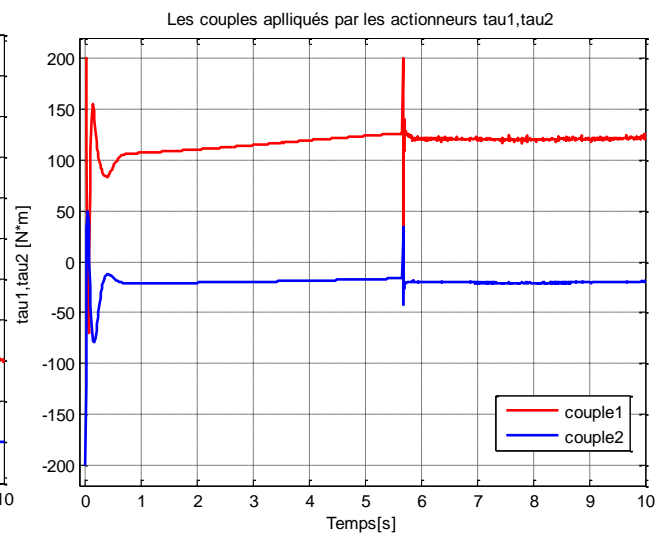
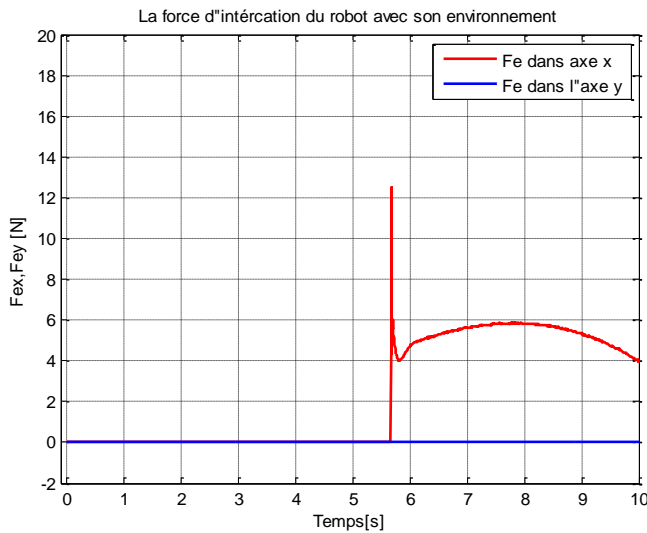
Enfin, Pour la matrice de raideur sont fixées des valeurs élevées relatives aux directions non contraintes pour obtenir une bonne précision, et des valeurs faibles relatives aux directions contraintes pour réduire l'effort de contact.

$$A^{-1} = \text{diag}(4, 4) ; K = \text{diag}(20, 300) ; B = \text{diag}(60, 40)$$

La raideur de l'environnement est fixée à :  $6 \times 10^4$  N/m



**Figure 4.23.** Poursuite de la trajectoire selon l'axe x. **Figure 4.24.** Poursuite de la trajectoire selon l'axe y.



**Figure 4.25.** Les forces exercées sur l'environnement.

**Figure 4.26.** Couples actionneurs fournis.

### Discussion :

Les résultats de simulation **Figure 4.23 -4.24** montrent une bonne poursuite des trajectoires, avec un temps de réponse inférieur à 1s.

La force d'interaction est très limitée, avec une valeur maximale environ  $F_{e\_max}=12N$ , sa convergence vers zéro est claire de la **Figure 4.25**.

Les couples appliqués aux moteurs sont acceptables et ne dépassent pas leurs limites maximales, ce qui rend cette commande réalisable.

#### ➤ La commande en impédance basée sur l'effort :

Dans cette type de commande, La position désirée n'est pas directement affectée, dans ce cas l'impédance désirée est introduite complètement dans la commande, sans passer par l'extraction d'une valeur de référence.

Le modèle dynamique du robot en contact avec l'environnement est donné par l'équation(4.3)

On reformule L'impédance désirée donnée par l'équation (2.21) du chapitre II comme suite :

$$\ddot{X} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F_e ] \quad (4.21)$$

Donc la loi commande est donnée par:

$$\tau = M J^{-1}(\mathbf{w} - \dot{J} \dot{q}) + V_m \dot{q} + G + J^T F_e \quad (4.22)$$

Avec :

$$\mathbf{w} = \ddot{X}_d + A^{-1} [ B(\dot{X}_d - \dot{X}) + K(X_d - X) - F_e ] \quad (4.23)$$

En boucle fermée on arrive à l'impédance désirée (4.21).

### Résultat de simulation :

Les éléments des matrices d'inertie, d'amortissement et de raideur, désirées sont choisies comme suit :

$$A = [0.5 \ 0; 0 \ 1] ; K = [15 \ 0; 0 \ 80] ; B = [20 \ 0; 0 \ 10]$$

La raideur de l'environnement est fixée à :  $6 \times 10^4 \text{ N/m}$

Les résultats obtenus de la commande en impédance basée sur l'effort et par l'application de la commande de linéarisation, sont présentés dans les figures suivantes (4.27, 4.28, 4.29, 4.30)

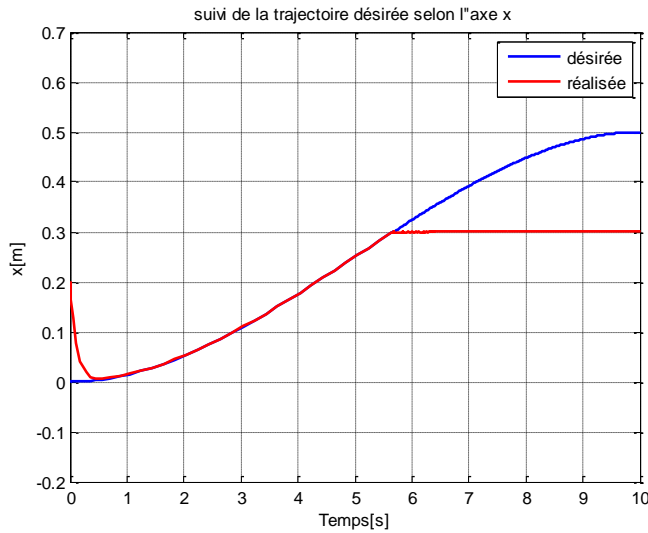


Figure 4.27. Poursuite de la trajectoire selon l'axe x.

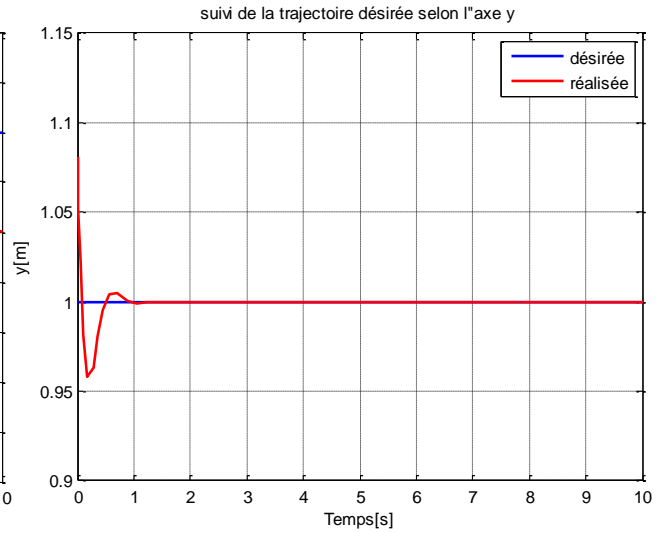


Figure 4.28. Poursuite de la trajectoire selon l'axe y.

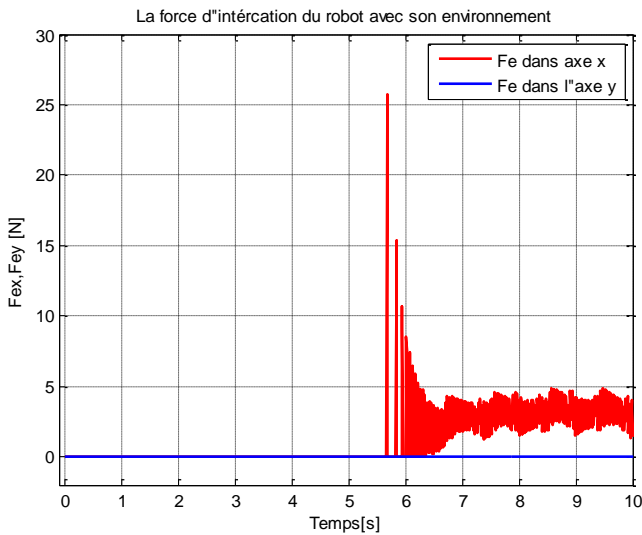


Figure 4.30. Les forces exercées sur l'environnement.

