

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
FACULTE DES SCIENCES DE L'INGENIEUR  
DEPARTEMENT DE GENIE MECANIQUE



MEMOIRE POUR L'OBTENTION DE DIPLOME DE MAGISTER  
SPECIALITE: GENIE MECANIQUE  
Option: ROBOTIQUE

THEME

**COMMANDE LINEARISANTE DU ROBOT PUMA 560  
AVEC ANALYSE DE LA ROBUSTESSE**

Présenté par  
**MSALAK Tawfiq**

*Jury:*

<i>Mr. ELAJRAMI</i> Mohamed	Pr.	U.D.L	S. B.A	President
<i>Mr. BELABBES</i> Bagdad	MCA	U.D.L	S. B.A	Encadreur
<i>Mr. REFASSI</i> Kaddour	Pr.	U.D.L	S. B.A	Examineur
<i>Mr. MEGUENI</i> Abdelkader	Pr.	U.D.L	S. B.A	Examineur
<i>Mr. YOUNES</i> Mimoun	Pr.	U.D.L	S. B.A	Examineur

**18-Décembre-2014**

## **REMERCIEMENTS**

J'exprime mes profonds remerciements à monsieur. **B.BELABBES MCA** à l'université Djillali Liabes pour leur l'encadrement, et pour l'aide compétente qu'il m'a apportée, pour sa patience et son encouragement à finir ce travail, ses conseils et sa confiance à mon égard, notamment lors des moments difficiles et de doutes, son oeil critique m'a été très précieux pour structurer et pour améliorer la qualité des différentes sections de ce travail.

La seconde personne sans laquelle ce travail n'aurait pas pu être mené à bien est sans nul doute monsieur **A.AZEDIN**, je le remercie sincèrement d'avoir accepté notre invitation.

J'exprime ma sincère gratitude à Monsieur **ELAJRAMI Mohamed P.r** de l'Université Djillali Liabes, qui m'a fait l'honneur de présider le jury de cette thèse.

Je remercie Messieurs, le professeur **K. REFASSI, A.MEGUENI et M.YOUNES** Professeurs à l'Université Djillali Liabes d'avoir accepté d'examiner ce travail.

Mes remerciements s'adressent à mon entourage proche, mes amis et ma famille qui ont toujours été à mes côtés et dont le soutien m'a été indispensable.

## **Résumé**

*Le but de ce travail est d'analyser les performances de commandes classiques PD et à couple calculé dans ses deux performances vis à vis des incertitudes sur le modèle dynamique du manipulateur robotique. Des perturbations sur les paramètres du modèle, masses des liaisons, positions des centres de masse, moments d'inertie des axes principaux, ainsi que l'effet d'une force extérieure, sur les trajectoires de commande en mode consigne ou poursuite de trajectoire. La robustesse est mesurée en termes de l'amplitude de l'erreur globale des graphes d sortie par rapport aux trajectoires nominales correspondant au cas non perturbé.*

**Keywords:** Computed torque control, PD control, joint control, robustness, dynamic model

**Mots Clefs :** Model dynamique, robustesse, commande PD, PID à couple calculé, commande articulaire.

# ***TABLES DES MATIERES***

<b>INTRODUCTION.....</b>	<b>01</b>
<b>I. Généralités Sur Les Systèmes Robotiques.....</b>	<b>05</b>
<b>I.1. Définition [1].....</b>	<b>06</b>
<b>I.2. Organes Constitutifs d'un Robot.....</b>	<b>06</b>
<b>I.2.1. Le système mécanique articulé (S.M.A.).....</b>	<b>06</b>
<b>I.2.2. Les actionneurs.....</b>	<b>07</b>
<b>I.2.3. Le système de perception.....</b>	<b>08</b>
<b>I.2.4. La partie commande.....</b>	<b>08</b>
<b>I.3. Classification des robots.....</b>	<b>08</b>
<b>I.3.1. Les manipulateurs.....</b>	<b>08</b>
<b>I.3.2. Les télémanipulateurs.....</b>	<b>08</b>
<b>I.3.3. Les robots.....</b>	<b>09</b>
<b>I.3.4. Les générations de robot.....</b>	<b>09</b>
<b>I.4. Caractéristiques d'un robot.....</b>	<b>10</b>
<b>I.5. Programmation des robots.....</b>	<b>10</b>
<b>I.5.1. L'apprentissage.....</b>	<b>10</b>
<b>I.5.2. La génération de trajectoires .....</b>	<b>11</b>
<b>I.6. Morphologie des robots Manipulateurs.....</b>	<b>11</b>
<b>II. Modélisation par les paramètres DH standards et modifiés.....</b>	<b>15</b>
<b>➤ La représentation de Denavit-Hartenberg standard et modifiée.....</b>	<b>16</b>
<b>II.1. La représentation de Denavit-Hartenberg standard .....</b>	<b>16</b>
<b>II.1.1. Algorithme pour placer les référentiels DH.....</b>	<b>17</b>
<b>II.1.2. Algorithme pour obtenir les paramètres DH.....</b>	<b>18</b>
<b>II.1.3. Equation de la cinématique directe.....</b>	<b>19</b>
<b>II.1.4. Exemple d'un robot sériel à cinq articulations rotoïdes.....</b>	<b>19</b>
<b>II.2. La représentation de Denavit-Hartenberg modifiée .....</b>	<b>23</b>
<b>II.2.1. Relations géométriques.....</b>	<b>25</b>
<b>II.2.2. Comparaison entre les deux représentations.....</b>	<b>27</b>
<b>III. Dynamique d'un système robotique.....</b>	<b>28</b>
<b>III.1. Approche par les équations de Lagrange.....</b>	<b>29</b>
<b>III.1.1.Introduction.....</b>	<b>29</b>
<b>III.1.2.Forme générale des équations dynamiques.....</b>	<b>30</b>
<b>III.1.3.Calcul des éléments de M, V et G.....</b>	<b>31</b>
<b>III.1.4.Prise en compte des frottements.....</b>	<b>32</b>
<b>III.1.5.Prise en compte des inerties des actionneurs.....</b>	<b>32</b>
<b>III.1.6.prise en compte des efforts exercés par l'organe terminal sur     l'environnement.....</b>	<b>33</b>
<b>III.1.7.Prise en compte des flexibilités des articulations.....</b>	<b>33</b>
<b>III.1.8.Modélisation des flexibilités.....</b>	<b>33</b>
<b>III.1.9.Application de la formulation de Lagrange sur le robot PELICAN .....</b>	<b>34</b>
<b>III.2. Approche par les équations de EULER –NEWTON.....</b>	<b>42</b>
<b>III.2.1.Equation de EULER –NEWTON.....</b>	<b>42</b>
<b>III.2.2.Équation de Newton.....</b>	<b>42</b>

III.2.3.	Équation d'Euler.....	42
III.2.4.	Formulation itérative de Newton-Euler.....	43
III.2.4.1.	Itérations vers l'extérieur pour calculer les vitesses et accélérations.....	43
III.2.4.2.	La force et le couple agissant sur un lien.....	44
III.2.4.3.	Itérations intérieures pour le calcul des forces et des couples.....	44
III.2.4.4.	L'algorithme dynamique itératif de Newton-Euler.....	46
III.2.4.5.	Prise en compte des forces de gravité dans l'algorithme dynamique.....	47
III.2.5.	Application de la formulation de Newton-Euler sur un manipulateur à deux-lien (2DDL).....	47
IV.	Dynamique de l'actionneur.....	52
IV.1.	Dynamique de l'actionneur.....	53
IV.2.	Dynamique complète du système robotique : Actionneur + Robot.....	58
V.	Commande Linéarisante à couple calculé (Computed Torque control).....	61
V.1.	Principe.....	61
V.2.	Commandes classiques PD.....	61
V.2.1.	Contrôle proportionnel avec rétroaction de vitesse.....	61
V.2.2.	Contrôle proportionnel-dérivé (PD).....	62
V.2.3.	La gravité n'est pas prise en compte.....	64
V.3.	Commandes par couple calculé.....	65
V.3.1.	Loi de commande.....	65
V.3.2.	Observateur de Perturbation.....	68
V.3.3.	Choix des paramètres PD.....	68
V.4.	Commande PID. Réjection des perturbations.....	69
V.4.1.	Commande PID.....	69
VI.	Fenêtre graphique matlab et simulation.....	72
VI.1.	Description De La Fenêtre.....	73
VI.1.1.	Aperçu de la Fenêtre.....	73
VI.1.2.	Barre des menus.....	73
VI.1.2.1.	Menu Fichier.....	73
VI.1.2.2.	Menu Méthodes.....	74
VI.1.2.3.	Menu Robots.....	74
VI.1.2.4.	Menu Trajectoires.....	74
VI.1.2.5.	Menu Aide.....	74
VI.1.3.	Barre des objets.....	74
VI.1.4.	Déroulement d'une simulation.....	75
VI.2.	Simulation.....	76
VI.2.1.	Commande PD avec compensation de la gravité.....	76
VI.2.2.	Commande à couple calculé.....	80
VI.2.3.	Table récapitulative de l'évolution des erreurs en fonction de la nature et de l'amplitude de la perturbation pour le robot PELICAN.....	83
1.	Commande PD.....	83
1.1.	commande consignes PD.....	83
1.2.	commande trajectoires PD.....	84

<b>2. Commande PID.....</b>	<b>85</b>
<b>2.1. commande consignes PID.....</b>	<b>85</b>
<b>2.2. commande trajectoires PID.....</b>	<b>85</b>
<b>3. Effet de perturbation.....</b>	<b>86</b>
<b>VI.3. Conclusion.....</b>	<b>87</b>
<b>Conclusion.....</b>	<b>90</b>
<b>Annexes.....</b>	<b>91</b>
<b>a. Annexes a.....</b>	<b>92</b>
<b>b. Annexes b.....</b>	<b>99</b>
<b>c. Annexes c.....</b>	<b>107</b>
<b>d. Annexes d.....</b>	<b>109</b>
 <b>Bibliographie.....</b>	 <b>115</b>

# ***CHAPITRE I***

## ***Généralités Sur Les Systèmes Robotiques***



## I.7. Définition [1]

La robotique peut être définie comme l'ensemble des techniques et études tendant à concevoir des systèmes mécaniques, informatiques ou mixtes, capables de se substituer à l'homme dans ses fonctions motrices, sensorielles et intellectuelles.

En fait, l'image que chacun s'en fait est généralement vague, souvent un robot est défini comme un manipulateur automatique à cycles programmables. Pour "mériter" le nom de robot, un système doit posséder une certaine flexibilité, caractérisée par les propriétés suivantes

- ✓ La versatilité : Un robot doit avoir la capacité de pouvoir exécuter une variété de tâches, ou la même tâche de différente manière ;
- ✓ L'auto-adaptativité : Un robot doit pouvoir s'adapter à un environnement changeant au cours de l'exécution de ses tâches.

L'Association Française de Normalisation (A.F.N.O.R.) définit un robot comme étant "*un système mécanique de type manipulateur commandé en position, reprogrammable, polyvalent (i.e., à usages multiples), à plusieurs degrés de liberté, capable de manipuler des matériaux, des pièces, des outils et des dispositifs spécialisés, au cours de mouvements variables et programmés pour l'exécution d'une variété de tâches*". Il a souvent l'apparence d'un, ou plusieurs, bras se terminant par un poignet. Son unité de commande utilise, notamment, un dispositif de mémoire et éventuellement de perception et d'adaptation à l'environnement et aux circonstances. Ces machines polyvalentes sont généralement étudiées pour effectuer la même fonction de façon cyclique et peuvent être adaptées à d'autres fonctions sans modification permanente du matériel.

## I.8. Organes Constitutifs d'un Robot

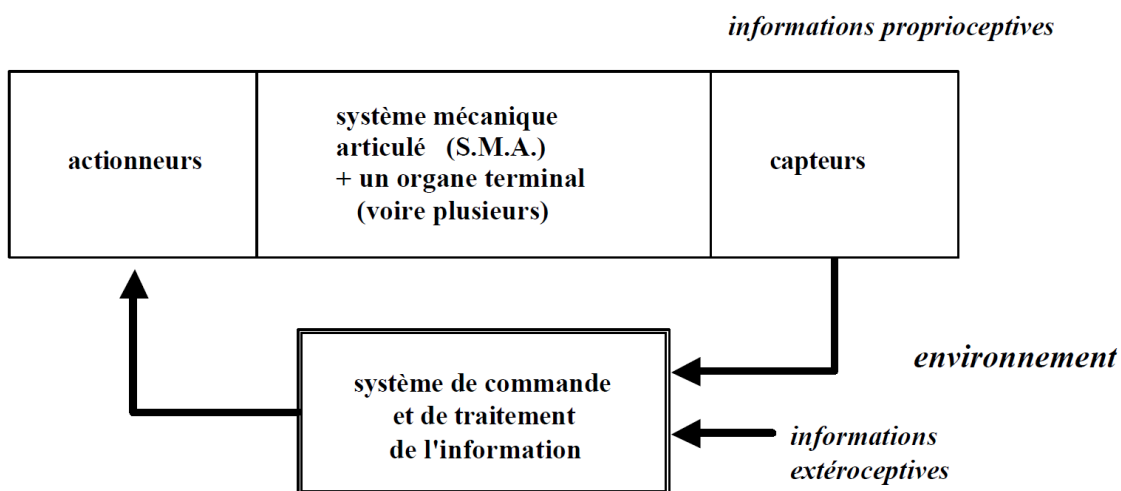


Figure I.1

### I.8.1. Le système mécanique articulé (S.M.A.)

C'est un mécanisme ayant une structure plus ou moins proche de celle du bras humain. Il permet de remplacer, ou de prolonger, son action (le terme "manipulateur" exclut implicitement les robots

mobiles autonomes). Son rôle est d'amener l'organe terminal dans une situation (position et orientation) donnée, selon des caractéristiques de vitesse et d'accélération données. Son architecture est une chaîne cinématique de corps, généralement rigides (ou supposés comme tels), assemblés par des liaisons appelées articulations. Sa motorisation est réalisée par des actionneurs électriques, pneumatiques ou hydrauliques qui transmettent leurs mouvements aux articulations par des systèmes appropriés.

- **Articulation rotoïde:** Il s'agit d'une articulation de type pivot, notée R, réduisant le mouvement entre deux corps à une rotation autour d'un axe qui leur est commun. La situation relative entre les deux corps est donnée par l'angle autour de cet axe



**Figure I.2**

- **Articulation prismatique :** Il s'agit d'une articulation de type glissière, notée P, réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe (voir la figure suivante).



**Figure I.3**

### **I.8.2. Les actionneurs**

Pour être animé, le S.M.A. comporte des moteurs le plus souvent avec des transmissions (courroies crantées), l'ensemble constitue les actionneurs. Les actionneurs utilisent fréquemment des moteurs électriques à aimant permanent, à courant continu, à commande par l'induit (la tension n'est continue qu'en moyenne car en général l'alimentation est un hacheur de tension à fréquence élevée ; bien souvent la vitesse de régime élevée du moteur fait qu'il est suivi d'un réducteur, ce qui permet d'amplifier le couple moteur). On trouve de plus en plus de moteurs à commutation électronique (sans balais), ou, pour de petits robots, des moteurs pas à pas. Pour les robots devant manipuler de

très lourdes charges (par exemple, une pelle mécanique), les actionneurs sont le plus souvent hydrauliques, agissant en translation (vérin hydraulique) ou en rotation (moteur hydraulique). Les actionneurs pneumatiques sont d'un usage général pour les manipulateurs à cycles (robots toutou rien). Un manipulateur à cycles est un S.M.A. avec un nombre limité de degrés de liberté permettant une succession de mouvements contrôlés uniquement par des capteurs de fin de course réglables manuellement à la course désirée (asservissement en position difficile dû à la compressibilité de l'air).

### **I.8.3. Le système de perception**

La perception permet de gérer les relations entre le robot et son environnement. Les organes de perception sont des capteurs dits proprioceptifs lorsqu'ils mesurent l'état interne du robot (positions et vitesses des articulations) et extéroceptifs lorsqu'ils recueillent des informations sur l'environnement (détection de présence, de contact, mesure de distance, vision artificielle)

### **I.8.4. La partie commande**

Elle synthétise les consignes des asservissements pilotant les actionneurs, à partir de la fonction de perception et des ordres de l'utilisateur. S'ajoutent à cela :

- L'interface homme-machine à travers laquelle l'utilisateur programme les tâches que le robot doit exécuter,
- Le poste de travail, ou l'environnement dans lequel évolue le robot.

**En conclusion** : La robotique est une science pluridisciplinaire qui requiert, notamment, des connaissances en mécanique, automatique, électronique, électrotechnique, traitement du signal, communications, informatique.

## **I.9. Classification des robots**

On retiendra pour notre part 3 types de robot :

### **I.9.1. Les manipulateurs :**

- Les trajectoires sont non quelconques dans l'espace,
- Les positions sont discrètes avec 2 ou 3 valeurs par axe,
- La commande est séquentielle.

### **I.9.2. Les télémanipulateurs**

Appareils de manipulation à distance (pelle mécanique, pont roulant), apparus vers 1945 aux USA :

- Les trajectoires peuvent être quelconques dans l'espace,
- Les trajectoires sont définies de manière instantanée par l'opérateur, généralement à partir d'un pupitre de commande (joystick).

### **I.9.3. Les robots :**

- Les trajectoires peuvent être quelconques dans l'espace
- L'exécution est automatique,
- Les informations extéroceptives peuvent modifier le comportement du robot.

Pour cette dernière classe, on peut distinguer :

➤ **Les robots manipulateurs industriels, chargés de manipuler, soient :**

Des pièces : Stockage – déstockage,

Palettisation – dé palettisation,

Chargement – déchargement de machine-outil,

Manipulation d'éprouvettes,

➤ **Assemblage de pièces, ...**

Des outils : Soudure en continu ou par points,

Peinture,

Collage,

Ebavurage,

➤ **Les robots didactiques**, qui sont des versions au format réduit des précédents robots. La technologie est différente, de même que les constructeurs. Ils ont un rôle de formation et d'enseignement, ils peuvent aussi être utilisés pour effectuer des tests de faisabilité d'un poste robotisé.