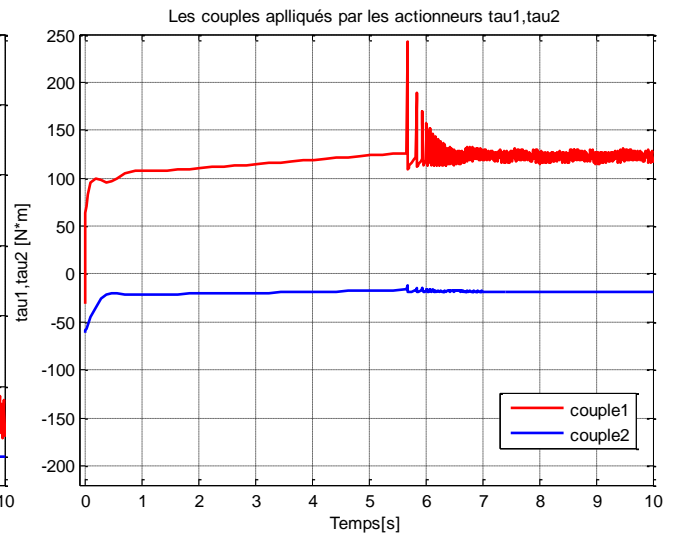


Figure 4.29. Couples actionneurs fournis.

### Discussion :

Dans ce type de commande nous n'avons pas utilisé un régulateur PD, puisque il n'y a pas de correction de position, l'objectif ici est d'asservir le robot avec une accélération issue de l'impédance désirée (4.21) cette dernière est injectée directement dans la dynamique du robot (double intégrateur après découplage non linéaire).

On remarque un bon suivi de la trajectoire Figure 4.27 - 4.28, la force d'interaction est limitée de l'ordre de 25N, Les couples appliqués aux moteurs sont acceptables et ne dépassent pas leurs limites maximales.

### 4.3.4 La commande en impédance à mode glissant :

Dans cette section l'impédance désirée est réalisée, par une commande mode glissant.

Le modèle dynamique du robot en contact avec l'environnement est donné par l'équation(4.3)

Les trajectoires références  $\ddot{X}_r, \dot{X}_r, X_r$  sont générées à partir de l'équation (4.15)

La surface de glissement  $s$  est choisit d'une manière à minimiser l'erreur entre la trajectoire référence et actuelle du robot.

$$s = \dot{e}_r + \lambda e_r \quad (4.24)$$

Avec :  $e_r = x_r - x$

La fonction de Lyapunov est choisit comme suite :

$$V = \frac{1}{2} s^T M s \quad (4.25)$$

Sa dérivée est donnée par :

$$\dot{V} = s^T M \dot{s} + \frac{1}{2} s^T \dot{M} s$$

$$\dot{V} = s^T (M \dot{s} + V_m s)$$

Avec :  $X (\dot{M} - 2C) = 0$

$$\dot{V} = s^T [ M (\ddot{e}_r + \lambda \dot{e}_r) + C s ]$$

$$\dot{V} = s^T [ -M \ddot{x} + M(\ddot{x}_r + \lambda \dot{e}_r) + C s ] \quad (4.26)$$

On pose :

$$\dot{x}_s = M(\ddot{x}_r + \lambda \dot{e}_r) \quad (4.27)$$

$$\dot{V} = s^T [-M \ddot{x} + M \dot{x}_s + V_m s] \quad (4.28)$$

On substitue  $\ddot{x}$  par sa valeur de l'équation (4.3) nous obtenons la relation suivante :

$$\begin{aligned} \dot{V} &= s^T [-M (J M^{-1}(\tau - V_m \dot{q} - G - J^T F_e) + j \dot{q}) + M \dot{x}_s + V_m s] \\ \dot{V} &= s^T [-M J M^{-1}(\tau - V_m \dot{q} - G - J^T F_e) - M j \dot{q} + M \dot{x}_s + V_m s] \end{aligned} \quad (4.29)$$

Pour assurer la négativité de  $\dot{V}$  la commande doit être de la forme suivante :

$$\tau = M J^{-1} [\dot{x}_s - j \dot{q} + M^{-1} V_m s + \rho \text{sign}(s)] + G + V_m \dot{q} + J^T F_e \quad (4.30)$$

Avec :  $\rho > 0$

Par la substitution de la loi de commande (4.30) dans l'équation de  $\dot{V}$  (4.29) on trouve:

$$\dot{V} = -\rho s^T \text{sign}(s) \quad (4.31)$$

Qui représente la loi d'attraction.

### Résultat de simulation :

Les éléments des matrices d'inertie, d'amortissement et de raideur, désirées sont choisies comme suit :

$$A^{-1} = [3 \ 0; 0 \ 6] ; K = [10 \ 0; 0 \ 250] ; B = [100 \ 0; 0 \ 25]$$

Le gain de mode glissant :  $\rho = [10 \ 0; 0 \ 10]$ , coefficient de la surface :  $\lambda = [30 \ 0; 0 \ 10]$

La raideur de l'environnement est fixée à :  $6 \times 10^4$  N/m. Les résultats obtenus de la commande en impédance basée sur l'effort et par l'application de la commande de linéarisation, sont présentés dans les figures suivantes (4.31 - 4.32 - 4.33 - 4.34)

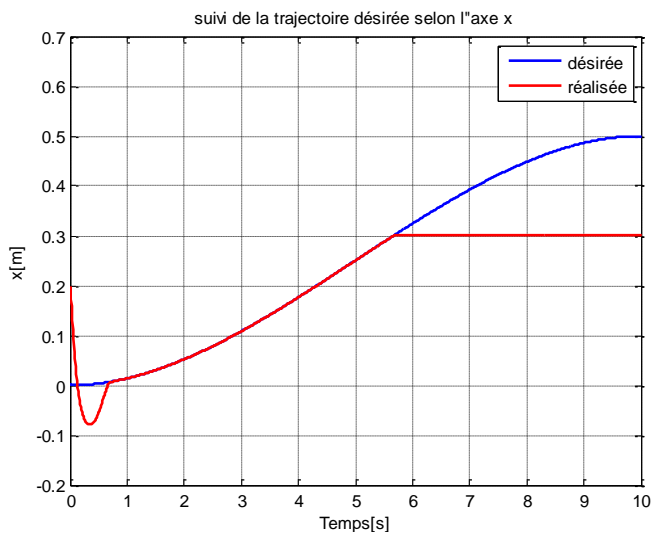


Figure 4.31. Poursuite de la trajectoire suivant l'axe x.

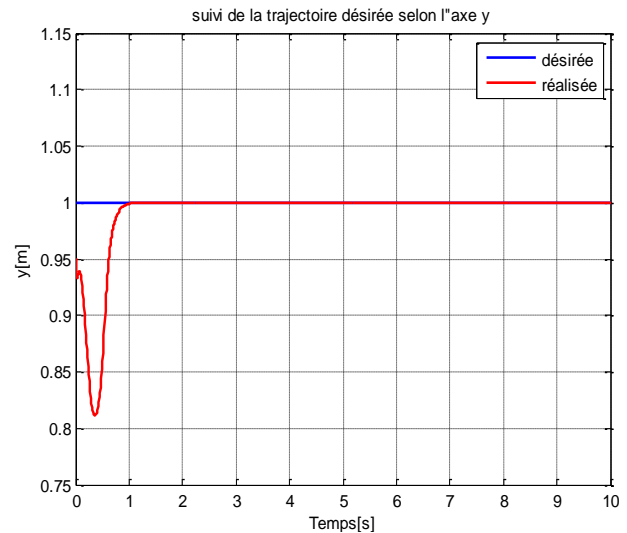


Figure 4.32. Poursuite de la trajectoire suivant l'axe y

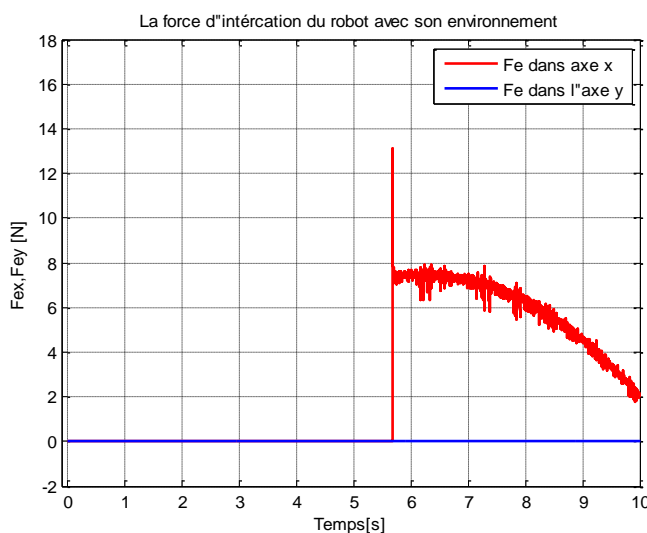


Figure 4.33. Les forces exercées sur l'environnement.

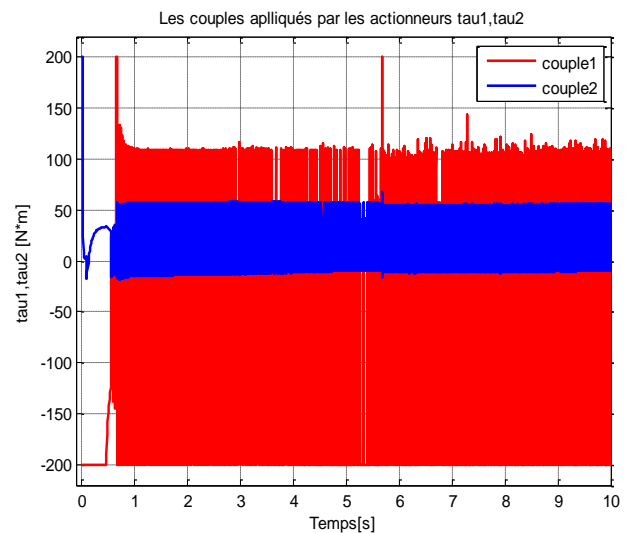


Figure 4.34. Couples actionneurs fournis.