➤ **Les robots mobiles autonomes** : Les possibilités sont plus vastes, du fait de leur mobilité. Notamment, ils peuvent être utilisés en zone dangereuse (nucléaire, incendie, sécurité civile, déminage), inaccessible (océanographie, spatial). De tels robots font appel à des capteurs et à des logiciels sophistiqués. On peut distinguer 2 types de locomotion : Les robots marcheurs qui imitent la démarche humaine, et les robots mobiles qui ressemblent plus à des véhicules.

### **I.9.4. Les générations de robot**

A l'heure actuelle, on peut distinguer 3 générations de robots :

- Le robot est passif** : Il est capable d'exécuter une tâche qui peut être complexe, mais de manière répétitive, il ne doit pas y avoir de modifications intempestives de l'environnement. L'auto-adaptativité est très faible. De nombreux robots sont encore de cette génération.
- Le robot devient actif** : Il devient capable d'avoir une image de son environnement, et donc de choisir le bon comportement (sachant que les différentes configurations ont été prévues). Le robot peut se calibrer tout seul.
- Le robot devient « intelligent »** : Le robot est capable d'établir des stratégies, ce qui fait appel à des capteurs sophistiqués, et souvent à l'intelligence artificielle

## **I.10. Caractéristiques d'un robot**

Un robot doit être choisi en fonction de l'application qu'on lui réserve. Voici quelques paramètres à prendre, éventuellement, en compte :

- ✓ La charge maximum transportable (de quelques kilos à quelques tonnes), à déterminer dans les conditions les plus défavorables (en élongation maximum).
- ✓ -L'architecture du S.M.A., le choix est guidé par la tâche à réaliser (quelle est la rigidité de la structure ? ).
- ✓ Le volume de travail, défini comme l'ensemble des points atteignables par l'organe terminal. Tous les mouvements ne sont pas possibles en tout point du volume de travail. L'espace de travail (reachable workspace), également appelé espace de travail maximal, est le volume de l'espace que le robot peut atteindre via au moins une orientation. L'espace de travail dextre (dextrous workspace) est le volume de l'espace que le robot peut atteindre avec toutes les orientations possibles de l'effecteur (organe terminal). Cet espace de travail est un sous-ensemble de l'espace de travail maximal.
- ✓ Le positionnement absolu, correspondant à 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.
- ✓ 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 maximale 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 vitesse de déplacement (vitesse maximum en élongation maximum), accélération.
- ✓ La masse du robot.
- ✓ Le coût du robot.
- ✓ La maintenance, ...

## **I.11. Programmation des robots**

Classiquement, deux (2) étapes sont utilisées pour faire en sorte qu'un robot connaisse la tâche à exécuter.

### **I.11.1. L'apprentissage :**

- Enregistrement dans une mémoire de la trajectoire à exécuter, sous contrôle d'un opérateur humain,

- Pantin : Structure mécanique identique à celle du robot, qui est déplacée et qui mémorise les points "pertinents",
- Syntaxeur : Un manche de pilotage (joystick) commande les déplacements de l'organe terminal,
- Boîte à boutons : Un interrupteur par actionneur.

### **I.11.2. La génération de trajectoires**

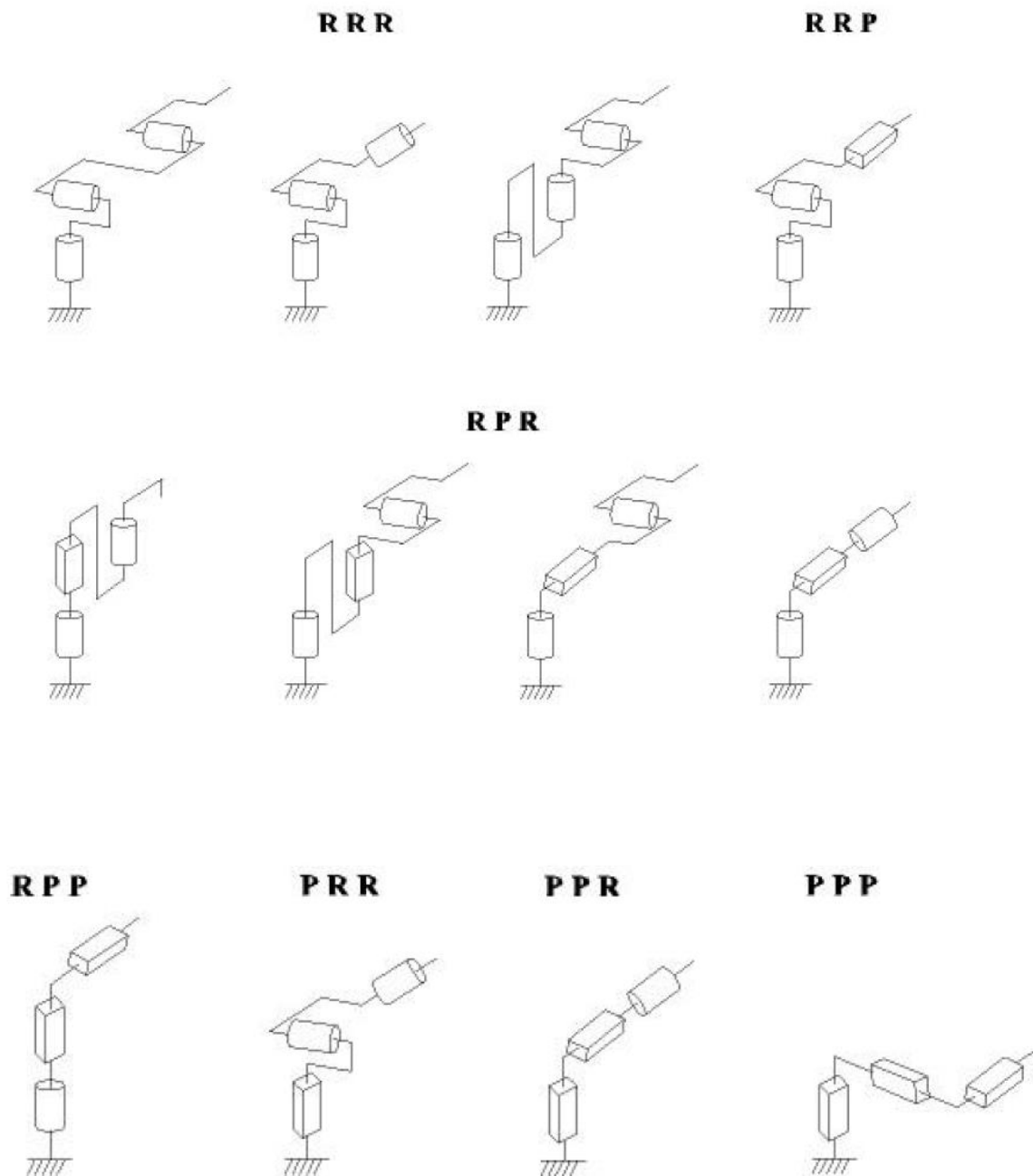
On fait appel à un logiciel qui, à partir du modèle du robot, et des trajectoires à réaliser, élabore la succession des commandes des actionneurs. Les langages de programmation les plus courants sont : WAVE, VAL (Unimate), LM (Hitachi). Nous utiliserons pour notre part les langages associés au robot Stäubli RX 90 (langage V+) et au robot FANUC ARC ou LR.

### **I.12. Morphologie des robots Manipulateurs**

On se limite aux chaînes ouvertes simples. Afin de dénombrer les différentes architectures possibles, on ne considère que 2 paramètres : le type d'articulation (rotoïde (R) ou prismatique (P)) et l'angle que font deux axes articulaires successifs ( $0^\circ$  ou  $90^\circ$ ); sauf cas très particulier, les axes consécutifs d'un robot sont soit parallèles, soit perpendiculaires.

On convient d'appeler les 3 premiers d.d.l, le porteur du robot. Les d.d.l. restants forment le poignet (effecteur), caractérisé par des dimensions beaucoup plus petites et une plus faible masse.

Sont schématisées dans la figure qui suit les 12 morphologies possibles du porteur (ces morphologies sont non redondantes (on élimine a priori les structures limitant les mouvements du porteur à des déplacements linéaires ou planaires : 3 liaisons prismatiques d'axes parallèles, par exemple, ou 3 liaisons rotoïdes d'axes parallèles).



**Figure 1.4**

Dans la pratique, on trouve les 5 structures suivantes :

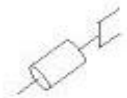
- ✓ Les porteurs anthropomorphes (*RRR*), et plus précisément la première structure de cette classe (*cf.* figure précédente), comme par exemple les robots FANUC (LR, ARC), STÄUBLI RX, ACMA (V80 et SR400), UNIMATION (PUMA), SCEMI (6P-01), AID

(V5), CINCINNATI (T3-7XX), AKR 3000, ASEA (IRB6 et 60), KUKA (IR600), AXEA (V08) ;

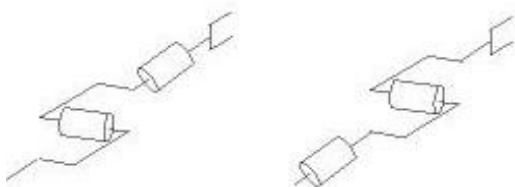
- ✓ Les porteurs sphériques (*RRP*) comme par exemple les robots STANFORD, UNIMATION (1000, 2000, 4000), PSA (BARNABE) ;
- ✓ Les porteurs toriques (*RPR*) et plus précisément la première structure de cette classe, comme par exemple les robots ACMA (H80), les robots de type SCARA (IBM, AXERA, ADEPT, ...) ;
- ✓ Les porteurs cylindriques (*RPP*) comme par exemple les robots ACMA (TH8), MANTEC (A, I et M), CINCINNATI (T3-363) ;
- ✓ Les porteurs cartésiens (*PPP*) comme par exemple les robots ACMA (P80), IBM (7565), SORMEL (CADRATIC), OLIVETTI (SIGMA).

La structure *RRR* dont les 3 axes sont concourants forme ainsi une rotule et s'utilise plus généralement comme un *poignet*. D'autres types de poignets de un à trois axes sont représentés sur la figure suivante.

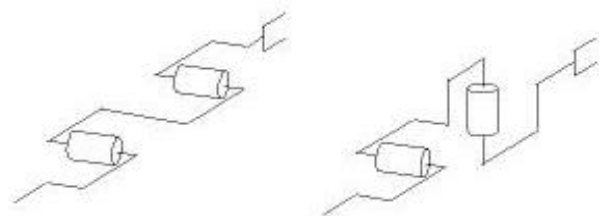
**poignet à 1 axe**



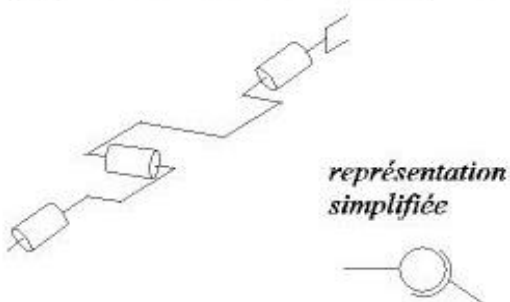
**poignet à 2 axes concourants**



**poignet à 2 axes non concourants**



**poignet à 3 axes concourants (rotule)**



**poignet à 3 axes non concourants**

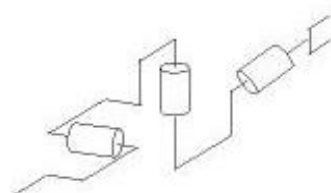


Figure 1.5



Dans la pratique, le *poignet* de type rotule est très répandu. Le robot, obtenu en lui associant un porteur à 3 *d.d.l.*, est la structure la plus classique à 6 *d.d.l.*. Elle permet d'assurer un découplage entre la position et l'orientation de l'organe terminal :

- 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 à l'orientation de l'organe terminal (pince, outil).

Voir la figure ci-contre

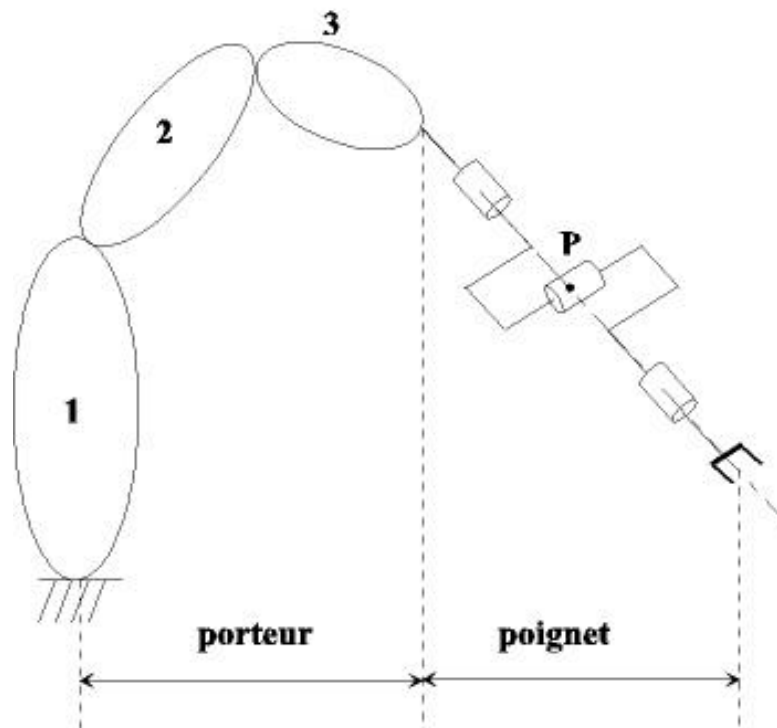


Figure 1.6

## ***CHAPITRE II***

# ***MODELISATION PAR LES PARAMETRES DH STANDARDS ET MODIFIES***

## II. La représentation de Denavit-Hartenberg standard et modifiée

### II.1. La représentation de Denavit-Hartenberg standard

Pour résoudre la cinématique directe d'un robot sériel spatial à  $n$  articulations (rotoïdes ou prismatiques), il est presque inévitable de placer un référentiel sur chaque lien du robot. Ainsi dans le cas d'un robot à  $n$  articulations, donc à  $n+1$  liens, le modèle de la cinématique directe du robot sera exprimé par l'équation suivante :

$$\mathbf{T}_n^0 = \mathbf{T}_1^0 \mathbf{T}_2^1 \mathbf{T}_3^2 \dots \dots \dots \mathbf{T}_n^{n-1} \quad (\text{II.1.1})$$

Où  $\mathbf{T}_n^0$  représente la pose du référentiel  $F_n$  par rapport au référentiel  $F_0$  (celui de la base). Le problème principal avec un choix aléatoire de référentiels est que l'obtention des matrices  $\mathbf{T}_i^{i-1}$  serait assez laborieuse. Un autre problème est bien sûr le fait qu'une telle approche ne serait pas systématique.

Dans les années 1950, **Jacques Denavit** et **Richard Hartenberg** [7] ont eu l'excellente idée de proposer une méthode simple et systématique pour placer des référentiels sur chaque lien d'un mécanisme sériel qui facilite énormément le calcul des matrices  $\mathbf{T}_i^{i-1}$ . Leur idée était de placer les référentiels de telle manière que l'axe  $z_i$  (où  $i = 0, 1, 2, \dots, n - 1$ ) soit placé le long de l'axe de l'articulation  $i + 1$ , dans le cas d'une articulation rotoïde, ou parallèle à la direction de l'articulation  $i + 1$ , dans le cas d'une articulation prismatique, et de contraindre l'axe  $x_i$  d'intersecter l'axe  $z_{i-1}$  à angle droit. Grace à cette double contrainte (non seulement intersecté mais aussi à angle droit), nous avons besoin de seulement quatre paramètres géométriques, plutôt que six, pour décrire la pose du référentiel  $F_i$  par rapport au référentiel  $F_{i-1}$ . Ces quatre paramètres géométriques, deux distances et deux angles, sont en plus très faciles à trouver [8] [9].

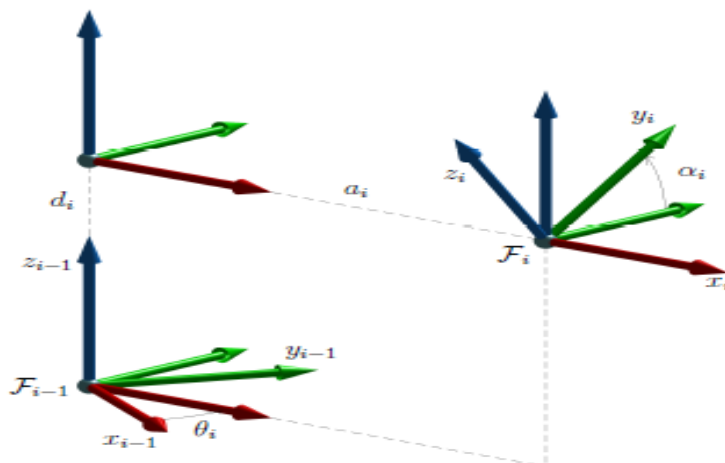


Figure II.1.1 : Les quatre transformations consécutives de base pour obtenir  $F_i$  à partir de  $F_{i-1}$ .

La figure **II.1.1** montre deux référentiels arbitraires,  $F_i$  et  $F_{i-1}$ , qui respectent cette contrainte (l'axe  $x_i$  intersecté l'axe  $z_{i-1}$  à angle droit). Pour l'instant, nous allons ignorer les articulations et les liens du robot. **La figure II.1.1** illustre clairement que nous pouvons obtenir le référentiel  $F_i$  à partir du référentiel  $F_{i-1}$ , à l'aide seulement de quatre transformations consécutives simples (translation le long d'un axes  $x$ ,  $y$  ou  $z$ , ou rotation autour d'un axe  $x$ ,  $y$  ou  $z$ ). Ces quatre transformations peuvent être faites dans l'ordre suivant :

1. rotation de  $\theta_i$  autour de l'axe  $z_{i-1}$  ;
2. translation de  $d_i$  le long de l'axe  $z$  du nouveau référentiel ;
3. translation de  $a_i$  le long de l'axe  $x$  du nouveau référentiel ;
4. rotation de  $\alpha_i$  autour de l'axe  $x$  du nouveau référentiel.