### Discussion :

Le rôle de La commande par mode glissant -dans cette stratégie de commande d'impédance- est d'agir directement sur l'erreur entre la position actuelle du robot est la référence fournie par l'impédance désirée, afin de l'annuler en temps fini.

Nous constatons un bon suivi de la trajectoire **Figure 4.31 - 4.32**, la force d'interaction est contrôlée implicitement elle tend vers zéro, Les couples appliqués aux moteurs sont acceptables et ne dépassent pas leurs limites maximales, le phénomène de chattering est clair sur la figure 4.34 à cause du terme de commutation dans la commande mode glissant.



### ➤ Test de robustesse :

Nous allons tester la robustesse de la commande mode glissant par la modélisation d'une charge de masse  $m_{per}$  attachée à l'outil terminal du manipulateur, et après certain temps ( $t_{per} = 3s$ ) la charge chute provoquant ainsi une perturbation, donc la commande par mode glissant doit surmonter cette perturbation sans connaître tous ces détails (la masse de la charge ainsi l'instant de sa chute).

Le model du robot en contact avec son environnement est donnée par l'expression suivante :

$$M J^{-1}(\ddot{X} - \dot{J} \dot{q}) + V_m \dot{q} + G + J^T F_e + \tau_{per} = \tau \quad (4.32)$$

Où :  $\tau_{per}$  est le couple dû à la masse attachée à l'outil terminal du manipulateur  $m_{per}$ .

Le couple dû à la charge est donné par : [4]

$$\tau_{per} = m_{per} J (J \ddot{q} + \dot{J} \dot{q} + G) \quad (4.33)$$

### Résultat de simulation :

Les éléments des matrices d'inertie, d'amortissement et de raideur, désirées sont choisies comme suit :

$$A^{-1} = [0.6 \ 0; 0 \ 1.8] ; K = [10 \ 0; 0 \ 200] ; B = [80 \ 0; 0 \ 20]$$

Le gain de mode glissant :  $\rho = [30 \ 0; 0 \ 60]$

La raideur de l'environnement est fixée à :  $6 \times 10^4$  N/m. Les résultats obtenus de la commande en impédance basée sur l'effort et par l'application de la commande de linéarisation, sont présentés dans les figures suivantes (4.35 - 4.36 - 4.37 - 4.38)

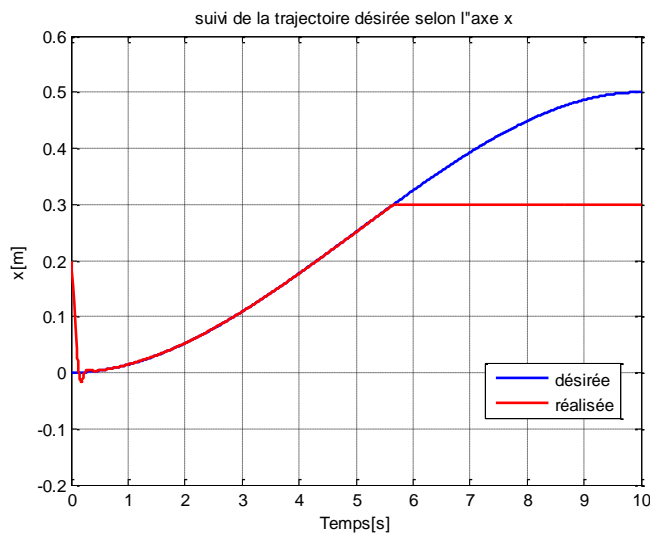


Figure 4.35. Poursuite de la trajectoire suivant l'axe x

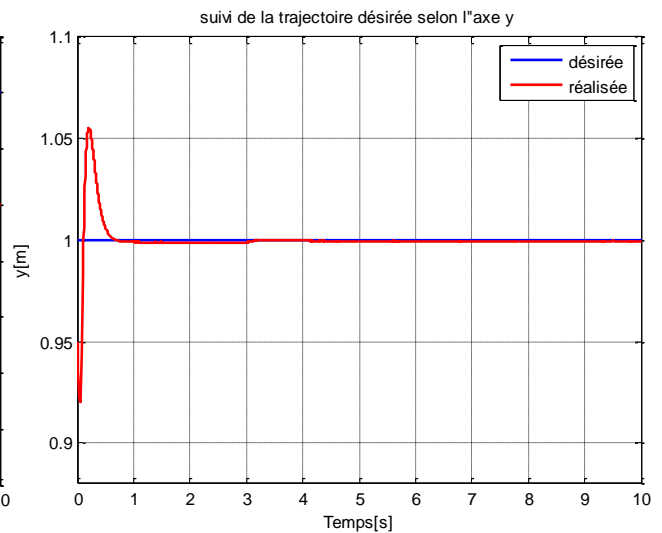


Figure 4.36. Poursuite de la trajectoire suivant l'axe y

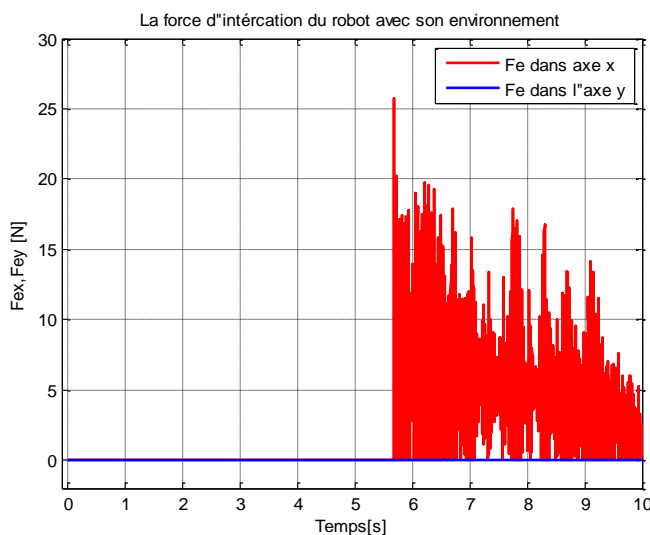


Figure 4.37. Les forces exercées sur l'environnement.

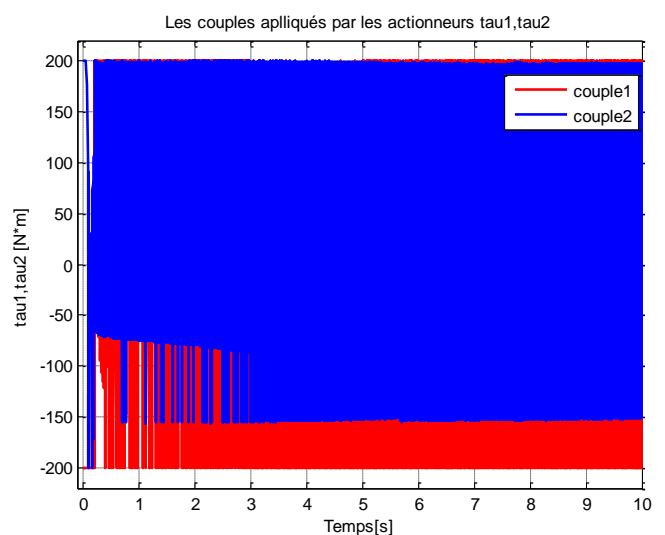


Figure 4.38. Couples actionneurs fournis.

### Discussion :

On a pris pour la charge  $m_{per} = 1kg$  , nous constatons d'après ces résultats que cette méthode de commande est robuste vis-à-vis les perturbations externes. Le bon suivi des trajectoires **Figure 4.35 - 4.36**, montre la robustesse de cette commande.

Quant aux couples appliqués au robot, ils ne sont pas lisses, notamment à ( $t_{per} = 3s$ ) l'instant de la chute de charge, on remarque une fréquence importante des couples à partir de cet instant, où le robot essaye de rattraper la chute de la charge.

### 4.3.5 La commande en impédance par Backstepping:

Nous avons vu dans la section §3.3.2 la commande backstepping appliquée au manipulateur en espace libre, dans cette section nous allons appliquer la même commande mais en espace contraint, alors on doit compenser la force d'interaction et utiliser l'accélération cartésienne.

Le modèle dynamique du robot en contact avec l'environnement est donné par l'équation(4.3)

Les trajectoires références  $\ddot{X}_r, \dot{X}_r, X_r$  sont générées à partir de l'équation (4.15)

On utilise pour le développement de la commande par backstepping en espace contraint, la même méthode utilisée dans la section §3.3.2 du chapitre III.

Le modèle dynamique du robot après reformulation est donnée par :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \ddot{X} = JM^{-1}(\tau - V\dot{q} - G - J^T F_e) + j\dot{q} \end{cases} \quad (4.34)$$

Après la constitution des fonctions de Lyapunov en deux étapes on arrive à l'espace  $z_1 z_2$  suivant :

$$\begin{cases} \dot{z}_1 = z_2 - c_1 z_1 \\ \dot{z}_2 = JM^{-1}(\tau - V\dot{q} - G) + j\dot{q} - \ddot{X}_d + c_1 (z_2 - c_1 z_1) \end{cases} \quad (4.35)$$

Avec :

$$\begin{cases} z_1 = x_1 - a_0 \\ z_2 = x_2 - x_2^r \end{cases} \quad (4.36)$$

Et :

$$\begin{cases} a_0 = x_{1d} \\ a_1 = x_2^r \end{cases} \quad (4.37)$$

On forme la fonction de Lyapunov sur la base de cet espace, comme suite :

$$\begin{aligned} \dot{V} &= z_1^T \dot{z}_1 + z_2^T \dot{z}_2 \\ &= -c_1 z_1^T z_1 + z_2^T \left[ JM^{-1}(\tau - V\dot{q} - G - J^T F_e) + j\dot{q} - \ddot{X}_d + z_1 \right. \\ &\quad \left. + c_1 (z_2 - c_1 z_1) \right] \end{aligned} \quad (4.38)$$

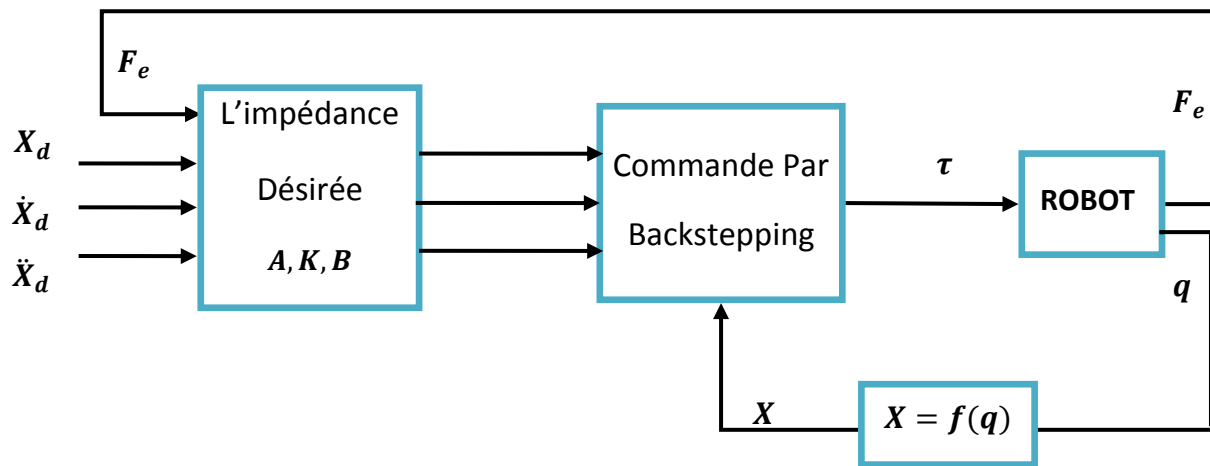
Sur la base de cette fonction qu'on développe la loi de commande qui assure la convergence de  $z_1$   $z_2$  vers 0, comme suite :

$$\tau = MJ^{-1} \left[ \ddot{X}_d - j \dot{q} - z_1 - c_1 (z_2 - c_1 z_1) - c_2 z_2 \right] + V\dot{q} + G + J^T F_e \quad (4.39)$$

Ou sous la forme suivante :

$$\tau = MJ^{-1} \left[ \ddot{X}_d - j \dot{q} + (c_1 c_1 - 1) z_1 - (c_1 + c_2) z_2 (z_2 - c_1 z_1) - c_2 z_2 \right] + V\dot{q} + G + J^T F_e \quad (4.40)$$

Le schéma de principe de cette commande est représenté dans la figure 4.39 suivante :



**Figure 4. 39.** Principe de la commande en impédance par Backstepping.

### Résultat de simulation :

Les éléments des matrices d'inertie, d'amortissement et de raideur, désirées sont choisies comme suit :

$$A^{-1} = [1 \ 0; 0 \ 1] ; K = [20 \ 0; 0 \ 100] ; B = [250 \ 0; 0 \ 20]$$

Les gains de la commande backstepping :  $C_1 = 10, C_2 = 10$

La raideur de l'environnement est fixée à :  $6 \times 10^4$  N/m. Les résultats obtenus de la commande en impédance basée sur l'effort et par l'application de la commande de linéarisation, sont présentés dans les figures suivantes (4.35 - 4.36 - 4.37 - 4.38)

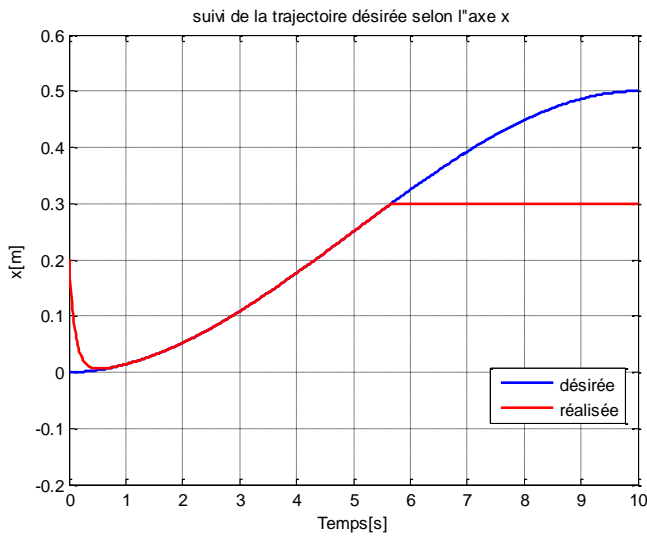


Figure 4.40. Poursuite de la trajectoire suivant l'axe x

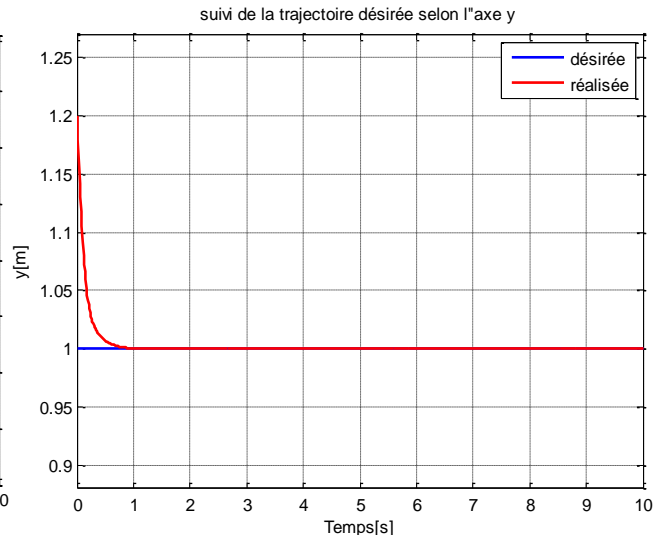


Figure 4.41. Poursuite de la trajectoire suivant l'axe y

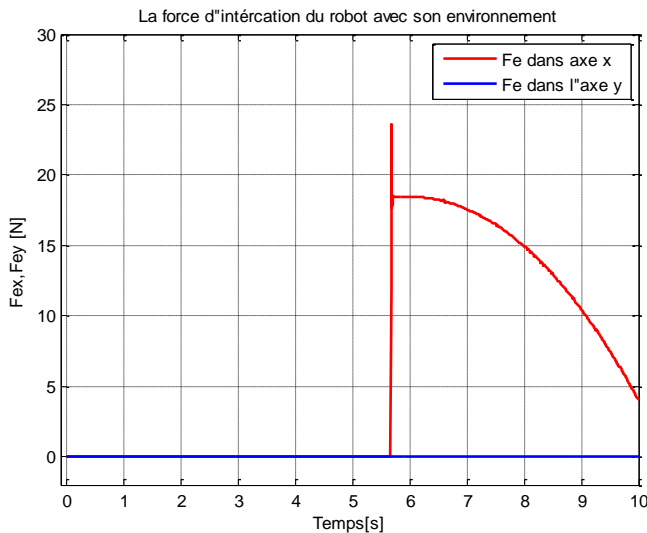


Figure 4.42. Les forces exercées sur l'environnement.

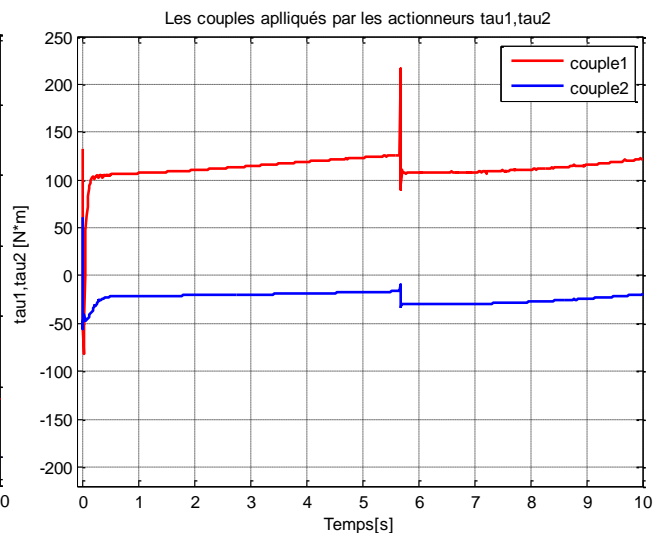


Figure 4.43. Couples actionneurs fournis.