Bien sûr, il est possible de trouver d'autres séquences de quatre transformations consécutives, mais nous nous intéressons uniquement au résultat final.

Nous allons appeler  $\theta_i$ ,  $d_i$ ,  $a_i$  et  $\alpha_i$ , les quatre paramètres **DH**. Ainsi, la matrice  $T_i^{i-1}$  qui représente la pose du référentiel  $F_i$  par rapport au référentiel  $F_{i-1}$  peut être obtenue en fonction des quatre paramètres **DH** par l'équation suivante :

$$T_i^{i-1} = T_{rot,z}(\theta_i)T_{trans}(\mathbf{0}, \mathbf{0}, d_i)T_{trans}(a_i, \mathbf{0}, \mathbf{0})T_{rot,x}(\alpha_i)$$

$$= \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ \mathbf{0} & \sin\alpha_i & \cos\alpha_i & d_i \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \quad (\text{II.1.2})$$

Maintenant que nous avons vu l'idée derrière la méthode **Denavit-Hartenberg**, nous pouvons présenter l'algorithme pour placer ce que nous allons appeler les référentiels **DH** sur un robot sériel et ensuite présenter la méthode pour le calcul des quatre paramètres **DH**.

### II.1.1. Algorithme pour placer les référentiels DH

Pour rendre l'algorithme plus facile à expliquer, nous allons donner les étapes à suivre selon trois situations différentes.

- Pour le référentiel  $F_0$ , placer son axe  $z_0$  le long de l'axe de l'articulation 1, dans le cas d'une articulation rotoïde, en respectant la règle de la main droite, ou parallèle à la direction de l'articulation  $i+1$  et dans le même sens, dans le cas d'une articulation prismatique. Placez l'origine du référentiel  $F_0$  et les axes  $x_0$  et  $y_0$  selon vos préférences. Il est important de comprendre que le référentiel  $F_0$  est fixé à la base du robot. Donc, lorsque les articulations du robot se déplacent, ce référentiel reste immobile par rapport à la base du robot.

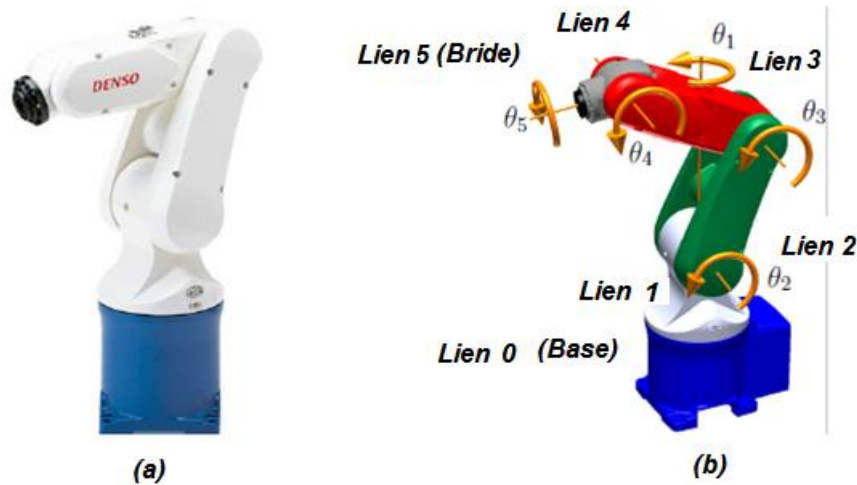
- Pour le référentiel  $F_i$ , où  $i = 1; 2, \dots, n-1$ , placer son axe  $z_i$  le long de l'axe de l'articulation  $i+1$ , dans le cas d'une articulation rotoïde, en respectant la règle de la main droite, ou parallèle à la direction de l'articulation  $i+1$  et dans le même sens, dans le cas d'une articulation prismatique. Choisissez l'origine du référentiel  $F_i$ , de telle manière que l'axe  $x_i$  intersecte l'axe  $z_{i-1}$  à angle droit. Si les axes  $z_i$  et  $z_{i-1}$  sont parallèles ou confondus, il existe une infinité de possibilités. Sinon, il existe une seule possibilité pour l'origine  $F_i$  et deux pour l'axe  $x_i$ . Choisissez ces possibilités selon vos préférences. Il est très important de comprendre que le référentiel  $F_i$  est fixé au lien  $i$ . Donc, lorsque seule l'articulation  $i + 1$  se déplace, le référentiel  $F_i$  reste immobile par rapport à la base du robot.
- Pour le référentiel  $F_n$ , placer l'axe  $x_n$  de telle manière qu'il intersecte l'axe  $z_{n-1}$  à angle droit. Placer l'origine du référentiel  $F_n$  et les axes  $z_n$  et  $y_n$  selon vos préférences. Normalement, on positionne l'origine au centre de l'extrémité de la bride du robot.

### II.1.2. Algorithme pour obtenir les paramètres DH

Une fois tous les  $n+1$  référentiels DH placés, il faut trouver les paramètres DH pour chaque référentiel  $F_i$  (où  $i = 1, 2, \dots, n$ ). En se référant à **la figure III.1.1**, il est facile de déduire les quatre règles suivantes pour les paramètres **DH** :

- $\theta_i$  est l'angle entre l'axe  $x_{i-1}$  et l'axe  $x_i$ , mesuré autour de l'axe  $z_{i-1}$  en utilisant la règle de la main droite. Il est important de comprendre que si l'articulation  $i$  est rotoïde,  $\theta_i$  est une variable articulaire, alors que si l'articulation  $i$  est prismatique,  $\theta_i$  est une constante (positive, zéro ou négative).
- $d_i$  est la distance dirigée de l'axe  $x_{i-1}$  vers l'axe  $x_i$ , mesurée le long de l'axe  $z_{i-1}$ . Il est important de comprendre que si l'articulation  $i$  est prismatique,  $d_i$  est une variable articulaire, alors que si l'articulation  $i$  est rotoïde,  $d_i$  est une constante (positive, zéro ou négative).
- $a_i$  est la distance dirigée de l'axe  $z_{i-1}$  vers l'axe  $z_i$ , mesurée le long de l'axe  $x_i$ . Il est important de se rappeler que  $a_i$  est toujours une constante (positive, zéro ou négative). Par exemple, dans **la figure II.1.1**,  $a_i$  est positive.
- $\alpha_i$  est l'angle entre l'axe  $z_{i-1}$  et l'axe  $z_i$ , mesuré autour de l'axe  $x_i$  en utilisant la règle de la main droite. Il est important de se rappeler que  $\alpha_i$  est toujours une constante. Par exemple, dans **la figure II.1.1**,  $\alpha_i$  est d'environ  $+30^\circ$ .

Lorsqu'on calcule les paramètres **DH** à partir d'une figure avec les référentiels **DH**, il faut se rappeler que le robot est articulé et non une structure. En d'autres termes, il faut toujours s'imaginer que le robot bouge avec ses référentiels **DH**.



**Figure II.1.2** : (a) Le robot VP-5243 de la compagnie DENSO Robotics et (b) un schéma illustrant les sens positifs de rotation.

### II.1.3. Equation de la cinématique directe

Une fois les paramètres **DH** obtenus, il ne reste que de les substituer dans l'équation (II.1.2) pour obtenir les matrices  $T$ ,  $T_2^1$ , ...,  $T_n^{n-1}$ , et puis les substituer dans l'équation (II.1.1). Il est important de comprendre que même si les référentiels  $F_0$  et  $F_n$  sont déjà choisis (par exemple, par le fabricant du robot), il existe tout de même une infinité de possibilités pour les autres référentiels **DH** et, par conséquence, pour les matrices  $T_i^{i-1}$ . Cependant, le résultat final (la matrice  $T_n^0$ ) devrait être le même.

Enfin, si on a un référentiel outil, et un référentiel atelier, la pose du premier par rapport au dernier est définie par l'équation suivante :

$$T_{outil}^{atelier} = T_0^{atelier} T_n^0 T_{outil}^n \quad (II.1.3)$$

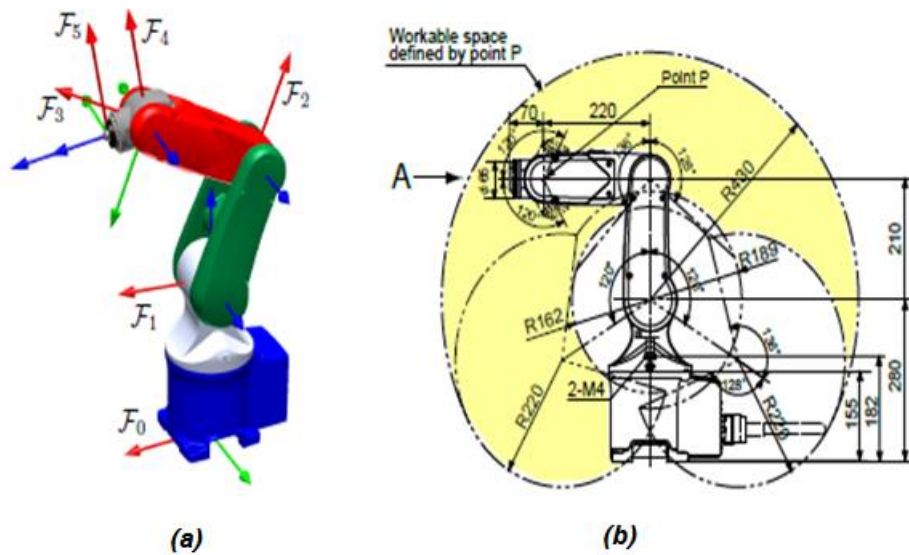
### II.1.4. Exemple d'un robot sériel à cinq articulations rotoïdes

Pour mieux comprendre la méthode **Denavit-Hartenberg**, nous allons l'appliquer dans le cas d'un robot réel. Il s'agit d'un robot sériel à cinq articulations rotoïdes, et plus précisément l'ancien modèle VP-5243 de la compagnie DENSO Robotics. Quelques autres fabricants de robots offrent également des modèles de robots à cinq articulations rotoïdes, destinés au transfert de matériel.

La **figure II.1.2** présente une photo du robot VP-5243, ainsi qu'un schéma qui illustre chacun des cinq axes des articulations rotoïdes avec un sens positif de rotation. La **figure II.1.2(b)** montre également la façon selon laquelle on numérote les liens et les articulations. La base est toujours le lien **0** et la bride du robot, le lien **n**, alors que l'axe 1 est toujours celui de l'articulation à la base.

La procédure pour placer les référentiels **DH** sera illustrée sur l'exemple qui suit (robot VP-5243). Nous allons donner une des nombreuses solutions possibles relatives à la **figure II.1.3(a)**. Il faut bien comprendre que le référentiel  $F_i$  est fixé au lien  $i$ , malgré le fait que son axe  $z_i$ , dans le cas  $i < n$ , est le long de l'axe de l'articulation  $i + 1$ .

Une fois les référentiels **DH** placés, l'étape suivante est de trouver les paramètres **DH** en se référant aux **figures II.1.3(a) et (b)**. Encore une fois, il est nécessaire de s'imaginer que le robot avec ses référentiels **DH** sont en mouvement, mais ce ne serait évidemment pas le cas à l'examen final ou en pratique. Une erreur fréquente lors du calcul des paramètres  $d_i$  et  $a_i$  est de considérer les axes d'un référentiel comme des segments et non comme des lignes d'une longueur infinie.



**Figure II.1.3:** (a) Une des solutions pour les référentiels DH et (b) les dimensions du robot.

$i$	$\alpha_i$	$a_i$	$\theta_i$	$d_i$
1	$-90^\circ$	0 mm	$\theta_1$	280 mm
2	$0^\circ$	210 mm	$\theta_2$	0 mm
3	$0^\circ$	220 mm	$\theta_3$	0 mm
4	$-90^\circ$	0 mm	$\theta_4$	0 mm
5	$0^\circ$	0 mm	$\theta_5$	70 mm

**Tableau II.1.1** Paramètres **DH** selon le choix de référentiels **DH** présente à la **figure II.1.3(a)**.

Les paramètres DH sont donnés au **tableau II.1.1**. La partie la plus difficile à comprendre est la colonne des  $\theta_i$ . Si nous continuons avec les paramètres DH trouvés dans le **tableau II.1.1**, la

configuration du robot où toutes les articulations sont à zéro degrés semble un peu étrange. Dans cette configuration, tous les axes  $x_i$  sont parallèles et dans le même sens. Dans un robot existant, le choix de configuration zéro a été déjà fait par son fabricant et nous devons nous conformer à ce choix. Dans sa configuration zéro, le robot VP-5243 réel est étiré vers le haut. En d'autres termes, dans la configuration illustrée à la **figure II.1.3 (b)**,  $\theta_1 = 0^\circ, \theta_2 = 0^\circ, \theta_3 = 90^\circ, \theta_4 = 0^\circ$ , et  $\theta_5 = 0^\circ$ . Or, selon le choix de référentiels DH, dans cette configuration,  $\theta_1 = 0^\circ, \theta_2 = -90^\circ, \theta_3 = 90^\circ, \theta_4 = -90^\circ$ , et  $\theta_5 = 0^\circ$ . Ainsi, pour être conforme avec le vrai robot VP-5243, il faut ajouter un décalage de  $-90^\circ$  sur  $\theta_2$  et un autre de  $-90^\circ$  aussi sur  $\theta_4$ . Les nouveaux paramètres DH sont montrés au **tableau II.1.2**.

$i$	$\alpha_i$	$a_i$	$\theta_i$	$d_i$
1	$-90^\circ$	0 mm	$\theta_1$	280 mm
2	$0^\circ$	210 mm	$\theta_2 - 90^\circ$	0 mm
3	$0^\circ$	220 mm	$\theta_3$	0 mm
4	$-90^\circ$	0 mm	$\theta_4 - 90^\circ$	0 mm
5	$0^\circ$	0 mm	$\theta_5$	70 mm

**Tableau II.1.2** : Paramètres DH pour être conforme avec le vrai robot VP-5243.

L'étape suivante est de substituer les paramètres DH du **tableau II.1.2** dans l'équation (II.1.2)

$$\begin{aligned}
 T_1^0 &= T_{rot,z}(\theta_1)T_{trans}(0, 0, 280)T_{trans}(0, 0, 0)T_{rot,x}(-90^\circ) \\
 &= \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_1 & 0 & \cos\theta_1 & 0 \\ 0 & -1 & 0 & 280 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.1.4}
 \end{aligned}$$

$$\begin{aligned}
 T_2^1 &= T_{rot,z}(\theta_2 - 90^\circ)T_{trans}(0, 0, 0)T_{trans}(210, 0, 0)T_{rot,x}(0^\circ) \\
 &= \begin{bmatrix} \sin\theta_2 & \cos\theta_2 & 0 & 210\sin\theta_2 \\ -\cos\theta_2 & \sin\theta_2 & 0 & -210\cos\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.1.5}
 \end{aligned}$$

$$\begin{aligned}
 T_3^2 &= T_{rot,z}(\theta_3)T_{trans}(0, 0, 0)T_{trans}(220, 0, 0)T_{rot,x}(0^\circ) \\
 &= \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & 220\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & 220\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.1.6}
 \end{aligned}$$



$$\begin{aligned} \mathbf{T}_4^3 &= T_{rot,z}(\theta_4 - 90^\circ)T_{trans}(0, 0, 0)T_{trans}(0, 0, 0)T_{rot,x}(-90^\circ) \\ &= \begin{bmatrix} \sin\theta_4 & 0 & \cos\theta_4 & 0 \\ -\cos\theta_4 & 0 & \sin\theta_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (\text{II.1.7})$$

$$\begin{aligned} \mathbf{T}_5^4 &= T_{rot,z}(\theta_5)T_{trans}(0, 0, 70)T_{trans}(0, 0, 0)T_{rot,x}(0^\circ) \\ &= \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 70 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (\text{II.1.8})$$

Enfin, il ne reste que de substituer les matrices ci-dessus dans l'équation (II.1) et faire quelques manipulations trigonométriques afin de rendre l'expression finale plus compacte :

$$\begin{aligned} \mathbf{T}_5^0 &= \mathbf{T}_1^0 \mathbf{T}_2^1 \mathbf{T}_3^2 \mathbf{T}_5^4 \\ &= \begin{bmatrix} -C_1 C_{234} C_5 + S_1 S_5 & C_1 C_{234} S_5 + S_1 C_5 & C_1 S_{234} & C_1 (70 S_{234} + 220 S_{23} + 210 S_2) \\ -S_1 C_{234} C_5 - C_1 S_5 & S_1 C_{234} S_5 - C_1 C_5 & S_1 S_{234} & S_1 (70 S_{234} + 220 S_{23} + 210 S_2) \\ S_{234} C_5 & -S_{234} S_5 & C_{234} & 280 + 70 C_{234} + 220 C_{23} + 210 C_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (\text{II.1.9})$$

Où :

$$\begin{aligned} S_1 &= \sin\theta_1, C_1 = \cos\theta_1 \\ S_2 &= \sin\theta_2, C_2 = \cos\theta_2 \\ S_5 &= \sin\theta_5, C_5 = \cos\theta_5 \\ S_{23} &= \sin(\theta_2 + \theta_3), C_{23} = \cos(\theta_2 + \theta_3) \\ S_{234} &= \sin(\theta_2 + \theta_3 + \theta_4), C_{234} = \cos(\theta_2 + \theta_3 + \theta_4) \end{aligned} \quad (\text{II.1.10})$$

Pour faire ce type de simplification, ou plutôt substitution, il faut utiliser une fonction qui fait des regroupements trigonométriques uniquement sur les matrices qui correspondent à des axes qui restent toujours parallèles (dans notre cas, les axes 2, 3 et 4).