### Discussion :

Nous constatons une très bonne poursuite des trajectoires **Figure 4.40 - 4.41**, la force d'interaction est contrôlée implicitement elle tend vers zéro rapidement, Les couples appliqués aux moteurs sont très acceptables et ne dépassent pas leurs limites maximales, donc ce type de commande donne un bon résultat avec des couples moteur lisses.

#### **4.4 Conclusion :**

Nous avons appliqué les différentes stratégies de commande en effort avec l'adoption des commandes non linéaires et parvenu aux objectifs prédéfinis, la commande hybride force/position a donné un résultat acceptable malgré que cette méthode nécessite l'élaboration de deux lois de commande celle en position et son complément en force dans les directions contraintes.

Les types de commande en impédance sont multiples, on a appliqué la commande par raideur active qui a montré un résultat passable avec une simplicité de réalisation, la commande par matrice de raideur qui a permis de rattraper l'insuffisance de la commande précédente en matière de la stabilisation de la force d'interaction.

La commande en impédance qui est un cas général des commandes précédentes [15] (par raideur active, par la matrice de raideur) montre un très bon résultat, elle permet de donner au robot un comportement dynamique désirée vis-à-vis de l'environnement durant l'interaction, elle permet aussi de commander le robot avant l'interaction en espace libre, la commande en impédance a l'avantage de gérer les deux situations du robot dans un espace contraint ou libre.

Ensuite nous avons appliqué la commande en impédance avec les trois types de commande non linéaire, premièrement le découplage non linéaire cette méthode représente une bonne solution de la commande en impédance que ce soit basée sur la position ou l'effort, néanmoins ce type de commande non linéaire a le recours d'un modèle exacte du robot si non elle présente des instabilités indésirables ;

La deuxième commande c'est celle de mode glissant, il a été appliqué avec succès ces résultats montrent sa robustesse vis-à-vis les perturbations externes,

le Backstepping donne un très bon résultat de poursuite de trajectoires, avec des couples lisses et convergence rapide de la force d'interaction vers le zéro.

## **Conclusion générale :**

L'objectif de ce mémoire est d'appliquer les commandes non linéaires au robot manipulateur dans un espace contraint, nous avons vu que les commandes en position qui permettent de l'asservir à suivre une trajectoire désirée en espace libre (pas d'obstacle dans son espace de travail), sont insuffisantes en présence des contraintes devant le robot.

A partir de ce constat nous avons abordé le problème de la commande en effort, qui dispose actuellement de plusieurs applications industrielles intéressantes.

La première partie de notre projet aborde la modélisation du manipulateur, que ce soit le modèle géométrique et cinématique qui sont utilisés pour les différentes transformations entre l'espace articulaire et l'espace cartésien et vice versa, ou le modèle dynamique qui est un outil primordial pour l'élaboration des différents types de commande, c'est pour cette raison, que les incertitudes du modèle dynamique affectent fortement la stabilité des systèmes.

Dans la deuxième partie les commandes en effort sont présentées et classées en deux grandes familles [27], celles qui contrôlent explicitement la force ce qu'on appelle la commande en force directe, dans cette catégorie les deux principaux types sont, premièrement la commande hybride force/position parallèle basée sur la subdivision de l'espace cartésien en deux directions celle qui doit être contrôlée en force et l'autre en position selon qu'elle soit contraint ou non,

Nous avons constaté lors de la simulation de cette commande une bonne poursuite de la trajectoire désirée dans la partie de l'espace où le robot est libre, et une bonne stabilisation de la force lorsque le robot entre en contact avec l'environnement ;

La commande hybride force/position série, est basée sur la déviation de la position engendrée par l'inverse de la matrice de raideur, multipliée par la force d'interaction du robot avec son environnement ;

L'inconvénient de ces types de commande c'est le besoin d'une connaissance parfaite de l'environnement ce qui n'est pas toujours accessible. Cette commande nécessite aussi le développement de deux lois de commande en position et en force.

La deuxième catégorie c'est la commande de la force indirecte dont la force est commandée implicitement, la commande en impédance est le type le plus intéressant dans cette catégorie,

## Conclusion Générale

elle est basée sur le principe de changement de la dynamique du robot afin qu'il se comporte comme un ensemble de masse-ressort-amortisseur vis-à-vis l'environnement.

Les simulations avec l'application des commandes non linéaires ont montré des bons résultats, le Backstepping a donné de bon résultat en absence des perturbations, en terme de stabilité de la position et aussi pour la force d'interaction, la commande linéarisante avec sa simplicité d'implémentation a donné des résultats satisfaisants, néanmoins elle nécessite un modèle précis, ce qui rend cette commande insuffisante en présence de moindre perturbation ;

La commande par mode de glissement montre une bonne robustesse vis-à-vis les perturbations externes. La simulation de chute d'une charge de masse  $m_{per}$  a montré l'efficacité de ce type de commande en présence des perturbations.

Les points essentiels qui devront pris en compte lors de l'élaboration des lois de commande sont :

- L'identification des paramètres optimaux pour l'élaboration des lois de commande répondants aux exigences de contrôle.
- Les limites des couples fournies par les actionneurs.
- L'observation des états du robot nécessaires pour le calcul du modèle dynamique avec précision.

A partir de ces points nous tirons les perspectives suivantes :

- L'utilisation des méthodes d'optimisation et - surtout l'intelligence artificielle - dans l'identification des paramètres des lois de commande répondant ainsi aux exigences de contrôle en terme de stabilité, précision et robustesse ; ces méthode doivent prendre en considération les limites des actionneurs.
- L'utilisation des méthodes d'observation efficaces afin de concevoir un modèle précis.
- L'utilisation de l'intelligence artificielle.
- Test expérimental des résultats obtenus.



# ANNEXE A

*Cette annexe expose le mode glissant d'ordre 2 et d'ordre supérieur*

## 1 Synthèse de la commande MGO2:

Dans le mode glissant classique :

- Dans la commande mode glissant conventionnelle la surface doit être d'ordre relatif 1 par rapport à la commande (apparition de la commande dans la 1<sup>ère</sup> dérivée de la surface de glissement  $s$ ) pour assurer sa convergence vers zéro.  
Supposant que la commande est apparait dans  $\ddot{s}$ , généralement la surface de glissement  $\sigma$  est choisit  $\sigma = \dot{s} + s$  ;  
 $\dot{\sigma} = \ddot{s} + \dot{s}$  Contient la commande donc  $\sigma \rightarrow 0$ , mais  $s$  tend vers Zéro **asymptotiquement**.
- L'effet du chattering est inacceptable physiquement.

Ces deux restrictions ont une solution dans la commande mode glissant d'ordre 2.

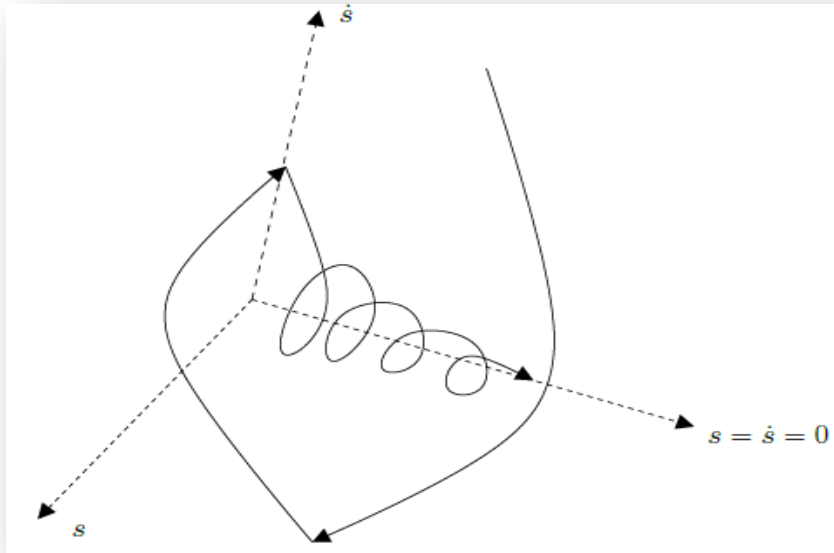
Considérons un système non linéaire décrit par

$$\begin{cases} \dot{x} = f(x, t) + g(x, t)u \\ s = s(x, t) \end{cases} \quad (\text{A.1})$$

Avec  $x \in \mathbb{R}$  le vecteur d'états,  $u \in \mathbb{R}$  entrée du système.  $f(x, t), g(x, t)$  sont des fonctions incertaines, suffisamment différentiables et  $s(x, t) \in \mathbb{R}$  est la sortie, aussi suffisamment différentiable.

L'objectif de la commande par MGO2 est de forcer la variable de glissement et sa première dérivée à zéro. Dans le cas où le degré relatif du système par rapport à la variable de glissement est 1 ou 2,  $s$  et  $\dot{s}$  convergent vers zéro en temps fini. La convergence à l'origine est présentée sur la Figure A.1 [23]

La synthèse des lois de commande par MGO2 dépend des bornes de fonctions incertaines, qui apparaissent dans la deuxième dérivée de la variable de glissement. Deux cas sont possibles, en fonction du degré relatif du système par rapport à  $s$ . Les degrés relatifs sont d'ordre 1 ou 2.