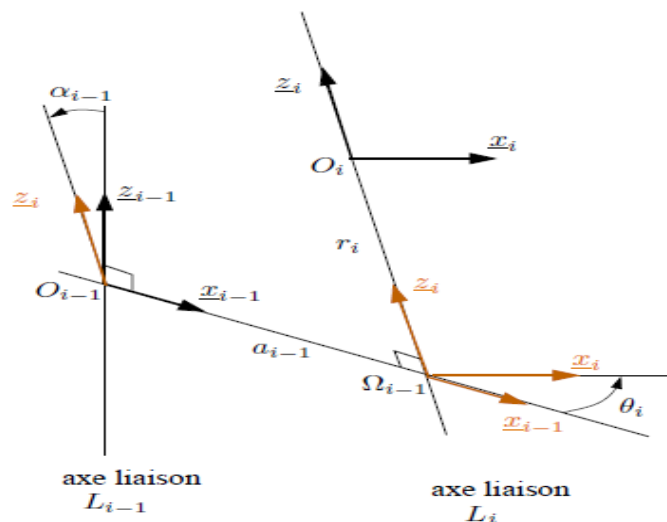
## II.2. La représentation de Denavit-Hartenberg modifiée

La technique la plus répandue pour décrire la géométrie d'un bras manipulateur consiste à utiliser les paramètres de *Denavit-Hartenberg modifiés*[10]. On associe au  $i^{\text{ème}}$  corps le repère  $R_i = (O_i, \underline{x}_i, \underline{y}_i, \underline{z}_i)$ , avec  $i = 0, 1, \dots, n$ . Pour placer les repères à  $R_{n-1}$  selon la règle des paramètres de *Denavit-Hartenberg modifiés*, il convient de suivre les règles suivantes (voir **figure II.2.1**), pour  $i = 2, 3, \dots, n$  :

- $O_{i-1}$  est le pied de la perpendiculaire commune aux axes des liaisons  $L_{i-1}$  et  $L_i$ , situé sur  $L_{i-1}$ . Si les axes des liaisons sont parallèles, il faut choisir arbitrairement une perpendiculaire commune ;
- $\underline{x}_{i-1}$  est le vecteur unitaire de cette perpendiculaire commune, orienté de  $L_{i-1}$  vers  $L_i$ . Si les axes des deux liaisons sont concourants ou confondus, l'orientation est arbitraire (règle avant, droite, haut) ;
- $\underline{z}_{i-1}$  est un vecteur unitaire de l'axe de la liaison  $L_{i-1}$ , librement orienté. On privilégie généralement les débattements positifs pour les liaisons prismatiques et on symétrise généralement par rapport à  $0$  le débattement des liaisons rotoïdes ;
- $\underline{y}_{i-1}$  est tel que le repère  $R_{i-1}$  soit orthonormé direct.

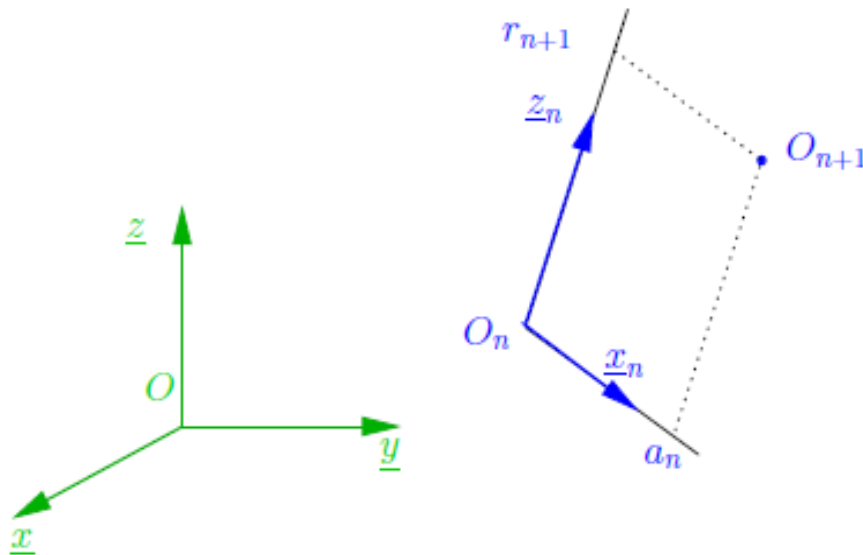
Ensuite, le repère  $R_0$  est défini librement en suivant des considérations de simplicité.

De même, le point  $O_n$  est choisi de façon à annuler un maximum de paramètres. Un point  $O_{n+1}$  est associé à l'organe terminal du bras manipulateur, selon des critères opérationnels : typiquement, si le bras manipulateur est équipé d'une pince, le point  $O_{n+1}$  représente Le centre de ses mors.



**Figure II.2.1** – Paramètres de *Denavit-Hartenberg modifiés*

On choisit alors enfin  $R_n$  de sorte que le point  $O_{n+1}$  se situe dans le plan  $R_i = (O_i, \underline{x}_i, \underline{y}_i, \underline{z}_i)$ , conformément à la **figure II.2.2**.

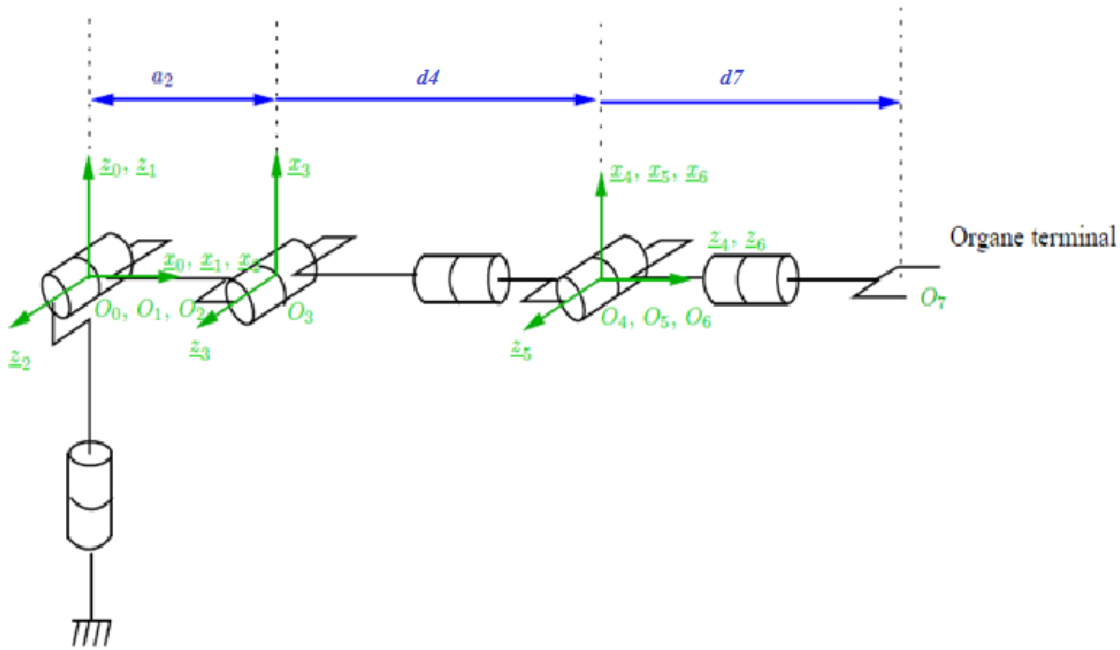


**Figure II.2.2** – Placements respectifs des repères  $R_n$  et du point  $O_{n+1}$

Les repères étant choisis, la situation du repère d'un corps respectivement à celle du repère du corps suivant peut être définie conformément à la **figure II.2.1**. Les paramètres  $\alpha_{i-1}, a_{i-1}$  et  $\theta_i$ , avec  $i = 1, 2, \dots, n$ , placés sur cette figure sont appelés paramètres de *Denavit-Hartenberg modifiés*. Textuellement, on peut les définir comme suit :

- $\alpha_{i-1}$  : Angle algébrique entre  $\underline{z}_{i-1}$  et  $\underline{z}_i$ , mesuré autour de  $\underline{x}_{i-1}$ ;
- $a_{i-1}$  : Distance arithmétique de la perpendiculaire commune aux axes des liaisons  $L_{i-1}$  et  $L_i$  mesurée le long de  $\underline{x}_{i-1}$  ;
- $\theta_i$  : Angle algébrique entre  $\underline{x}_{i-1}$  et  $\underline{x}_i$ , mesuré autour de  $\underline{z}_i$  ;
- $r_i$  : Distance algébrique du point  $O_i$  à la perpendiculaire, mesurée le long de  $\underline{z}_i$ .

Les paramètres de *Denavit-Hartenberg modifiés* permettent de déterminer de manière unique la configuration des corps et donc la géométrie du bras manipulateur. Lorsque l'on modélise un bras manipulateur, il est souhaitable de représenter celui-ci dans une configuration particulière, pour visualiser plus facilement les paramètres et annuler certains paramètres constants. On obtient alors les paramètres constants  $\alpha_{i-1}, a_{i-1}, \theta_i$  et  $d_i$  la configuration des liaisons décrite par les paramètres  $\theta_i$  et  $d_i$  variables. A la **figure II.2.3**, on a explicité un exemple de paramétrage dans le cas d'un bras manipulateur de type  $6R$ .



Articulation	$\alpha_{i-1}$	$a_{i-1}$	$\theta_i$	$d_i$	$\sigma_i$
1	0	0	$q_1$	0	0
2	$\frac{\pi}{2}$	0	$q_2$	0	0
3	0	$a_2$	$q_3$	0	0
4	$\frac{\pi}{2}$	0	$q_4$	$d_4$	0
5	$-\frac{\pi}{2}$	0	$q_5$	0	0
6	$\frac{\pi}{2}$	0	$q_6$	0	0

**Figure II.2.3** – Repérage et paramétrage d'un bras manipulateur de type 6R selon la même méthode des paramètres de **Denavit-Hartenberg** modifiés

### II.2.1. Relations géométriques

Conformément au paramétrage de Denavit-Hartenberg modifié, la matrice de rotation entre les corps  $C_{i-1}$  et  $C_i$  est donc :

$$T_{i-1,i} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_{i-1} & -\sin\alpha_{i-1} & 0 \\ 0 & \sin\alpha_{i-1} & \cos\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & 1 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.1})$$

➤ Rotation d'un angle  $\alpha_{i-1}$  autour de l'axe  $x_{i-1}$

$$R(x_{i-1}, \alpha_{i-1}) \Rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_{i-1} & -\sin\alpha_{i-1} & 0 \\ 0 & \sin\alpha_{i-1} & \cos\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.2})$$

➤ Translation de  $a_{i-1}$  suivant  $x_{i-1}$

$$\Rightarrow \begin{pmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.3})$$

➤ Rotation d'un angle de  $\theta_i$  autour de l'axe  $z_i$

$$R(z_i, \theta_i) \Rightarrow \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & 1 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.4})$$

➤ Translation de  $d_i$  le long de l'axe  $z_i$

$$\Rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.5})$$

Soit :

$$T_{i-1,i} = \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \cos\alpha_{i-1}\sin\theta_i & \cos\alpha_{i-1}\cos\theta_i & -\sin\alpha_{i-1} & -d_i\sin\alpha_{i-1} \\ \sin\alpha_{i-1}\sin\theta_i & \sin\alpha_{i-1}\cos\theta_i & \cos\alpha_{i-1} & d_i\cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{II.2.6})$$

Qui prend la forme :

$$T_{i-1,i} = \begin{pmatrix} R_{i-1,i} & p_{i-1,i} \\ 0 & 1 \end{pmatrix} \quad (\text{II.2.7})$$

Où  $R_{i-1,i}$  représente la rotation entre les repères  $\mathcal{R}_{i-1}$  et  $\mathcal{R}_i$  et  $p_{i-1,i}$  la translation entre ces mêmes repères.

### **II.2.2. Comparaison entre les deux représentations**

Comparativement aux paramètres DH standards, cette représentation est parfaitement adaptée pour la description des modèles cinématiques et dynamiques. Par contre, elle ne permet pas d'obtenir la position et l'orientation du repère lié à l'effecteur dans le cas où la dernière articulation est du type rotoïde. Ce problème n'existe pas dans la représentation DH standard. Une transformation finale correspondant le plus souvent à une translation d'axe est donc nécessaire (dans notre exemple translation suivant l'axe  $z_6$  de  $d_7$ )

## ***CHAPITRE III***

# ***DYNAMIQUE D'UN SYSTEME ROBOTIQUE***

### III. Dynamique d'un système robotique

Deux approches sont utilisées : celles qui utilisent des considérations énergétiques (méthodes variationnelles de Lagrange) et celles qui utilisent les équations de Newton (solides en translation) et d'Euler (solides en rotation). Les premières débouchent sur l'équation dynamique sous forme implicite, alors que les dernières donnent lieu à des équations sous forme explicite ou ouverte.

#### III.1. Approche par les équations de Lagrange

Définissons les notations suivantes et les grandeurs qu'elles désignent :

$G_j$  : Centre de gravité du corps  $c_j$  ;

$S_j$  : Vecteur ayant pour origine  $O_j$  et pour extrémité le centre de masse du corps  $c_j$ . Il est égal à  $O_j G_j$  ;

$L_j$  : Vecteur liant l'origine du repère  $R_j$ , antécédent du repère  $R_j$ , et l'origine du repère  $R_j$ . Il est égal à  $O_i O_j$  ;

$V_{G_j}$  : Vitesse du centre de gravité du corps  $c_j$  ;

$\omega_j$  : Vitesse de rotation du corps  $c_j$  ;

$I_{G_j}$  : Tenseur d'inertie du corps  $c_j$  par rapport au repère parallèle à  $R_j$  et d'origine  $c_j$  ;

$I_{a_j}$  : Moment d'inertie de l'actionneur  $j$  et de son réducteur pressenti par l'articulation  $j$  ;

${}^j J_j$  : Tenseur d'inertie du corps  $c_j$  par rapport au repère  $R_j$  qui exprime par :

$${}^j J_j = \begin{bmatrix} \int (y^2 + z^2) dm & - \int xy dm & - \int xz dm \\ - \int xy dm & \int (x^2 + z^2) dm & - \int yz dm \\ - \int xz dm & - \int yz dm & \int (x^2 + y^2) dm \end{bmatrix} = \begin{bmatrix} I_{xx_j} & I_{xy_j} & I_{xz_j} \\ I_{yx_j} & I_{yy_j} & I_{yz_j} \\ I_{zx_j} & I_{zy_j} & I_{zz_j} \end{bmatrix} \quad \text{(III.1.1)}$$

#### III.1.1. Introduction

Cette méthode n'est pas la plus performante du point de vue du nombre d'opération, mais c'est la méthode la plus simple compte tenu de ces objectifs. Considérons un robot idéal sans frottement, sans élasticité et ne subissant ou n'exerçant aucun effort extérieur.

Le formalisme de Lagrange décrit les équations du mouvement en termes de travail et d'énergie du système.

$$\tau_i = \frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{q}_i} \right] - \frac{\partial L}{\partial q_i} \quad i = 1, \dots, n \quad \text{(III.1.2)}$$

Avec:

- $L$  : Lagrangien du système égal à  $E - U$  ;
- $E$  : Energie cinétique totale du système



- $U$  : Energie potentielle totale du système ;

### III.1.2. Forme générale des équations dynamiques

$$E = \frac{1}{2} \dot{q}^T M \dot{q} \quad (\text{III.1.3})$$

Où  $M$  est la matrice ( $n \times n$ ) de l'énergie cinétique, d'élément générique  $m_{ij}$ , appelée aussi matrice d'inertie du robot, qui est symétrique et définie positive. Ses éléments sont fonctions des variables articulaires  $q$ .

L'énergie potentielle étant fonction des variables articulaires  $q$ , le couple  $\tau$  peut se mettre, à partir des

Équations [III.1.2] et [III.1.3], sous la forme :

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \quad (\text{III.1.4})$$

$C(q, \dot{q})\dot{q}$  : Vecteur de dimension ( $n \times 1$ ) représentant les couples/forces de Coriolis et des forces Centrifuges, tel que :

$$C\dot{q} = \dot{M}\dot{q} - \frac{\partial E}{\partial q} \quad (\text{III.1.5})$$

$G = [G_1 \dots \dots \dots G_n]^T$ : Vecteur des couples/forces de gravité.

$$\begin{cases} C_{ij} = \sum_{k=1}^n c_{i,jk} \dot{q}_k \\ c_{i,jk} = \frac{1}{2} \left[ \frac{\partial m_{ij}}{\partial q_k} + \frac{\partial m_{ik}}{\partial q_j} - \frac{\partial m_{jk}}{\partial q_i} \right] \end{cases} \quad (\text{III.1.6})$$

$c_{i,jk}$  étant les symboles de Christoffel qui seront définis plus loin (III.1.16)

Les éléments du vecteur  $G$  se calculent en écrivant que :

$$G_i = \frac{\partial U}{\partial q_i} \quad (\text{III.1.7})$$

Les éléments de  $M$ ,  $C$  et  $G$  sont fonctions des paramètres géométriques et inertiels du mécanisme. Les équations dynamiques d'un système mécanique articulé forment donc un système de  $n$  équations différentielles du second ordre, couplées et non linéaires.

### III.1.3. Calcul des éléments de $M$ , $V$ et $G$

Pour calculer les éléments de  $M$ ,  $V$  et  $Q$ , il faut d'abord calculer les énergies cinétiques et potentielles de tous les corps du robot. On procède ensuite comme suit :

- L'élément  $\mathbf{m}_{ii}$  est égal au coefficient de  $(\dot{\mathbf{q}}_i^2/2)$  dans l'expression de l'énergie cinétique, tandis que l'élément  $\mathbf{m}_{ij}$ , si  $i \neq j$ , est égal au coefficient de  $\dot{\mathbf{q}}_i \dot{\mathbf{q}}_j$  ;
- Le calcul de  $\mathbf{C}$  se fait selon la relation (III.1.5-6)
- le calcul de  $\mathbf{Q}$  se fait selon la relation (III.1.7)

L'énergie cinétique totale du système est donnée par la relation :

$$\mathbf{E} = \sum_{j=1}^n \mathbf{E}_j \quad (\text{III.1.8})$$

Où  $\mathbf{E}_j$  désigne l'énergie cinétique du corps  $\mathbf{C}_j$ , qui s'exprime par :

$$\mathbf{E}_j = \frac{1}{2} (\boldsymbol{\omega}_j^T \mathbf{I}_{G_j} \boldsymbol{\omega}_j + m_j \mathbf{V}_{G_j}^T \mathbf{V}_{G_j}) \quad (\text{III.1.9})$$

D'autre part l'énergie potentielle s'écrit :

$$\mathbf{U} = \sum_{j=1}^n \mathbf{U}_j = - \sum_{j=1}^n m_j \mathbf{g}^T (\mathbf{L}_{0,j} + \mathbf{S}_j) \quad (\text{III.1.10})$$

Les énergies cinématique et potentielles étant linéaires par rapport aux paramètres inertiels, le modèle dynamique l'est également.

Les jacobiens vitesses linéaires et angulaires permettent d'établir une relation entre les grandeurs vitesses dans l'espace opérationnel et articulaire

$$\boldsymbol{\omega}_j = \mathbf{J}_{\omega_j} \dot{\mathbf{q}}, \mathbf{V}_{G_j} = \mathbf{J}_{v_j} \dot{\mathbf{q}} \quad (\text{III.1.11})$$

$$\mathbf{E} = \sum_{j=1}^n \mathbf{E}_j = \frac{1}{2} \dot{\mathbf{q}}^T \sum_{j=1}^n (\mathbf{J}_{\omega_j}^T \mathbf{I}_{G_j} \mathbf{J}_{\omega_j} + m_j \mathbf{J}_{v_i}^T \mathbf{J}_{v_i}) \dot{\mathbf{q}} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M} \dot{\mathbf{q}} \quad (\text{III.1.12})$$

$$\mathbf{M} = \sum_{j=1}^n (\mathbf{J}_{\omega_j}^T \mathbf{I}_{G_j} \mathbf{J}_{\omega_j} + m_j \mathbf{J}_{v_i}^T \mathbf{J}_{v_i}) \quad (\text{III.1.13})$$

$\mathbf{J}_{v_i}$  est le jacobien relatif au centre de masse du corps  $c_i$ . De même  $\mathbf{J}_{\omega_j}$  est le jacobien vitesse angulaire relatif au centre de gravité du corps  $c_i$ .

$$\mathbf{C}(\mathbf{q}) = \begin{bmatrix} \mathbf{b}_{1,11} & \mathbf{b}_{1,22} & - & - & - & \mathbf{b}_{1,nn} \\ \mathbf{b}_{2,11} & \mathbf{b}_{2,22} & - & - & - & \mathbf{b}_{2,nn} \\ \mathbf{b}_{3,11} & \mathbf{b}_{3,22} & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ \mathbf{b}_{n,11} & \mathbf{b}_{n,22} & \mathbf{b}_{n,33} & - & - & \mathbf{b}_{n,nn} \end{bmatrix} \quad (\text{III.1.14})$$

$$\mathbf{B}(\mathbf{q}) = \begin{bmatrix} 2\mathbf{b}_{1,12} & 2\mathbf{b}_{1,13} & 2\mathbf{b}_{1,14} & - & - & 2\mathbf{b}_{1,(n-1)n/2} \\ 2\mathbf{b}_{2,12} & 2\mathbf{b}_{2,13} & 2\mathbf{b}_{2,14} & - & - & 2\mathbf{b}_{2,(n-1)n/2} \\ 2\mathbf{b}_{3,12} & 2\mathbf{b}_{3,13} & 2\mathbf{b}_{2,14} & - & - & 2\mathbf{b}_{3,(n-1)n/2} \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ 2\mathbf{b}_{n,12} & 2\mathbf{b}_{n,13} & 2\mathbf{b}_{n,14} & - & - & 2\mathbf{b}_{n,(n-1)n/2} \end{bmatrix} \quad (\text{III.1.15})$$

$b_{i,jk}$  : Symboles de Christoffel définis comme suit :

$$b_{i,jk} = \frac{1}{2} (m_{ijk} + m_{ikj} - m_{jki}) \quad (\text{III.1.16})$$

Avec :  $m_{ijk} = \frac{\partial m_{ij}}{\partial q_k}$  (III.1.17)

$$G(q) = -\sum_{i=1}^n J_{v_i}^T (m_i g) = -\left( J_{v_1}^T (m_1 g) + J_{v_2}^T (m_2 g) \dots \dots \dots + J_{v_n}^T (m_n g) \right) \quad (\text{III.1.18})$$

Ces relations III.13, III.14, III.15 et III.18 déterminent la forme fermée ou explicite des équations de la dynamique du robot.

### III.1.4. Prise en compte des frottements

Les frottements non compensés provoquent en effet des erreurs statiques, des retards et des cycles limites. Différents modèles de frottements ont été proposés, le modèle de frottements du type frottement secs (ou de coulomb) fait l'hypothèse d'un couple constant de frottement en opposition au mouvement. Au démarrage (vitesse nulle) un couple supérieur au couple de frottement sec doit être développé pour amorcer le mouvement.

Des tests expérimentaux ont prouvé que le Couple de frottement au démarrage décroît exponentiellement à faible vitesse, puis croit ensuite proportionnellement à la vitesse. Le modèle correspondant est donné par :

$$\Gamma_{fi} = F_{si} \text{signe}(\dot{q}_i) + F_{vi} \dot{q}_i + F_{ei} e^{|\dot{q}_i| B_i} \text{signe}(\dot{q}_i) \quad (\text{III.1.19})$$

$F_{si}$  Et  $F_{vi}$  désignent respectivement les paramètres de frottements sec et visqueux de l'articulation.

Dans bon nombre d'applications, une approximation du couple de frottement ramène l'expression (III.1.19) à une forme simplifiée

$$\Gamma_{fi} = F_{si} \text{signe}(\dot{q}_i) + F_{vi} \dot{q}_i \quad (\text{III.1.20})$$

### III.1.5. Prise en compte des inerties des actionneurs

On représente l'énergie cinétique de l'actionneur j par un terme de la forme  $\frac{1}{2} I_{aj} \dot{q}_j^2$ . Le paramètre inertiel  $I_{aj}$  peut s'écrire :

$$I_{aj} = N_j^2 J_{mj} \quad (\text{III.1.21})$$

Ou :  $J_{mj}$  est le moment d'inertie du rotor de l'actionneur j.  $N_j$  est le rapport de réduction de l'axe j  
Egal à  $\left( \dot{q}_{m_j} / \dot{q}_j \right)$  :  $\dot{q}_{m_j}$  désigne la vitesse du rotor de l'actionneur.

### III.1.6. prise en compte des efforts exercés par l'organe terminal sur l'environnement

L'organe terminal d'un robot exerce un effort statique  $\mathcal{F}_{en}$  sur l'environnement tel que

$$\Gamma_e = J_n^T \mathcal{F}_{en} \quad (\text{III.1.22})$$

### III.1.7. Prise en compte des flexibilités des articulations

Les robots utilisant des transmissions par courroies, des arbres de dimensions importantes ou des réducteurs harmoniques, présentent une certaine élasticité dans les articulations

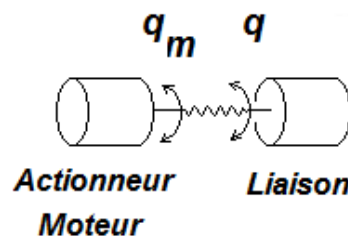


Figure III.1. Modélisation de la flexibilité des articulations .

### III.1.8. Modélisation des flexibilités

L'introduction de la flexibilité des articulations double le nombre de degré de liberté.

Si  $\mathbf{q}_m$  et  $\mathbf{q}$  désignent respectivement les vecteurs variables moteurs et variables articulaires, l'énergie potentielle de l'élasticité s'écrit :

$$U_K = \frac{1}{2} \mathbf{K} (\mathbf{q} - \mathbf{q}_m)^T (\mathbf{q} - \mathbf{q}_m) \quad (\text{III.1.23})$$

Avec :

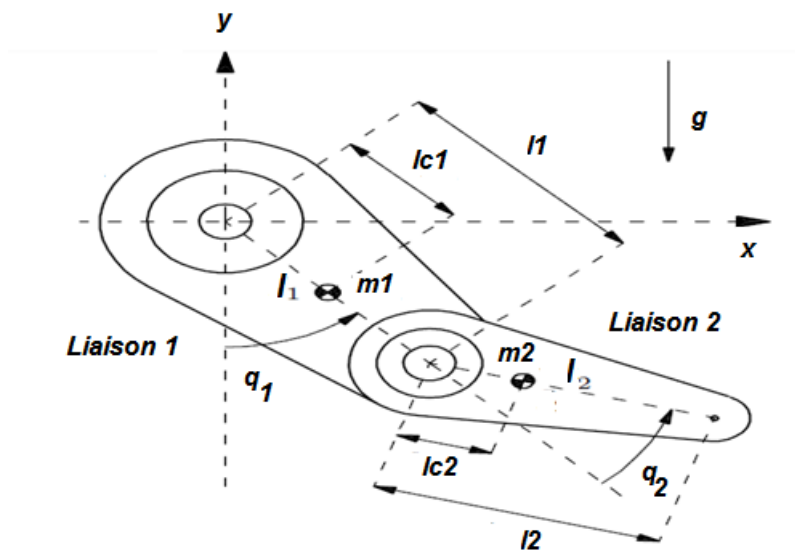
$\mathbf{q}_m$  : Vecteur des variables articulaires moteur exprimées cotés bras

$\mathbf{K}$  : matrice ( $\mathbf{n} \times \mathbf{n}$ ) diagonale définie positive qui traduit la raideur de torsion au niveau de l'articulation.

### III.1.9. Application de la formulation de Lagrange sur le robot PELICAN



**Figure III.2.** PELICAN: bras de robot expérimental à CICESE, laboratoire de robotique.



**Figure III.3.** Schéma du robot prototype 2 degrés de liberté PELICAN

La **Figure III.3**, représente le robot Pélican schématiquement. Comme Il s'agit d'un robot d'un bras planaire avec deux liens reliés par des articulations rotoides, c.-à-d., il possède 2 degrés de liberté. Les articulations sont entraînées par deux moteurs électriques situés sur l'épaule » (base) et au niveau du « coude ». Il s'agit d'un mécanisme d'entraînement direct, c'est-à-dire les axes des moteurs sont connectés directement aux liens sans engrenages ou transmissions (courroies ou ceintures). Le bras manipulateur se compose de deux liens rigides de longueur  $l_1$  et  $l_2$ , de masses  $m_1$  et  $m_2$  respectivement. Le robot se déplace sur le plan  $x - y$  comme illustré dans la **figure**

**III.3.** Les distances entre les axes de rotation et les centres de masse sont notées par  $l_{c1}$  et  $l_{c2}$  pour les liens **1** et **2**, respectivement. Enfin,  $I_1$  et  $I_2$  désignent les moments d'inertie des liens relatifs aux axes qui passent par les centres de masse respectifs lesquels sont parallèles à l'axe  $z$ .

Les degrés de liberté sont associés à l'angle de  $q_1$ , qui est mesuré à partir de la position verticale, et  $q_2$ , qui est mesuré par rapport au prolongement de la première liaison vers la deuxième liaison. Le sens antihoraire étant défini comme sens positif pour les angles. [11]. Le vecteur des coordonnées articulaires  $q$  est défini comme :

$$q = [q_1 q_2]^T \quad (\text{III.1.24})$$

La signification des divers paramètres constants en cause ainsi que leur les valeurs numériques sont résumés dans le **tableau III.1.1**.

DESCRIPTION	NOTATION	VALEURS	UNITE
Langueur de 1er lien	$l_1$	0.26	m
Langueur de 2ème lien	$l_2$	0.26	m
Distance axe au CDM du lien 1	$l_{c1}$	0.0983	m
Distance axe au CDM du lien 2	$l_{c2}$	0.0229	m
Masse de lien 1	$m_1$	6.5225	kg
Masse de lien 2	$m_1$	2.0458	kg
Inertie de CDM du lien 1	$I_1$	0.1213	$kg/m^2$
Inertie de CDM du lien 2	$I_2$	0.0116	$kg/m^2$
	$g$	9.81	$m/s^2$

**Tableau III.1.** Paramètres physiques de bras de robot PELICAN

Afin obtenir les équations de Lagrange, nous commençons par déterminer l'énergie cinétique  $E(q, \dot{q})$ . Pour ce manipulateur, elle peut être décomposée en la somme de deux parties :

- Le produit de moitié la masse multipliée par le carré de la vitesse du centre de masse.
- le produit de la moitié de son moment d'inertie (par rapport au centre de masse) fois le carré de sa vitesse angulaire (orientés vers le centre de masse).

Soit pour les deux liaisons ::

$$E(q, \dot{q}) = E_1(q, \dot{q}) + E_2(q, \dot{q}) \quad (\text{III.1.25})$$

Où  $E_1(q, \dot{q})$  et  $E_2(q, \dot{q})$  sont les énergies cinétiques associées aux masses  $m_1$  et  $m_2$ , respectivement.

Développons maintenant plus en détail, les expressions mathématiques correspondantes. A cet effet, on observe tout d'abord que les coordonnées du centre de masse de lien 1, exprimées sur le plan  $x$ - $y$ , sont :

$$x_1 = l_{c1} \sin(q_1) \quad (\text{III.1.26})$$

$$y_1 = -l_{c1} \cos(q_1) \quad (\text{III.1.27})$$

Le vecteur de vitesse  $v_1$  du centre de masse de cette liaison est ensuite,

$$v_1 = \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} l_{c1} \cos(q_1) \dot{q}_1 \\ l_{c1} \sin(q_1) \dot{q}_1 \end{bmatrix} \quad (\text{III.1.28})$$

Par conséquent, le carré de la vitesse,  $\|v_1\|^2 = v_1^T v_1$ , du centre de masse devient

$$v_1^T v_1 = l_{c1}^2 \dot{q}_1^2 \quad (\text{III.1.29})$$

Enfin, l'énergie cinétique correspondant au mouvement de la liaison 1 peut être obtenue comme

$$\begin{aligned} E_1(q, \dot{q}) &= \frac{1}{2} m_1 v_1^T v_1 + \frac{1}{2} I_1 \dot{q}_1^2 \\ &= \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 \end{aligned} \quad (\text{III.1.30})$$

D'autre part, les coordonnées du centre de masse de lien 2, exprimée dans le plan  $x$ - $y$ .

$$x_2 = l_1 \sin(q_1) + l_{c2} \sin(q_1 + q_2) \quad (\text{III.1.31})$$

$$y_2 = -l_1 \cos(q_1) - l_{c2} \cos(q_1 + q_2) \quad (\text{III.1.32})$$

Par conséquent, le vecteur de vitesse  $v_2$  du centre de masse d'un tel lien est :

$$v_2 = \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} \quad (\text{III.1.33})$$

$$= \begin{bmatrix} l_1 \cos(q_1) \dot{q}_1 + l_{c2} \cos(q_1 + q_2) [\dot{q}_1 + \dot{q}_2] \\ l_1 \sin(q_1) \dot{q}_1 + l_{c2} \sin(q_1 + q_2) [\dot{q}_1 + \dot{q}_2] \end{bmatrix} \quad (\text{III.1.34})$$

Par conséquent, en utilisant les identités trigonométriques

$$\cos(\theta)^2 + \sin(\theta)^2 = 1$$

Et

$$\sin(q_1) \sin(q_1 + q_2) + \cos(q_1) \cos(q_1 + q_2) = \cos(q_2),$$

Nous ne concluons que la vitesse carrée,

$$\|v_2\|^2 = v_2^T v_2 \quad (\text{III.1.35})$$

Du centre de masse de lien 2 satisfait

$$v_2^T v_2 = l_1^2 \dot{q}_1^2 + l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + 2l_1 l_{c2} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) \quad (\text{III.1.36})$$

Ce qui implique que

$$\begin{aligned} E_2(q, \dot{q}) &= \frac{1}{2} m_2 v_2^T v_2 + \frac{1}{2} I_2 [\dot{q}_1 + \dot{q}_2]^2 \\ &= \frac{m_2}{2} l_1^2 \dot{q}_1^2 + \frac{m_2}{2} l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c2} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) + \\ &\frac{1}{2} I_2 [\dot{q}_1 + \dot{q}_2]^2 \end{aligned} \quad (\text{III.1.37})$$

De même, l'énergie potentielle peut être décomposée en la somme des termes

$$U(q) = U_1(q) + U_2(q) \quad (\text{III.1.38})$$

Ou  $U_1(q)$  et  $U_2(q)$  sont les énergies potentielles associées aux masses  $m_1$  et  $m_2$ , respectivement. Ainsi, en supposant que l'énergie potentielle est nulle en  $y = 0$ , on obtient

$$U_1(q) = -m_1 l_{c2} g \cos(q_1) \quad (\text{III.1.39})$$

Et

$$U_2(q) = -m_2 l_1 g \cos(q_1) - m_2 l_{c2} g \cos(q_1 + q_2) \quad (\text{III.1.40})$$



À partir des équations (III.1.30) – (III.1.40), nous obtenons le Lagrangien comme :

$$L(q, \dot{q}) = E(q, \dot{q}) - U(q) \quad (\text{III.1.41})$$

$$= E_1(q, \dot{q}) + E_2(q, \dot{q}) - U_1(q) - U_2(q) \quad (\text{III.1.42})$$

$$= \frac{1}{2} [m_1 l_{c1}^2 + m_2 l_1^2] \dot{q}_1^2 + \frac{1}{2} m_2 l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c2} \cos(q_2) [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] + [m_1 l_{c1} + m_2 l_1] g \cos(q_1) + m_2 g l_{c2} \cos(q_1 + q_2) + \frac{1}{2} I_1 \dot{q}_1^2 + \frac{1}{2} I_2 [\dot{q}_1 + \dot{q}_2]^2 \quad (\text{III.1.43})$$