**Figure A.1.** Trajectoire 2-glissant.

**Cas A : système de degré relatif 1 :**

(L'utilité du MGO2 ici est d'éliminer la réticence) Dans le cas où le degré relatif du système est égale à 1 par rapport à  $s$ , la commande  $u$  apparaît dans la première dérivée de  $s$  par rapport au temps,

$$\dot{s}(x, t) = \frac{\partial s}{\partial t} \dot{t} + \frac{\partial s}{\partial x} \dot{x}$$

$$\dot{s}(x, t) = \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} (f + g u)$$

D'où  $\ddot{s}$  s'écrit comme suite :

$$\ddot{s} = a(x, u, t) + b(x, u, t) \dot{u} \quad (\text{A.2})$$

Avec  $a(x, u, t), b(x, u, t)$  des fonctions incertaines bornées, dans un voisinage de la surface de glissement, défini par  $|s(x, t)| < s_0$  [23].

$$\begin{aligned} 0 < K_m < b(x, u, t) < K_M \\ |a(x, u, t)| &\leq C \end{aligned} \quad (\text{A.3})$$

Selon ces bornes, on peut écrire par l'inclusion différentielle suivante :

$$\ddot{s} \in [-C, C] + [K_m, K_M] \dot{u} \quad (\text{A.4})$$

**Cas B : Système de degré relatif 2 :**

Dans le cas où le système est de degré relatif 2, la commande  $u$  apparaît dans la deuxième dérivée de  $s$ , celle-ci est donnée comme suit :

$$\ddot{s}(x, t) = \frac{\partial \dot{s}}{\partial t} + \frac{\partial}{\partial x} \dot{s} (f + g u)$$

Puis, la deuxième dérivée de  $s$  peut être écrite sous la forme suivante :

$$\ddot{s} = a'(x, u, t) + b'(x, u, t) u \quad (\text{A.5})$$

Avec  $a'(x, u, t), b'(x, u, t)$  des fonctions incertaines bornées, De même que dans les systèmes de degré relatif 1,

$$\begin{aligned} 0 < K_m' < b'(x, u, t) < K_M' \\ |a'(x, u, t)| \leq C' \end{aligned} \quad (\text{A.6})$$

$$\ddot{s} \in [-C', C'] + [K_m', K_M'] u \quad (\text{A.7})$$

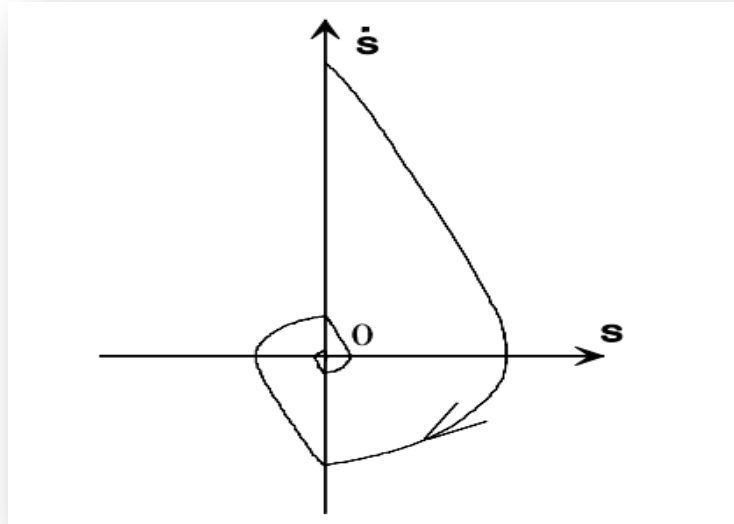
Les deux relations (A.4) et (A.7), sont très importantes pour l'étude de la robustesse d'un système, dans le but d'assurer la convergence en temps fini de  $s$  et  $\dot{s}$  vers zéro.

Plusieurs types d'algorithmes du mode glissant d'ordre 2 présentés dans la littérature, on va citer les méthodes principales :

### 1.1 Algorithme du Twisting (TWG):

Historiquement cet algorithme est le premier mode glissant d'ordre 2, son principe repose sur la commutation de l'amplitude de la commande entre deux valeurs en fonction du quadrant dans lequel se trouve l'état du système. [23] La trajectoire du système dans le plan de phase tourne autour de l'origine avec convergence. L'expression de la loi de commande pour un système de degré relatif 2 est :

$$u = -(r_1 \text{sign}(s) + r_2 \text{sign}(\dot{s})) \quad \text{Avec} \quad r_1 > r_2 > 0 \quad (\text{A.8})$$



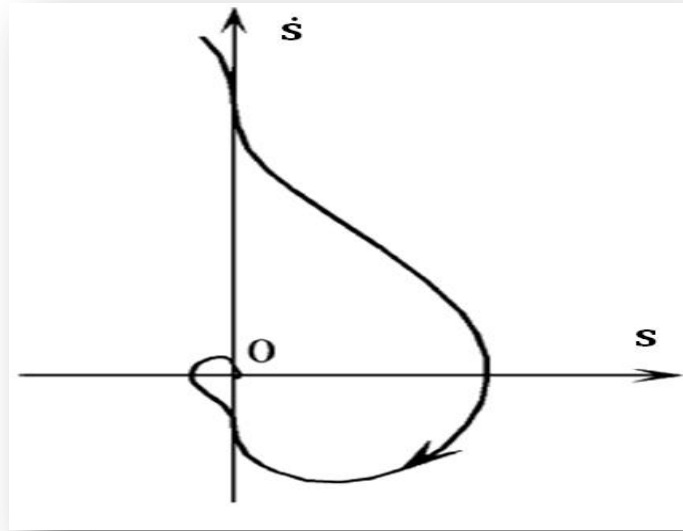
**Figure A.2.** Convergence de l'algorithme Twisting dans le plan  $(s, \dot{s})$ .

La trajectoire du système différentiel converge au point d'équilibre  $s = \dot{s} = 0$  en un temps fini sous les conditions suivantes :

$$\begin{aligned} (r_1 + r_2) K_m - C_0 &> (r_1 - r_2) K_M + C_0 \\ (r_1 - r_2) K_m &> C_0 \end{aligned} \tag{A.9}$$

## 1.2 Algorithme du super-Twisting :

Cette loi de commande a été proposée par Emelyanov en 1990 ; Cet algorithme ne s'applique qu'aux systèmes de degré relatif 1 par rapport à la surface de glissement. L'algorithme du Super-Twisting figure parmi les algorithmes de modes glissants d'ordre deux les plus utilisés, son intérêt réside dans la réduction du chattering, dû à la continuité du signal de commande.



**Figure A.3.** La trajectoire de la commande super-Twisting dans le plan  $(s, \dot{s})$ .

L'implémentation de cet algorithme passe par les étapes suivantes [7]

Supposant le système est du 2<sup>ème</sup> ordre .

- définition de la surface de glissement comme :

$$s = \dot{x} + cx$$

- la dérivée de cette surface est donnée :

$$\dot{s} = \ddot{x} + c\dot{x}$$

- la substitution de  $\ddot{x}$  par son expression donc:

$$\dot{s} = f + u + c\dot{x} \tag{A.10}$$

- la loi de commande est :

$$u = -c\dot{x} - \rho_1 |s|^{1/2} \text{sign}(s) - \int_0^t \rho_2 \text{sign}(s) d\tau \tag{A.11}$$

Par la substitution du (11) dans (10) on trouve :

$$\begin{cases} \dot{s} = -\rho_1 |s|^{1/2} \text{sign}(s) + v \\ \dot{v} = -\rho_2 \text{sign}(s) + \dot{f} \end{cases} \quad \rho_1, \rho_2 > 0 \tag{A.12}$$

Pratiquement nous n'avons pas les valeurs de  $x, \dot{x}$  mais ses estimées  $\hat{x}, \hat{\dot{x}}$  donc cette dernière loi devient :

$$u = -c\hat{\dot{x}} - \rho_1 |\hat{s}|^{1/2} \text{sign}(\hat{s}) - \int_0^t \rho_2 \text{sign}(\hat{s}) d\tau \tag{A.13}$$

$$\text{Avec } \hat{s} = \hat{x} + c \hat{x}$$

### 1.3 Algorithme sous-optimal (Algorithme Sub optimal) :

L'algorithme sous-optimal est appliquée aux systèmes de degré relatif 1 et 2 par rapport à  $s(x, t)$ . Cet algorithme est inspirée de la commande optimale de type bang-bang. Il assure la convergence de  $s$  et  $\dot{s}$  en temps fini vers zéro, en limitant les trajectoires par des arcs paraboliques qui comprennent l'origine [25] (Figure A.4). Dans le cas où le degré relatif de système est égale à 2 par rapport à  $s(x, t)$ , la loi de commande est définie par :

$$u = -r_1 \text{sign}\left(s - \frac{s^*}{2}\right) + r_2 \text{sign}(s^*) \quad (\text{A.14})$$

Où  $r_1 - r_2 > \frac{c}{K_m}$        $r_1 + r_2 > \frac{4c + K_M(r_1 - r_2)}{3K_m}$

$s^*$  : c'est la valeur de  $s$  détectée dans le dernier moment où  $s$  était égal à zéro.