A partir de cette dernière équation, on obtient l'expression suivante :

$$\frac{\partial L}{\partial \dot{q}_1} = [m_1 l_{c1}^2 + m_2 l_1^2] \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + 2 m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_2 + I_1 \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2] \quad (\text{III.1.44})$$

$$\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{q}_1} \right] = [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2 m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_1 + [m_1 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_2 - 2 m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2^2 + I_1 \ddot{q}_1 + I_2 [\ddot{q}_1 + \ddot{q}_2] \quad (\text{III.1.45})$$

$$\frac{\partial L}{\partial q_1} = -[m_1 l_{c1} + m_2 l_1] g \sin(q_1) - m_2 g l_{c2} \sin(q_1 + q_2) \quad (\text{III.1.46})$$

$$\frac{\partial L}{\partial \dot{q}_2} = m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2] \quad (\text{III.1.47})$$

$$\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{q}_2} \right] = m_2 l_{c2}^2 \ddot{q}_1 + m_2 l_{c2}^2 \ddot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \ddot{q}_1 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 + I_2 [\ddot{q}_1 + \ddot{q}_2] \quad (\text{III.1.48})$$

$$\frac{\partial L}{\partial q_2} = -m_2 l_1 l_{c2} \sin(q_2) [\dot{q}_1 \dot{q}_2 + \dot{q}_1^2] - m_2 g l_{c2} \sin(q_1 + q_2) \quad (\text{III.1.49})$$

Les équations dynamiques qui modélisent le bras de robot sont obtenues en appliquant Les équations de Lagrange (III.1.2),

$$\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{q}_i} \right] - \frac{\partial L}{\partial q_i} = \mathcal{F}_i \quad i = 1, 2$$

à partir de laquelle on obtient finalement :

$$\begin{aligned} \mathcal{F}_1 = & [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + \\ & I_2] \ddot{q}_2 - 2m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2^2 + [m_1 l_{c1} + m_2 l_1] g \sin(q_1) + \\ & m_2 g l_{c2} \sin(q_1 + q_2) \end{aligned} \quad (\text{III.1.50})$$

$$\begin{aligned} \mathcal{F}_2 = & [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + I_2] \ddot{q}_2 + m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1^2 + \\ & m_2 g l_{c2} \sin(q_1 + q_2) \end{aligned} \quad (\text{III.1.51})$$

**Forme matricielle implicite de Lagrange ::**

$$M(q) \ddot{q} + C(q) \dot{q}^2 + B[\dot{q}\dot{q}] + G(q) = \mathcal{F} \quad (\text{III.1.52})$$

$$M(q) \ddot{q} : \quad \text{Forces d'inertie} \quad \mathbf{M}_{n \times n} \quad \ddot{q} : \quad : n \times 1$$

$$C(q) \dot{q}^2 : \quad \text{Forces centrifuges} \quad \mathbf{C}_{n \times n} : \quad \dot{q}^2 \quad : n \times 1$$

$$B[\dot{q}\dot{q}] : \quad \text{Forces de Coriolis} \quad \mathbf{B}_{n \times n(n-1)/2} \quad [\dot{q}\dot{q}] \quad : (n \times (n-1)/2) \times 1$$

$$G(q) : \quad \text{Forces de Gravité} \quad \mathbf{G}_{n \times 1}$$

$$\mathcal{F} : \quad \text{Forces généralisées (couples - Forces)} \quad \mathcal{F} : n \times 1;$$

$$M = \begin{bmatrix} m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2 & m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2 \\ m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix}$$

(III.1.53)

$$C = \begin{bmatrix} 0 & -m_2 l_1 l_{c2} \sin(q_2) \\ m_2 l_1 l_{c2} \sin(q_2) & 0 \end{bmatrix} \quad (\text{III.1.54})$$

$$B = \begin{bmatrix} -2m_2 l_1 l_{c2} \sin(q_2) \\ 0 \end{bmatrix} \quad (\text{III.1.55})$$

$$G = \begin{bmatrix} [m_1 l_{c1} + m_2 l_1] g \sin(q_1) + m_2 g l_{c2} \sin(q_1 + q_2) \\ m_2 g l_{c2} \sin(q_1 + q_2) \end{bmatrix} \quad (\text{III.1.56})$$

$$\text{Autre expression : } M(q)\ddot{q} + V(q, \dot{q}) + G(q) = \mathcal{T} \quad (\text{III.1.57})$$

$$\text{Avec } V(q, \dot{q}) = C(q)\dot{q}^2 + B(q)[\dot{q}\dot{q}] \quad (\text{III.1.58})$$

L'équation dynamique globale s'écrira en ajoutant les forces de frottements :

$$M(q)\ddot{q} + V(q, \dot{q}) + G(q) + F(q, \dot{q}) = \mathcal{T} \quad (\text{III.1.59})$$

Nous avons renvoyé en annexe A, le programme Matlab qui calcule analytiquement les matrices M,C,B et G à partir de la donnée des paramètres statiques DH (modifiés) et dynamiques à partir des équations III.1.13 à III.1.18. ce programme utilise la fonction qui calcule le jacobien des différentes liaisons qui est également donné dans la même annexe.

Où  $\mathcal{T}_1$  et  $\mathcal{T}_2$  sont les couples externes délivrés par les actionneurs au niveau des joints **1** et **2**.

Ainsi, les équations dynamiques du robot (III.1.58)-(III.1.59) constituent un ensemble de deux équations différentielles non linéaires matricielles des variables d'état  $x = [q^T \dot{q}^T]^T$ , c'est-à-dire de la forme, dans l'espace articulaire ou des variables généralisées :

$$f_{EL}(q, \dot{q}, \ddot{q}, \tau) = \mathbf{0}, \quad (\text{III.1.60})$$

Et, dans l'espace opérationnel ou cartésien obtenu par géométrie inverse :

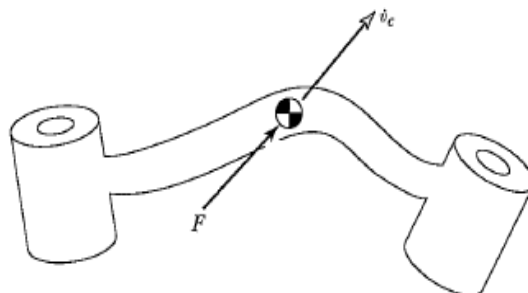
$$f_C(X, \dot{X}, \ddot{X}, \tau) = \mathbf{0}, \quad (\text{III.1.61})$$

Le vecteur  $\tau$  représente les forces et les couples appliqués au niveau des articulations par les actionneurs. Le modèle dynamique (III.1.60) est le modèle dynamique dans l'espace articulaire, tandis que (III.1.61) correspond au modèle dynamique dans l'espace opérationnel. Dans ce qui suit, nous nous concentrons sur le modèle dynamique dans l'espace articulaire.

### III.2. Approche par les équations de EULER -NEWTON

#### III.2.1. Equation de EULER -NEWTON

Nous examinerons chaque maillon d'un manipulateur comme un corps rigide. Si nous connaissons l'emplacement du centre de masse et le tenseur d'inertie de la liaison, sa répartition des masses est complètement caractérisée. Afin de déplacer les liens, nous devons soit les accélérer, soit les décélérer. Les forces nécessaires pour un tel mouvement sont fonction de l'accélération souhaitée et de la distribution de masse des liaisons. L'équation de Newton pour les mouvements de translation, ainsi que son analogue en rotation, (équation d'Euler), décrit comment les forces inerties et accélérations agissent [12].



**Figure III.2.1 :** Une force  $F$  agissant au centre de masse provoque une accélération  $\dot{v}_c$ .

### III.2.2. Équation de Newton

**Figure III.2.1 :** montre un corps rigide dont le centre de masse accélère avec l'accélération  $\dot{v}_c$ . Dans une telle situation la force  $F$  agissant au centre de gravité est donnée par l'équation de Newton.

$$F = m \dot{v}_c \quad (\text{III.2.1})$$

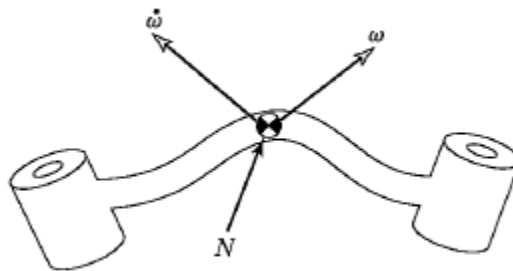
Où  $m$  est la masse totale du corps.

### III.2.3. Équation d'Euler

La Figure III.2.2 montre un corps rigide en rotation avec la vitesse angulaire  $\omega$  et l'accélération angulaire  $\dot{\omega}$ . Dans une telle situation, le moment  $N$ , qui doit agir sur le corps et qui est à l'origine de ce mouvement, est donné par l'équation d'Euler.

$$N = {}^c I \dot{\omega} + \omega \times {}^c I \omega \quad (\text{III.2.2})$$

Où  ${}^c I$  est le tenseur d'inertie du corps, calculé par rapport au repère lié au centre de masse  $\{C\}$



**Figure III.2.2:** Un moment  $N$  agit sur un corps, lequel sera en rotation avec une vitesse  $\omega$  et une accélération angulaire  $\dot{\omega}$ .

### III.2.4. Formulation itérative de Newton-Euler

Nous considérons maintenant le problème du calcul des couples qui correspondent à une trajectoire donnée d'un manipulateur. Nous supposons que nous connaissons les position, vitesse et accélération des articulations,  $(\Theta, \dot{\Theta}, \ddot{\Theta})$ . Avec ces connaissances et l'information sur la cinématique et la distribution des masses du robot, nous pouvons calculer les couples articulaires requis pour provoquer ce mouvement. L'algorithme présenté est basé sur la méthode publiée par **Luh, Walker** et **Paul** dans [13].

#### III.2.4.1. Itérations vers l'extérieur pour calculer les vitesses et accélérations

Afin de calculer les forces d'inertie agissant sur les liens, il est nécessaire de calculer la vitesse de rotation et l'accélération linéaire et de rotation du centre de gravité de chaque lien du manipulateur à un instant donné. Ces calculs doivent se faire de manière itérative, en commençant par le lien 1 et en se déplaçant successivement, liaison par liaison, vers l'extérieur jusqu'au lien  $n$ .

La « propagation » de la vitesse de rotation de lien à lien est donnée (l'articulation  $i + 1$  étant rotoïde) par :

$${}^{i+1}\boldsymbol{\omega}_{i+1} = {}^{i+1}\mathbf{R}^i \boldsymbol{\omega}_i + \dot{\theta}_{i+1} {}^{i+1}\hat{\mathbf{Z}}_{i+1} \quad (\text{III.2.3})$$

On obtient l'équation pour transformer l'accélération angulaire d'un lien à l'autre :

$${}^{i+1}\dot{\boldsymbol{\omega}}_{i+1} = {}^{i+1}\mathbf{R}^i \dot{\boldsymbol{\omega}}_i + {}^{i+1}\mathbf{R}^i \boldsymbol{\omega}_i \times \dot{\theta}_{i+1} {}^{i+1}\hat{\mathbf{Z}}_{i+1} + \ddot{\theta}_{i+1} {}^{i+1}\hat{\mathbf{Z}}_{i+1} \quad (\text{III.2.4})$$

Lorsque l'articulation ( $i+1$ ) est prismatique, cette équation se simplifie à

$${}^{i+1}\dot{\boldsymbol{\omega}}_{i+1} = {}^{i+1}\mathbf{R}^i \dot{\boldsymbol{\omega}}_i \quad (\text{III.2.5})$$

L'accélération linéaire de chaque origine du repère lié à l'articulation est :

$${}^{i+1}\dot{\mathbf{v}}_{i+1} = {}^{i+1}\mathbf{R}^i [ {}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_{i+1} + {}^i\boldsymbol{\omega}_i \times ( {}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_{i+1} ) + {}^i\dot{\mathbf{v}}_i ] \quad (\text{III.2.6})$$

Pour la liaison prismatique  $i + 1$ ,

$$\begin{aligned} {}^{i+1}\dot{\mathbf{v}}_{i+1} &= {}^{i+1}\mathbf{R}^i ( {}^i\dot{\boldsymbol{\omega}}_i \times {}^i\mathbf{P}_{i+1} + {}^i\boldsymbol{\omega}_i \times ( {}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_{i+1} ) + {}^i\dot{\mathbf{v}}_i ) \\ &+ 2 {}^{i+1}\boldsymbol{\omega}_{i+1} \times \dot{d}_{i+1} {}^{i+1}\hat{\mathbf{Z}}_{i+1} + \ddot{d}_{i+1} {}^{i+1}\hat{\mathbf{Z}}_{i+1} \end{aligned} \quad (\text{III.2.7})$$

Nous aurons également besoin de l'accélération linéaire du centre de gravité de chaque lien,

$${}^i\dot{\mathbf{v}}_{Ci} = {}^i\dot{\boldsymbol{\omega}}_i \times {}^i\mathbf{P}_{Ci} + {}^i\boldsymbol{\omega}_i \times ( {}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_{Ci} ) + {}^i\dot{\mathbf{v}}_i \quad (\text{III.2.8})$$

Ici, nous imaginons un cadre,  $\{C_i\}$  attachée à chaque lien, ayant son origine située au centre de masse de la liaison et ayant la même orientation que celle du repère lié à l'articulation  $\{i\}$ .

**Équation (III.2.8)** n'implique pas de mouvement de l'articulation et reste donc valable pour l'articulation  $i + 1$ , qu'elle soit rotoïde ou prismatique.

A noter que l'application de ces équations au lien 1 est particulièrement simple, car

$${}^0\boldsymbol{\omega}_0 = {}^0\dot{\boldsymbol{\omega}}_0 = \mathbf{0} \quad (\text{III.2.9})$$

### III.2.4.2. La force et le couple agissant sur un lien

Ayant calculé les accélérations linéaires et angulaires du centre de masse de chaque lien, on peut déterminer les forces et les couples agissant agissant au centre de gravité de chaque lien, en appliquant les équations de **Euler-Newton**. Ainsi, nous avons

$$F_i = m \dot{v}_{C_i} \quad (\text{III.2.10})$$

$$N_i = {}^{C_i}I\dot{\omega}_i + \omega_i \times {}^{C_i}I\omega_i \quad (\text{III.2.11})$$

Où  $\{C_i\}$  a son origine au centre de masse de la liaison et a la même orientation que le repère lié au lien,  $\{i\}$ .

### III.2.4.3. Itérations intérieures pour le calcul des forces et des couples

Après avoir calculé les forces et les couples agissant sur chaque lien, nous devons maintenant calculer les couples généralisés qui se traduiront par des forces nettes et des couples appliqués à chaque articulation.

Nous pouvons le faire en écrivant une équation bilan-force et bilan-moment basée sur le diagramme de corps libre d'un lien typique. (**Figure III.2.3**). Chaque lien a des forces et couples qui s'exercent sur lui par ses voisins et de plus subit une force d'inertie et du couple.

$f_i$  = force exercée sur le lien  $i$  par lien  $i-1$ ,

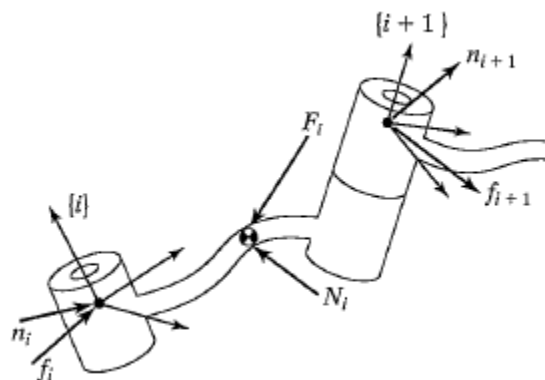
$n_i$  = couple exercée sur le lien  $i$  par lien  $i-1$ ,

En additionnant les forces qui agissent sur le lien  $i$ , nous arrivons à la relation d'équilibre des forces

$${}^iF_i = {}^if_i - {}_{i+1}R^{i+1}f_{i+1} \quad (\text{III.2.11})$$

En additionnant les couples sur le centre de masse et en les égalisant à zéro, nous arrivons à l'équation du couple-bilan :

$${}^iN_i = {}^in_i - {}^in_{i+1} + (-{}^iP_{ci}) \times {}^if_i - ({}^iP_{i+1} - {}^iP_{ci}) \times {}^if_{i+1} \quad (\text{III.2.12})$$



**Figure III.2.3:** L'équilibre de forces incluant les forces inertielles un manipulateur à un seul lien

En utilisant le résultat de la relation d'équilibre des forces (III.2.11) et après quelques manipulations de matrices de rotation, on peut écrire (III.2.12) comme :

$${}^iN_i = {}^i n_i - {}_{i+1}{}^iR^{i+1}n_{i+1} - {}^iP_{ci} \times {}^iF_i - {}^iP_{i+1} \times {}_{i+1}{}^iR^{i+1}f_{i+1} \quad (\text{III.2.13})$$

Enfin, nous pouvons réarranger les équations de la force et le couple afin qu'elles apparaissent en tant que relations itératives de voisin le plus éloigné (portant le numéro le plus élevé) au voisin le plus proche (portant le numéro le plus faible) :

$${}^if_i = {}_{i+1}{}^iR^{i+1}f_{i+1} + {}^iF_i \quad (\text{III.2.14})$$

$${}^i n_i = {}^iN_i + {}_{i+1}{}^iR^{i+1}n_{i+1} + {}^iP_{ci} \times {}^iF_i + {}^iP_{i+1} \times {}_{i+1}{}^iR^{i+1}f_{i+1} \quad (\text{III.2.15})$$

Ces équations sont évaluées liaison par liaison, à partir de lien  $n$  et en travaillant vers l'intérieur en direction de la base du robot. Ces itérations de force vers l'intérieur sont analogues aux itérations forces statiques, sauf que les couples et les forces d'inertie sont désormais considérés au niveau de chaque liaison.

Comme dans le cas statique, les couples mixtes requis sont déterminés en prenant la composante  $\widehat{Z}$  du couple appliqué par un lien sur son voisin :

$$\tau_i = {}^i n_i^T \widehat{Z}_i \quad (\text{III.2.16})$$

Pour une articulation  $i$  prismatique, nous utilisons :

$$\tau_i = {}^if_i^T \widehat{Z}_i \quad (\text{III.2.17})$$

Où nous avons utilisé le symbole  $\tau$  pour une force linéaire de l'actionneur

Notons que pour un robot se déplaçant dans l'espace libre  ${}^{N+1}f_{N+1}$  et  ${}^{N+1}n_{N+1}$  sont fixées à zéro, et donc la première application des équations pour le lien  $n$  est très simple. Si le robot est en contact avec l'environnement, les forces et les couples en raison de ce contact peuvent être inclus dans l'équilibre des forces en ayant  ${}^{N+1}f_{N+1}$  et  ${}^{N+1}n_{N+1}$  différents de zéro.

#### III.2.4.4. L' algorithme dynamique itératif de Newton-Euler

L'algorithme complet pour calculer les couples articulaires à partir des mouvements des articulations se compose de deux parties. Tout d'abord, les vitesses et accélérations des liaisons sont calculées itérativement à partir du lien 1 vers le lien  $n$  et les équations d'Euler-Newton sont appliquées à chaque lien. Deuxièmement, les forces et les couples d'interaction et les couples des actionneurs agissant sur les articulations sont calculés récursivement de lien  $n$  vers le lien 1. Les équations sont ensuite résumées pour le cas où tous les articulations sont rotoïdes.

➤ Itérations extérieures:  $i : 0 \longrightarrow 5$

$${}^{i+1}\omega_{i+1} = {}^{i+1}{}_iR {}^i\omega_i + \dot{\theta}_{i+1} {}^{i+1}\widehat{Z}_{i+1} \quad (\text{III.2.18})$$



$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}R^i \dot{\omega}_i + {}^{i+1}R^i \omega_i \times \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (\text{III.2.19})$$

$${}^{i+1}\dot{\nu}_{i+1} = {}^{i+1}R^i (\dot{\omega}_i \times {}^iP_{i+1} + \omega_i \times (\omega_i \times {}^iP_{i+1})) + \dot{\nu}_i \quad (\text{III.2.20})$$

$${}^{i+1}\dot{\nu}_{Ci+1} = {}^{i+1}\dot{\omega}_{i+1} \times {}^{i+1}P_{Ci+1} + {}^{i+1}\omega_{i+1} \times ({}^{i+1}\omega_{i+1} \times {}^{i+1}P_{Ci+1}) + {}^{i+1}\dot{\nu}_{i+1} \quad (\text{III.2.21})$$

$${}^{i+1}F_{i+1} = m_{i+1} {}^{i+1}\dot{\nu}_{Ci+1} \quad (\text{III.2.22})$$

$${}^{i+1}N_{i+1} = {}^{Ci+1}I_{i+1} {}^{i+1}\dot{\omega}_{i+1} + {}^{i+1}\omega_{i+1} \times {}^{Ci+1}I_{i+1} {}^{i+1}\omega_{i+1} \quad (\text{III.2.23})$$

➤ Itérations intérieures :  $i : 6 \longrightarrow$

$${}^i f_i = {}^{i+1}R^{i+1} f_{i+1} \quad {}^i F_i \quad (\text{III.2.24})$$

$${}^i n_i = {}^i N_i + {}^{i+1}R^{i+1} n_{i+1} + {}^i P_{Ci} \times {}^i F_i + {}^i P_{i+1} \times {}^{i+1}R^{i+1} f_{i+1} \quad (\text{III.2.25})$$

$$\tau_i = {}^i n_i^T {}^i \hat{Z}_i \quad (\text{III.2.26})$$

### III.2.4.5. Prise en compte des forces de gravité dans l'algorithme dynamique

L'effet de la gravité sur les liens de chargement peut être inclus tout simplement en définissant

${}^0\dot{\nu}_0 = \mathbf{G}$ , où  $\mathbf{G}$  est l'amplitude du vecteur de la gravité mais pointant dans la direction opposée. Cela revient à dire que la base du robot s'accélère vers le haut avec une accélération de  $\mathbf{1g}$ . Cette accélération ascendante fictive provoque exactement le même effet sur les liens comme le ferait la gravité. Ainsi, avec aucune dépense supplémentaire du point de vue du volume de calcul, l'effet de la gravité est calculé.

### V.2.5. Application de la formulation de Newton-Euler sur un manipulateur à deux-lien (2DDL)

Ici, nous calculons les équations dynamiques sous forme finie pour le manipulateur planaire à deux-liens illustré à la **figure III.2.4**. Pour simplifier, nous supposons que la répartition des masses est extrêmement simple : toute la masse existe sous la forme d'une masse ponctuelle à l'extrémité de chaque lien. Ces masses sont  $m_1$  et  $m_2$ .

Tout d'abord, nous déterminons les valeurs des quantités diverses qui apparaîtront dans l'équation le récursive de Newton-Euler. Les vecteurs qui situent le centre de masse pour chaque lien sont :

$${}^1P_{C1} = l_1 \hat{X}_1 \quad (\text{III.2.27})$$

$${}^2P_{C2} = l_2 \hat{X}_2 \quad (\text{III.2.28})$$

En raison de l'hypothèse de masse ponctuelle, le tenseur d'inertie écrit au centre de masse pour chaque lien est la matrice nulle :

$${}^{c^1}I_1 = \mathbf{0} \quad (\text{III.2.29})$$

$${}^{c^2}I_2 = \mathbf{0} \quad (\text{III.2.30})$$

Il n'y a pas de forces agissant sur l'effecteur, donc nous avons

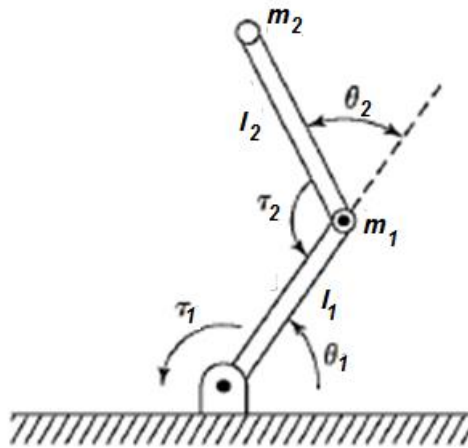
$$f_3 = \mathbf{0} \quad (\text{III.2.31})$$

$$n_3 = \mathbf{0} \quad (\text{III.2.32})$$

La base du robot ne tourne pas ; par conséquent, nous avons

$$\omega_0 = \mathbf{0} \quad (\text{III.2.33})$$

$$\dot{\omega}_0 = \mathbf{0} \quad (\text{III.2.34})$$



**Figure III.2.4:** Manipulateur à deux-liens planaires avec masses ponctuelles aux extrémités des liens

Pour inclure des forces de gravité, nous utiliserons

$${}^0\dot{v}_0 = g \hat{Y}_0 \quad (\text{III.2.35})$$

La rotation entre les images successives de lien est donnée par

$${}^{i+1}R_i = \begin{bmatrix} c_{i+1} & -s_{i+1} & 0 \\ s_{i+1} & c_{i+1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{III.2.36})$$

$${}^{i+1}R_i = \begin{bmatrix} c_{i+1} & s_{i+1} & 0 \\ -s_{i+1} & c_{i+1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{III.2.37})$$

Nous appliquons maintenant les équations (III.2.18) à (III.2.26).

Les itérations vers l'extérieur pour lien 1 sont les suivants :

$${}^1\omega_1 = \dot{\theta}_1 {}^1\hat{Z}_1 = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \dot{\theta}_1 \end{bmatrix} \quad (\text{III.2.38})$$

$${}^1\dot{\omega}_1 = \ddot{\theta}_1 {}^1\hat{Z}_1 = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \ddot{\theta}_1 \end{bmatrix} \quad (\text{III.2.39})$$

$${}^1\dot{v}_1 = \begin{bmatrix} c_1 & s_1 & 0 \\ -s_1 & c_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ g \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} g s_1 \\ g c_1 \\ 0 \end{bmatrix} \quad (\text{III.2.40})$$

$${}^1\dot{v}_{c1} = \begin{bmatrix} \mathbf{0} \\ l_1 \dot{\theta}_1 \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} -l_1 \dot{\theta}_1^2 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} g s_1 \\ g c_1 \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} -l_1 \dot{\theta}_1^2 + g s_1 \\ l_1 \ddot{\theta}_1 + g c_1 \\ \mathbf{0} \end{bmatrix} \quad (\text{III.2.41})$$

$${}^1F_1 = \begin{bmatrix} -m_1 l_1 \dot{\theta}_1^2 + m_1 g s_1 \\ m_1 l_1 \ddot{\theta}_1 + m_1 g c_1 \\ \mathbf{0} \end{bmatrix} \quad (\text{III.2.42})$$

$${}^1N_1 = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (\text{III.2.43})$$

Les itérations vers l'extérieur pour le lien 2 sont comme suit :

$${}^2\omega_2 = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix} \quad (\text{III.2.44})$$

$${}^2\dot{\omega}_2 = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} \quad (\text{III.2.45})$$

$${}^2\dot{\mathbf{v}}_2 = \begin{bmatrix} c_2 & s_2 & 0 \\ -s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -l_1\dot{\theta}_1^2 + g s_1 \\ l_1\ddot{\theta}_1 + g c_1 \\ 0 \end{bmatrix} = \begin{bmatrix} l_1\ddot{\theta}_1 s_2 - l_1\dot{\theta}_1^2 c_2 + g s_{12} \\ l_1\ddot{\theta}_1 c_2 + l_1\dot{\theta}_1^2 s_2 + g c_{12} \\ 0 \end{bmatrix} \quad (\text{III.2.46})$$

$${}^2\dot{\mathbf{v}}_{c2} = \begin{bmatrix} 0 \\ l_2(\ddot{\theta}_1 + \ddot{\theta}_2) \\ 0 \end{bmatrix} + \begin{bmatrix} -l_2(\dot{\theta}_1 + \dot{\theta}_2)^2 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} l_1\ddot{\theta}_1 s_2 - l_1\dot{\theta}_1^2 c_2 + g s_{12} \\ l_1\ddot{\theta}_1 c_2 + l_1\dot{\theta}_1^2 s_2 + g c_{12} \\ 0 \end{bmatrix} \quad (\text{III.2.47})$$

$${}^2F_2 = \begin{bmatrix} m_2 l_1 \ddot{\theta}_1 s_2 - m_2 l_1 \dot{\theta}_1^2 c_2 + m_2 g s_{12} - m_2 l_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \\ m_2 l_1 \ddot{\theta}_1 c_2 + m_2 l_1 \dot{\theta}_1^2 s_2 + m_2 g c_{12} + m_2 l_2 (\ddot{\theta}_1 + \ddot{\theta}_2) \\ 0 \end{bmatrix} \quad (\text{III.2.48})$$

$${}^2N_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{III.2.49})$$

Les itérations vers l'intérieur pour le lien 2 sont comme suit :

$${}^2\mathbf{f}_2 = {}^2F_2$$

$${}^2\mathbf{n}_2 = \begin{bmatrix} 0 \\ 0 \\ m_2 l_1 l_2 c_2 \ddot{\theta}_1 + m_2 l_1 l_2 s_2 \dot{\theta}_1^2 + m_2 l_2 g c_{12} + m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) \end{bmatrix} \quad (\text{III.2.50})$$

Les itérations vers l'intérieur pour lien 1 sont les suivants :

$${}^1\mathbf{f}_1 = \begin{bmatrix} c_2 & -s_2 & 0 \\ s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_2 l_1 s_2 \ddot{\theta}_1 - m_2 l_1 c_2 \dot{\theta}_1^2 + m_2 g s_{12} - m_2 l_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \\ m_2 l_1 c_2 \ddot{\theta}_1 + m_2 l_1 s_2 \dot{\theta}_1^2 + m_2 g c_{12} + m_2 l_2 (\ddot{\theta}_1 + \ddot{\theta}_2) \\ 0 \end{bmatrix} +$$

$$\begin{bmatrix} -m_1 l_1 \dot{\theta}_1^2 + m_1 g s_1 \\ m_1 l_1 \ddot{\theta}_1 + m_1 g c_1 \\ 0 \end{bmatrix} \quad (\text{III.2.51})$$

$$\begin{aligned}
{}^1\mathbf{n}_1 = & \begin{bmatrix} 0 \\ 0 \\ m_2 l_1 l_2 c_2 \ddot{\theta}_1 + m_2 l_1 l_2 s_2 \dot{\theta}_1^2 + m_2 l_2 g c_{12} + m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ m_1 l_1^2 \ddot{\theta}_1 + m_1 l_1 g c_1 \end{bmatrix} \\
+ & \begin{bmatrix} 0 \\ 0 \\ m_2 l_1^2 \ddot{\theta}_1 + m_2 l_1 l_2 s_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 l_1 g s_2 s_{12} + m_2 l_1 l_2 c_2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_2 g c_2 c_{12} \end{bmatrix}
\end{aligned} \tag{III.2.52}$$

Extrayons les composantes de  $\widehat{\mathbf{Z}}$  de  ${}^1\mathbf{n}_1$ , et  ${}^2\mathbf{n}_2$ , nous trouvons les couples articulaires déjà établis par la méthode directe du Lagrangien et que le logiciel que nous avons mis au point détermine facilement sous condition d'une définition correcte des paramètres statiques et dynamiques.

$$\begin{aligned}
\tau_1 = & m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_1 l_2 c_2 (2\ddot{\theta}_1 + \ddot{\theta}_2) + (m_1 + m_2) l_1^2 \ddot{\theta}_1 - m_2 l_1 l_2 s_2 \dot{\theta}_2^2 - \\
& 2m_2 l_1 l_2 s_2 \dot{\theta}_1 \dot{\theta}_2 + m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1
\end{aligned} \tag{III.2.53}$$

$$\tau_2 = m_2 l_1 l_2 c_2 \ddot{\theta}_1 + m_2 l_1 l_2 s_2 \dot{\theta}_1^2 + m_2 l_2 g c_{12} + m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) \tag{III.2.29} \tag{III.2.54}$$

Les équations (III.2.29) donnent des expressions des couples appliqués par les actionneurs sur les articulations en fonction de la position, de la vitesse et de l'accélération articulaires. Notons que ces fonctions assez complexes découlent de l'un des plus simples manipulateurs imaginables où les masses ont été supposées être aux extrémités et en supposant des inerties nulles. Évidemment, les équations de la forme fermée (implicite) pour un manipulateur à six degrés de liberté seront beaucoup plus complexes.

## ***CHAPITRE IV***

# ***DYNAMIQUE DE L'ACTIONNEUR***

## IV.1. Dynamique de l'actionneur

Les forces agissant au niveau des articulations sont produites par les actionneurs. Ces derniers peuvent être de type électrique, hydraulique ou pneumatique. Nous nous intéressons particulièrement au cas du moteur à courant continu à aimants permanents qui est le plus fréquemment utilisé dans les manipulateurs robotisés (**figure IV.1**). Le fonctionnement d'un moteur à courant continu est basé sur le principe de la force  $\mathbf{F}$  produite par un conducteur placé dans un champ magnétique et traversé par un courant  $\mathbf{i}$  [15]. La force  $\mathbf{F}$ , le courant  $\mathbf{i}$  et le flux magnétique  $\Phi$  sont liés par la relation suivante :

$$\mathbf{F} = \Phi \times \mathbf{i} \quad (\text{IV.1})$$

Le moteur est constitué d'un stator fixe à l'intérieur duquel se trouve un rotor mobile. Le stator produit un flux magnétique radial  $\Phi$ . Le rotor est traversé par un courant  $\mathbf{i}$ . De ceci, il se produit un couple au niveau du rotor produisant son mouvement. Le couple est donné par l'expression suivante:

$$\tau_m = K\Phi i_a \quad (\text{IV.2})$$

$\tau_m$  : Le couple moteur

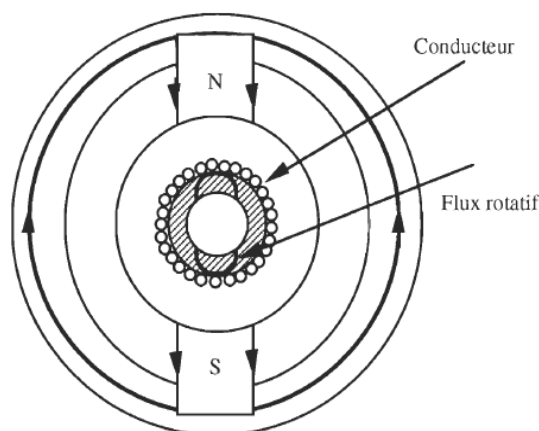
$\Phi$  : Le flux magnétique

$i_a$  : Le courant dans l'armature (rotor)

$K$  : Constante physique

Aussi, lorsqu'un conducteur se déplace dans un champ magnétique, une tension  $V_b$  est générée à travers ses extrémités, elle est proportionnelle à la vitesse du conducteur et s'exprime par la relation :

$$V_b = K_2 \Phi \omega_m \quad (\text{IV.3})$$



**Figure IV.1** Section transversale d'un moteur à aimants permanents

Avec,

$V_b$  : La force électromotrice

$\omega_m$  : La vitesse angulaire du rotor

$K_2$  : Constante de proportionnalité

Les moteurs à courant continu sont classés en fonction de la façon dont est produit le champ magnétique dans les bobines d'induits désignées par les armatures. Dans notre cas, nous discuterons le cas du moteur à aimant permanent à flux constant. Le contrôle de couple agissant sur le rotor peut se faire en contrôlant le courant  $i_a$  de l'armature.

Considérons le schéma de la **figure IV.2**, avec :

$V(t)$  : Tension de l'armature

$L$  : Inductance de l'armature

$R_a$  : Résistance de l'armature

$V_b$  : f.e.m.