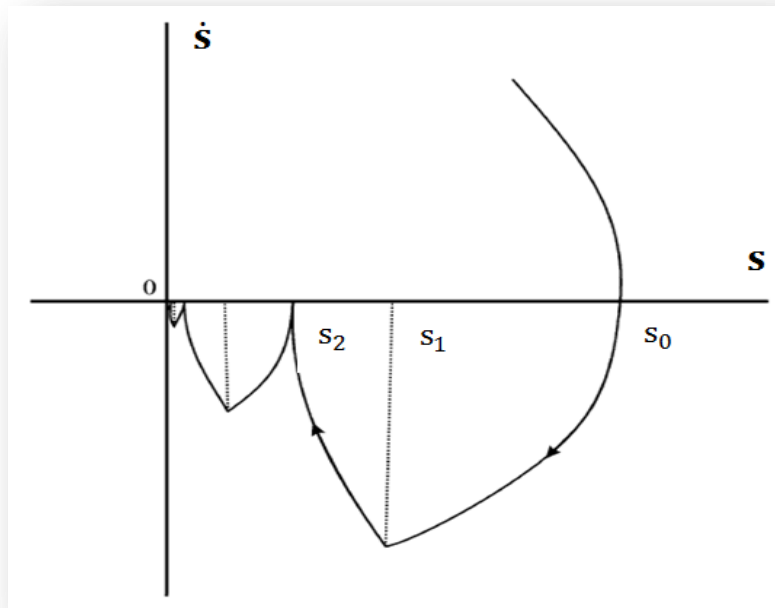


Figure A.4. Principe de l'algorithme sous-optimal.

### 1.4 Algorithme prescribed convergence law (PCL):

Cet algorithme est caractérisé par

- sa simplicité de mise en œuvre.
- un besoin d'une quantité d'information relativement faible sur l'état du système [38].

La formulation générale de ce contrôleur est donnée par l'expression suivante :

$$u = -\alpha \operatorname{sign}(\dot{s} - \xi(s)) \quad \alpha > 0$$

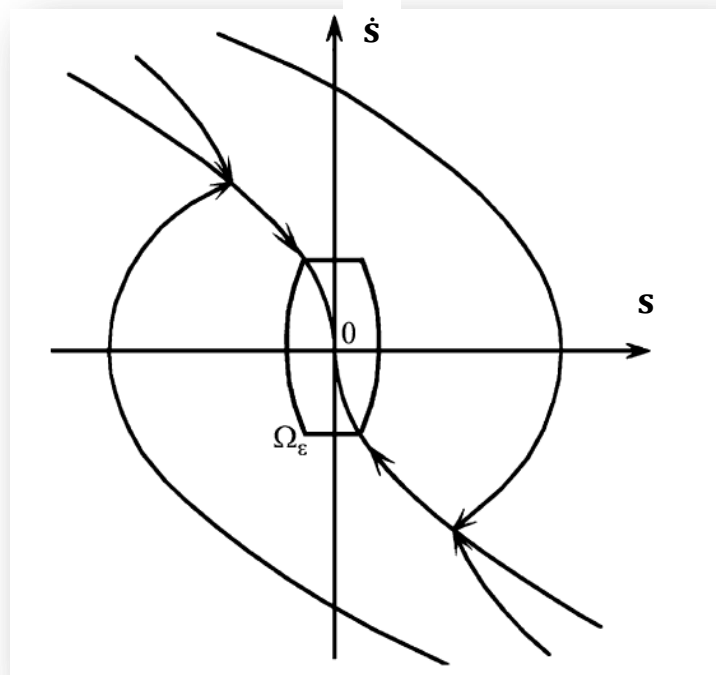
$\xi(s)$  Une fonction continue lisse partout sauf pour  $s = 0$ , définie par :

$$\xi(s) = \beta |s|^{\frac{1}{2}} \operatorname{sign}(s)$$

On écrit la commande comme suite :

$$u = -\alpha \operatorname{sign}\left(\dot{s} - \beta |s|^{\frac{1}{2}} \operatorname{sign}(s)\right) \quad \alpha, \beta > 0 \quad (\text{A.15})$$

Une condition suffisante pour une convergence en temps fini vers la surface de glissement est définie par la relation suivante :  $\alpha K_m - C_0 > \beta^2/2$



**Figure A.5.** Principe de l'algorithme prescribed convergence law.

## 2 Mode glissant d'ordre supérieur :

La commande mode glissant d'ordre supérieur a été introduite par Emel'yanov [13]. Elle représente une extension du mode glissant d'ordre 1 à un degré plus élevé. Cette généralisation conserve la principale caractéristique de robustesse que celle du mode glissant

classique, et permet aussi de réduire leur principal inconvénient : l'effet de réticence au voisinage de la surface de glissement.

L'extension des modes glissants d'ordre un aux modes glissants d'ordre supérieur est caractérisée par le choix d'une commande discontinue agissant non pas sur la surface de glissement mais sur ses dérivées supérieures.

L'inconvénient principal de la commande en mode glissant d'ordre supérieur réside dans le besoin de connaître les variables d'état et leurs dérivées. Il en résulte alors que pour la synthèse d'un contrôleur en modes glissants d'ordre  $r$ , les fonctions  $(s, \dot{s}, \ddot{s}, \dots, s^{(r-1)})$  sont nécessaires

Soit le système défini par l'équation suivante :

$$\begin{cases} \dot{x} = f(x, t) + g(x, t)u \\ y = s(x, t) \end{cases} \quad (\text{A.16})$$

Avec  $x \in \mathbb{R}^n$  le vecteur d'états,  $u \in \mathbb{R}$  entrée du système et  $s(x, t) \in \mathbb{R}$  est la sortie suffisamment différentiables.  $f(x, t), g(x, t)$  sont des fonctions incertaines, suffisamment différentiable.

L'objectif de la commande est de rendre la fonction de sortie  $s \equiv 0$  en temps fini. Selon le degré relatif  $r$  du système nous avons deux cas :

- degré relatif  $r = 1$  si et seulement si  $\partial \dot{s} / \partial u \neq 0$
- degré relatif  $r \geq 2$  si et seulement si  $\partial s^{(i)} / \partial u \neq 0$  ( $i = 1, 2, \dots, r - 1$ ), et  $\partial s^{(r)} / \partial u \neq 0$

L'idée de base d'une commande de mode glissant d'ordre supérieur est que la fonction discontinue agit sur un ordre supérieur (dérivées  $s^{(i)}$  ( $i = 1, 2, \dots, r - 1$ )) de la surface de glissement :

$$s(x, t) = \dot{s}(x, t) = \ddot{s}(x, t) = \dots = s^{(r-1)}(x, t) = 0 \quad (\text{A.17})$$

Supposant que le degré relatif du système (A.16) égale à  $r$ , d'une manière générale lorsque la commande apparaît dans la dérivée de  $s$  d'ordre  $r$  ( $s^{(r)}$ ), soit  $\partial s^{(r)} / \partial u \neq 0$ , donc nous prenons  **$r$  - dérivée de  $s$  comme sortie du système** (A.16), par la sélection d'une nouvelle coordination nous avons donc :

$$y = (y_1, y_2, \dots, y_r) = (s, \dot{s}, \ddot{s}, \dots, s^{(r-1)}) \quad (\text{A.18})$$

Donc la formule suivante est obtenue :

$$s^{(r)} = a(y, t) + b(y, t) \quad b(y, t) \neq 0 \quad (\text{A.19})$$



Donc la commande par mode glissant d'ordre supérieur est transformée à la stabilité d'un système dynamique d'ordre  $r$  (A.17) et (A.19).

Il est facile de confirmer la formule suivante par la dérivée de Lie :

$$\begin{aligned} b &= L_g L_f^{r-1} = ds^{(r)}/du \\ a &= L_f^r s \end{aligned} \quad (\text{A.20})$$

Supposant :  $\Phi = (y_{r+1}, y_{r+2}, \dots, y_n)$  donc :

$$\Phi = \xi(t, s, \dot{s}, \dots, s^{(r-1)}, \Phi) + \chi(t, s, \dot{s}, \dots, s^{(r-1)}, \Phi) u \quad (\text{A.21})$$

Les équations (A.18), (A.19) et (A.21) sont transformées à la forme canonique d'Isidori-Brunowsky, la commande équivalente est  $u_{eq} = -a(y, t)/b(y, t)$  [34].

En ce moment l'objectif de la commande est de construire une commande discrète

$u = U(x, t)$  de telle façon le nouveau système converge vers l'origine en temps fini, avec  $a(y, t)$  et  $b(y, t)$  des fonctions bornées telle que :

$$\begin{aligned} 0 < K_m < b(x, t) < K_M \\ |a(x, t)| &\leq C \end{aligned} \quad (\text{A.22})$$

Théorème : [22] supposant que le degré relatif du système (A.16) par rapport à la sortie  $s(x, t)$  est  $r$ , et satisfait la condition (A.22), donc la commande mode glissant d'ordre arbitraire est donnée par l'expression suivante :

$$u = -a \operatorname{sign}[\psi_{r-1,r}(s, \dot{s}, \dots, s^{(r-1)})] \quad (\text{A.23})$$

Où,

$$\psi_{0,r} = s$$

$$\psi_{1,r} = \dot{s} + \beta_1 N_{1,r} \operatorname{sign}(s)$$

$$\psi_{i,r} = s^{(i)} + \beta_i N_{i,r} \operatorname{sign}(\psi_{i-1,r}) \quad i = 1, \dots, r-1$$

$$N_{1,r} = |s|^{(r-1)/r}$$

$$N_{i,r} = (|s|^{p/r} + |\dot{s}|^{p/(r-1)} + \dots + |s^{(i-1)}|^{p/(r-i+1)})^{(r-1)/p} \quad i = 1, \dots, r-1$$

$$N_{r-1,r} = (|s|^{p/r} + |\dot{s}|^{p/(r-1)} + \dots + |s^{(r-2)}|^{p/2})^{1/p} \quad (\text{A.24})$$

Ici les lois de commande pour  $r \leq 4$

- si  $r = 1 \Rightarrow u = -a \operatorname{sign}(s)$
  - si  $r = 2 \Rightarrow u = -a \operatorname{sign}(\dot{s} + |s|^{\frac{1}{2}} \operatorname{sign}(s))$
  - si  $r = 3 \Rightarrow u = -a \operatorname{sign}(\ddot{s} + 2(|\dot{s}|^3 + |s|^2)^{\frac{1}{6}} \operatorname{sign}(\dot{s} + |s|^{\frac{2}{3}} \operatorname{sign}(s)))$
  - si  $r = 4 \Rightarrow \mathbf{u} = -a \operatorname{sign}\{ \ddot{\ddot{s}} + 3(\ddot{s}^6 + \dot{s}^4 + |s|^3)^{1/12} \operatorname{sign}[\ddot{s} + (\dot{s}^4 + |s|^3)^{1/6} + \operatorname{sign}(\dot{s} + \frac{1}{2}|s|^{3/4} \operatorname{sign}(s))] \}$
- (A.25)

On constate de la dernière équation (A.25) que lorsque  $r = 1$  la commande est celle d'un mode glissant conventionnel, et quand  $r = 2$  la commande est équivalente à celle du Super-twisting second mode glissant.

La généralisation des modes glissants d'ordre 2 aux modes glissants d'ordre supérieur, est caractérisée par le choix d'une commande discontinue agissant non pas sur la surface de glissement et sa 1<sup>ère</sup> dérivée mais sur ses dérivées supérieures, malgré les performances qu'ils manifestent, ils sont pénalisés par l'information élevée dont ils ont besoin sur les variables du système, ainsi que par le choix de la valeur des gains de commande.

Les dérivées de surface sont estimées par le différentiateur (observateur), et le problème du choix des gains de la commande, il suffit de connaître les bornes supérieures des perturbations et des incertitudes du système ainsi que leurs dérivées.

## ANNEXE B

Les paramètres du robot choisi sont : [8]

$$L_1 = 1m$$

$$L_{c_1} = 0.5m$$

$$I_1 = 1kgm^2$$

$$m_1 = 10kg$$

$$L_2 = 0.5m$$

$$L_{c_2} = 0.25m$$

$$I_2 = 1kgm^2$$

$$m_2 = 10kg$$

## Bibliographies:

- [1] « Force-Impedance Control: a new control strategy of robotic manipulators»  
Fernando Almeida, Antonio Lopes, Paulo Abreu  
IDMEC - Polo FEUP, Faculdade de Engenharia da Universidade do Porto,  
Portugal.
- [2] <http://www.wikipedia.org/>
- [3] La norme A.F.N.O.R NF.E61-100 Normes françaises.
- [4] « Méthodes de commande par mode de glissement appliquées au bras manipulateur et au système hydraulique CE105 à réservoirs couplés » Par M. BOUBAKIR Ahcene, Pour l'obtention du diplôme Magister en automatique à L'EMP – ALGER 2007.
- [5] L.Al-bassit, « Structures mécaniques a modules sphériques optimisées pour un robot médical de télé-échographie mobile », Thèse de doctorat, Université d'Orléans.2005.
- [6] « Théorie Générale des Systèmes Articulés Introduction à la Robotique »  
André PREUMONT 2nde Edition – 2001.
- [7] « How to Implement Super-Twisting Controller based on Sliding Mode Observer» Asif Chalanga1VSS14, Nantes, June 29 July 2 2014.
- [8] Robust Impedance Control of Robot Manipulators S.P. Chan, Member, IEEE, and B. Yao - International Journal of Robotics and Automation Vol. 6, No. 4, 1991.
- [9] « Generalized Impedance Control of Robot for Assembly Tasks Requiring

Compliant Manipulation » S . P. Chan, Member, IEEE, and H. C. Liaw -  
IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 43,  
NO. 4, AUGUST 1996,page 454.

- [10] « Sliding mode control Theory and applications » Christopher Edwards and Sarah K. Spurgeon 1998.
- [11] “Introduction to Robotics Mechanics and Control” Third Edition PEARSON 2005 -- John J. Craig.
- [12] Modeling, Performance Analysis and Control of Robot Manipulators Edited by Etienne Dombre Wisama Khalil 2007 iSTE.
- [13] Emelyanov, S.V.; Korovin, S.K. & Levantovskiy, A. (1990). New class of second order sliding algorithm. Mathematical modeling, Vol.2, No.3, pp. 85-100.
- [14] Thème « Backstepping adaptatif pour le contrôle la poursuite et la synchronisation des systèmes dynamiques non linéaires chaotiques » Par Mme Laarem GUESSAS Pour l’obtention du diplôme de Doctorat es sciences Université Ferhat ABBAS – Sétif 2013.
- [15] Impedance Control: An Approach to Manipulation: Part 1-Theory--Part II-Implementation--Part III-Applications /Neville Hagan 1985.
- [16] “Impedance and Interaction Control” Neville Hogan Massachusetts Institute of Technology Stephen P. Buerger Massachusetts Institute of Technology.2005.
- [17] Notes de cours GPA546 Ilian Bonev, ing. Yanick Noiseux, ing. 20 décembre 2013.

- [18] « ROBOTIQUE » Jean-Louis Boimond Université Angers.
- [19] Robust Compliant Motion for Manipulators, Part I: The Fundamental Concepts of Compliant Motion H. KAZEROONI, T. B. SHERIDAN, FELLOW, IEEE, AND P. K. HOUPY.
- [20] Modeling Identification and Control of Robots Wisama Khalil·Etienne Dombre 2004.
- [21] Bases de la modélisation et des robots de la robots-manipulateurs de type série commande Wisama KHALIL, Etienne DOMBRE. Date de création : 08 mai 2012
- [22] Levant A. (2003). "Higher-order sliding modes, differentiation and output feedback control." International Journal of Control, Vol.76, No.9, pp.924-941, ISSN: 0020- 7179.
- [23] Levant A. (1993). "Sliding order and sliding accuracy in sliding mode control". International Journal of Control, Vol. 58, No.6, pp.1247-1263, ISSN: 0020-7179.
- [24] Compliance and Force Control for Computer Controlled Manipulators MATTHEW T. MASON 1981.
- [25] Thèse de Doctorat « Contribution à la commande non-linéaire robuste des systèmes d'alimentation en air des piles à combustible de type PEM » Imad MATRAJI - Université De Technologie BELFORT-MONTBÉLIARD 2013.

- [26] Miroslav Krstić and Petar V. Kokotović "Adaptive Nonlinear Output-Feedback Schemes" with Marino-Tomei Controller, *Ieee Transactions on Automatic Control*, VOL. 41, NO.2, February 1996, pp274-280.
- [27] Perdereau V « Contribution à la commande hybride force-position ». Thèse de Doctorat, University Pierre et Marie Curie, Paris, France, February 1991.
- [28] Raibert M.H., Craig J.J., "Hybrid Force-Position Control of Manipulators", *Trans.of the ASME, Journal of Dynamic Systems, Measurement and Control*, vol. 103, June 1981, pp. 126-133.
- [29] Intelligent control of manipulators interacting with an uncertain environment based on generalized impedance Sukhan Lee and Hahk Sung Lee / *Proceedings of the 1991 IEEE International Symposium of Intelligent control / 13 - 15 August 1991. Arlington. Virginia. USA.*
- [30] «Active stiffness control of a manipulator in cartesian coordinates» J. Kenneth Salisbury 1980 IEEE
- [31] "Applied NonLinear control" JEAN-JACQUES E. SLOTINE And WEIPING LI Massachusetts Institute of Technology 1991 Prentice Hall Englewood Cliffs, New Jersey 07632
- [32] *Robot dynamics and control* Second Edition. Mark W. Spong, Seth Hutchinson, and M. Vidyasagar.
- [33] Tadej Bajd .Matjaž Mihelj . Jadran Lenarcic . Aleš Stanovnik . Marko Munih *Robotics* .Springer 2010.

- [34] Vadim I. Utkin. "Sliding modes in control optimization." Communication and Control engineering series. Springer-Verlag1992.
- [35] "On the Equivalence of Second Order Impedance Control and Proportional Gain Explicit Force Control" Richard Volpe Pradeep Khosla Carnegie Mellon University1995, pkhosla@cmu.edu page 6.
- [36] D. E. Whitney, "Force feedback control of manipulator fine motions", Journal of dynamic systems, measurement and control, June 1977, pp.91-97.
- [37] "Historical perspective and state of the art in robot force control" 1985 IEEE.
- [38] "Sliding Mode Control and Observation" Yuri Shtessel, Christopher Edwards, Leonid Fridman, Arie Levant 2014 **Birkhäuser**.