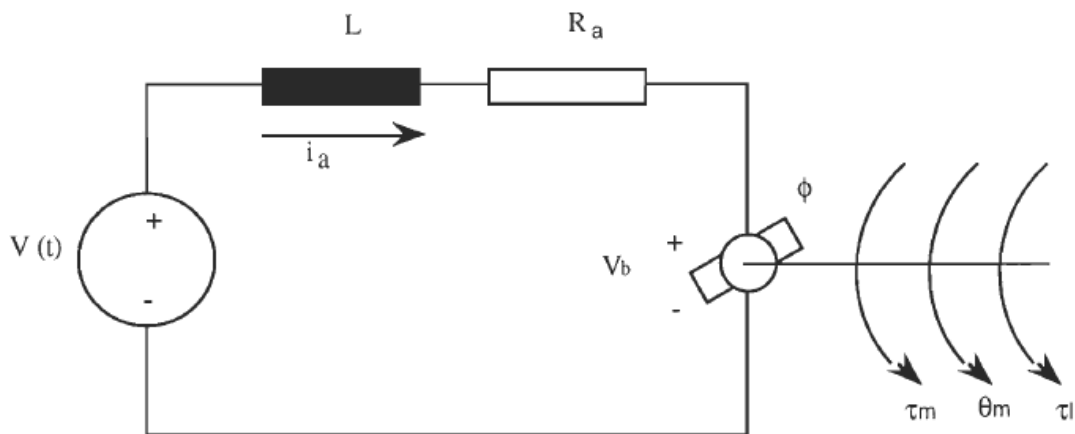
$i_a$  : Courant de l'armature

$\theta_m$  : Position du rotor

$\tau_m$  : Couple moteur

$\tau_l$  : Couple charge

$\Phi$  : Flux magnétique du stator



**Figure IV.2** Schéma équivalent d'un moteur à courant continu

L'équation différentielle du courant de l'armature de l'actionneur contrôlant la liaison  $k$  est :

$$L_K \frac{di_{aK}}{dt} + R_{aK} i_{aK} = V_k(t) - V_{bK}(t) \quad (\text{IV.4})$$

$(k)$  : Indice désignant l'articulation  $k$ .

L'équation mécanique du système suppose que le moteur est connecté au segment par l'intermédiaire d'un réducteur de rapport  $r: 1$  (**figure IV.3**), et est donnée par l'expression suivante:



$$J_{mK} \frac{d^2\theta_{mK}}{dt^2} + B_{mK} \frac{d\theta_{mK}}{dt} = \tau_{mK} - r_K \tau_{IK} \quad (IV.5)$$

Avec,

$$J_{mK} = J_{aK} + J_{rK} \quad (IV.6)$$

$J_{aK}$  : Inertie de l'actionneur  $k$

$J_{rK}$  : Inertie du réducteur  $k$

$B_{mK}$  : La friction de l'actionneur  $k$

Sachant que le flux est constant le couple développé par le moteur et la f.e.m s'expriment par les relations ci-dessous :

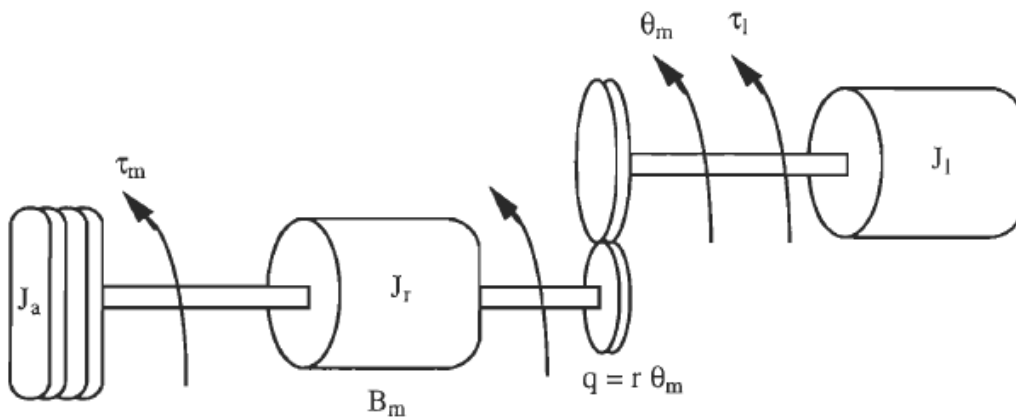


Figure IV.3 Modèle d'un système de réduction

$$\tau_{mK} = K_{IK} \Phi_K i_{aK} = K_{mK} i_{aK} \quad (IV.7)$$

$$V_{bK} = K_{bK} \omega_{mK} = K_{bK} \frac{d\theta_{mK}}{dt} \quad (IV.8)$$

Où,

$K_{mK}$  : La constante de couple exprimée en N.m/amp

$K_{bK}$  : La constante de force électromotrice exprimée V/rad/s

L'équation (IV.4) dans le domaine de Laplace conduit à :

$$L_K S I_{aK}(S) + R_{aK} I_{aK}(S) = V_K(S) - K_{bK} S \theta_{mK}(S) \quad IV.9$$

L'équation (IV.5) dans le domaine de Laplace conduit à :

$$J_{mK} S^2 \theta_{mK}(S) + B_{mK} S \theta_{mK}(S) = K_{mK} I_{aK}(S) - r_K \tau_{IK}(S) \quad IV.10$$

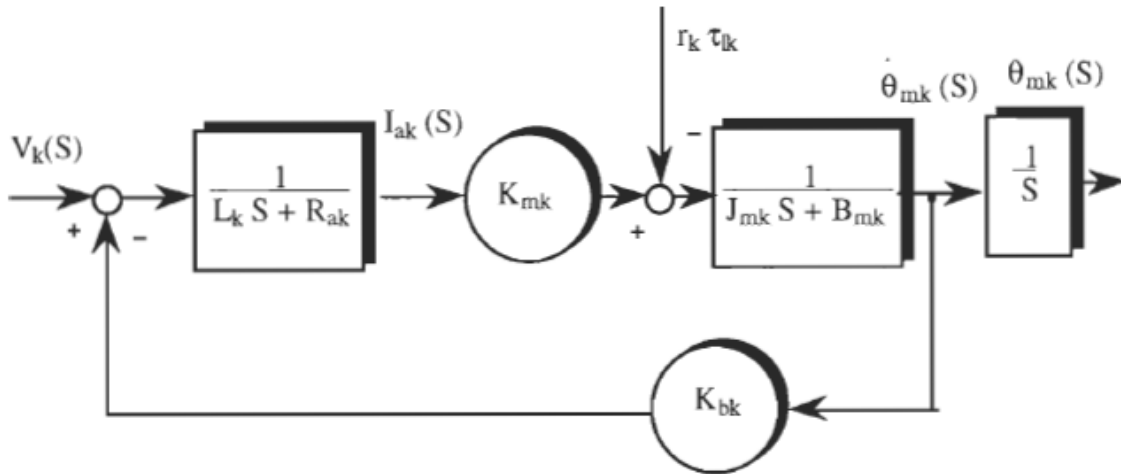
De (IV.8) nous déduisons:

$$I_{aK}(S) = \frac{1}{L_k S + R_{aK}} [V_k(S) - K_{bk} S \theta_{mK}(S)] \quad (\text{IV.11})$$

De (IV.9) nous déduisons:

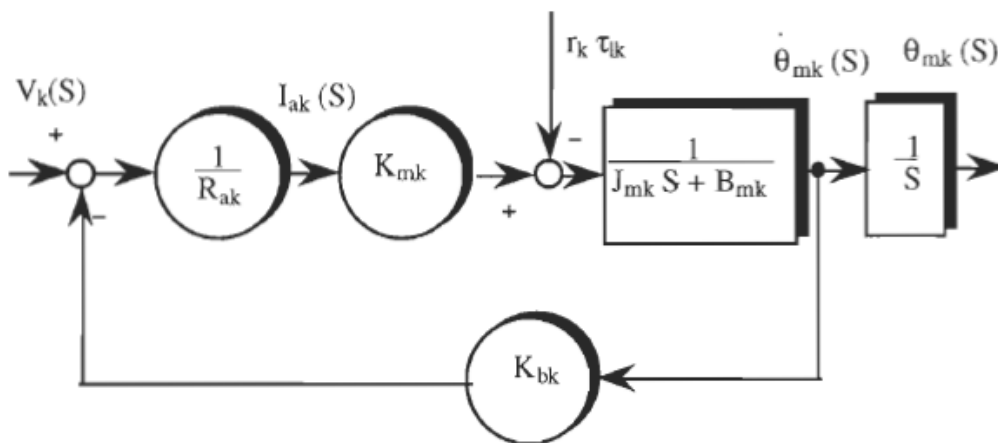
$$\theta_{mK}(S) = \frac{1}{J_{mK} S^2 + B_{mK} S} (K_{mK} I_{aK}(S) - r_k \tau_{lk}(S)) \quad (\text{IV.12})$$

Le schéma bloc représentant ces deux équations est le suivant



**Figure IV.4** Schéma bloc d'un moteur à courant continu

Généralement, la constante de temps électrique  $L/R_a$  est supposé négligeable devant la constante de temps mécanique  $J_m/B_m$ . Cette hypothèse est raisonnable pour la plupart des systèmes électromécaniques et elle nous permettra de réduire l'ordre du système. Le schéma du système d'ordre réduit est donné par la **Figure IV.5**

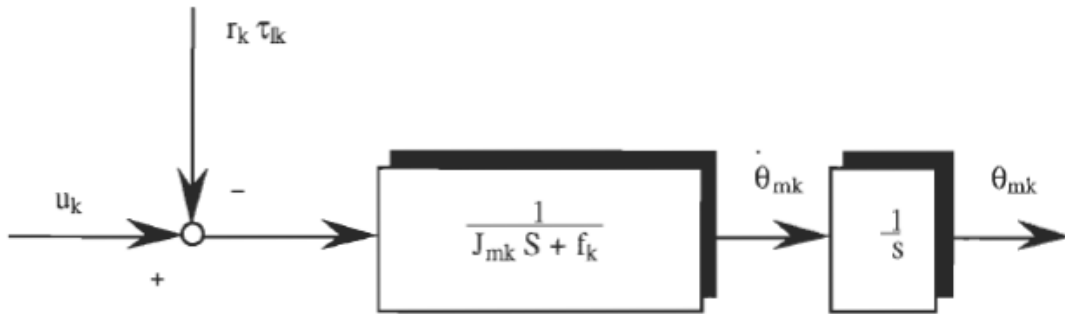


**Figure IV.5** Schéma bloc du système d'ordre réduit

L'élimination de l'intensité  $I_a(t)$  entre les deux équations d'ordre réduit conduit à l'équation suivante du système exprimée dans le domaine temporel :

$$J_{mK}\ddot{\theta}_{mK} + \left( B_{mK} + \frac{K_{bK}K_{mK}}{R_{aK}} \right) \dot{\theta}_{mK} = \left( \frac{K_{mK}}{R_{aK}} V_K(t) - r_K \tau_{lK} \right) \quad (\text{IV.13})$$

Le schéma bloc correspondant à cette équation est montré à la **figure IV.7**.



**Figure IV.6** Schéma bloc du système simplifié

Avec,

$$U_K(s) = \left( \frac{K_{mK}}{R_{aK}} \right) V_K(S) \quad (\text{IV.14})$$

$$f_K = B_{mK} + \frac{K_{bK}K_{mK}}{R_{aK}} \quad (\text{IV.15})$$

$$\theta_{mK} = \frac{1}{r_K} q_K \quad (\text{IV.16})$$

Si la sortie du réducteur est directement liée au segment, les déplacements des articulations sont liés à ceux du rotor par la relation suivante:

$$q_K = r_K(\theta_m)_K \quad k=1,2,\dots,n \quad (\text{IV.17})$$

$r_K$  étant le **kième** rapport de réducteur.

À noter que le couple  $\tau_K$  développé au niveau de l'articulation  $k$ , est donné par l'équation dynamique du bras manipulateur et représente le couple de charge du moteur. En d'autres termes:

$$(\tau_l)_K = \tau_K \quad (\text{IV.18})$$

Ou ;  $K=1,2,\dots, n$ .

## IV.2. Dynamique complète du système robotique : Actionneur + Robot

Finalement les équations décrivant le mouvement du manipulateur au complet sont:

$$\sum_{j=1}^n \mathbf{m}_{jK}(\mathbf{q}) \ddot{\mathbf{q}}_j + \sum_{i,j=1}^n \mathbf{C}_{ij}(\mathbf{q}) \dot{\mathbf{q}}_i \dot{\mathbf{q}}_j + \mathbf{G}_K(\mathbf{q}) = \boldsymbol{\tau}_K \quad (\text{IV.19})$$

$$\mathbf{J}_{mK} \ddot{\boldsymbol{\theta}}_{mK} + \mathbf{f}_K \dot{\boldsymbol{\theta}}_{mK} = \mathbf{U}_K - \mathbf{r}_K \boldsymbol{\tau}_K \quad (\text{IV.20})$$

De plus, en utilisant la relation :

$$\mathbf{q} = \mathbf{r} \boldsymbol{\theta}_m \quad (\text{IV.21})$$

Nous démontrons l'équation qui décrit le modèle du manipulateur au complet:

$$\frac{1}{r^2_K} \mathbf{J}_{mK} \ddot{\mathbf{q}}_K + \sum_{j=1}^n \mathbf{m}_{jK}(\mathbf{q}) \ddot{\mathbf{q}}_j + \sum_{i,j=1}^n \mathbf{C}_{ij}(\mathbf{q}) \dot{\mathbf{q}}_i \dot{\mathbf{q}}_j + \frac{1}{r^2_K} \mathbf{f}_K \dot{\mathbf{q}}_K + \mathbf{G}_K(\mathbf{q}) = \mathbf{U}_K \quad (\text{IV.22})$$

$\sum_{i,j=1}^n \mathbf{C}_{ij}(\mathbf{q}) \dot{\mathbf{q}}_i \dot{\mathbf{q}}_j$  correspond à la matrice  $\mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})$  qui regroupe les force de coriolis et centrifuges  
 Cette équation peut s'écrire sous la forme matricielle suivante:

$$(\mathbf{M}(\mathbf{q}) + \mathbf{r}^{-2} \mathbf{J}_m) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}^2 + \mathbf{B}(\mathbf{q}) [\dot{\mathbf{q}} \dot{\mathbf{q}}] + \mathbf{r}^{-2} \mathbf{f} \dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{u}(\mathbf{t}) \quad (\text{IV.23})$$

Avec,

- $\mathbf{u}(\mathbf{t}) = \mathbf{K}_m \mathbf{r}^{-1} \mathbf{V}(\mathbf{t})$  : Vecteur entrée du système.
- $\mathbf{f} = \mathbf{B}_m + \mathbf{K}_b \mathbf{K}_m (\mathbf{R}_a)^{-1}$  : Matrice de frottement.
- $\mathbf{G}(\mathbf{q})$  : Vecteur généralisé de gravitation.
- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  : Matrice de Coriolis
- $\mathbf{r}$  : Matrice de réduction diagonale du système dont les éléments sont les inverses au carré des rapports de réduction des articulations.
- $\mathbf{r}^{-1}$  : Matrice inverse de  $\mathbf{r}$ .
- $\mathbf{r}^{-2}$  : Matrice carré inverse de  $\mathbf{r}$ .
- $\mathbf{J}_m$  : Matrice d'inertie diagonale dont les éléments sont les inerties des actionneurs de chaque articulation.
- $\mathbf{R}_a$  : Matrice diagonale des résistances des armatures.
- $\mathbf{R}_a^{-1}$  : Matrice inverse de  $\mathbf{R}_a$ .
- $\mathbf{K}_m$  : Matrice diagonale des constantes de couple.
- $\mathbf{K}_b$  : Matrice diagonale des constantes de force électromotrice.
- $\mathbf{B}_m$  : Matrice des frictions des actionneurs.

Sous une forme plus simplifiée, l'équation dynamique du système est donnée par:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{u} \quad (\text{IV.24})$$

Avec,

$\mathbf{M}(\mathbf{q}) = \mathbf{D}(\mathbf{q}) + \mathbf{r}^{-2} \mathbf{J}_m$  : Matrice d'inertie du système

$\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{r}^{-2}\mathbf{f}\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})$  : Somme des termes de Coriolis, Frottement et Gravitation.

Il en ressort des équations ci-dessous, le terme de réduction étant inférieur à l'unité, car la rotation côté moteur (actionneur) est toujours plus élevée que celle côté bras entraîné, que la dynamique de l'actionneur est prépondérante sur celle du robot, ce qui permet d'obtenir des équations dynamiques linéaires. Ce résultat n'est évidemment pas valable si l'entraînement de l'articulation est direct

# ***CHAPITRE V***

## ***COMMANDE LINEARISANTE A COUPLE CALCULE (COMPUTED TORQUE CONTROL)***

## V. Commande Linéarisante à couple calculé (Computed Torque control)

### V.1.Principe

La commande des robots manipulateurs constitue à l'heure actuelle l'une des préoccupations majeures des recherches en robotique. En effet la majorité des tâches confiées aux robots sont délicates et exigent une très grande précision sous des trajectoires rapides. Durant ces trois dernières décennies, afin d'améliorer les performances des manipulateurs, des recherches avancées ont permis de faire émerger de nouvelles techniques de commande appliquées aux robots manipulateurs, notamment celles basées sur la linéarisation et le découplage du système à commander [15]. Elle est connue sous le nom de commande linéarisante ou commande en couple calculé.

Les premiers travaux portant sur cette technique de commande ont donné lieu à la conception de la commande linéarisant à paramètres connus et constants. L'implémentation de cette loi de commande nécessite la connaissance précise et complète du modèle du robot. Dans une telle situation, cette commande est parfaite. Cependant, dans la pratique cette exigence est très difficile à satisfaire vu les perturbations agissant sur le robot. Dans de telles conditions cette technique de commande s'est avérée inefficace et reste inutilisable [16].

Notre travail porte sur l'influence des incertitudes sur le modèle sur la trajectoire du système. Pour cela nous comparons cette méthode aux méthodes classiques de commande PD avec compensation de la gravité et qui n'exigent pas une connaissance du modèle dynamique du système.

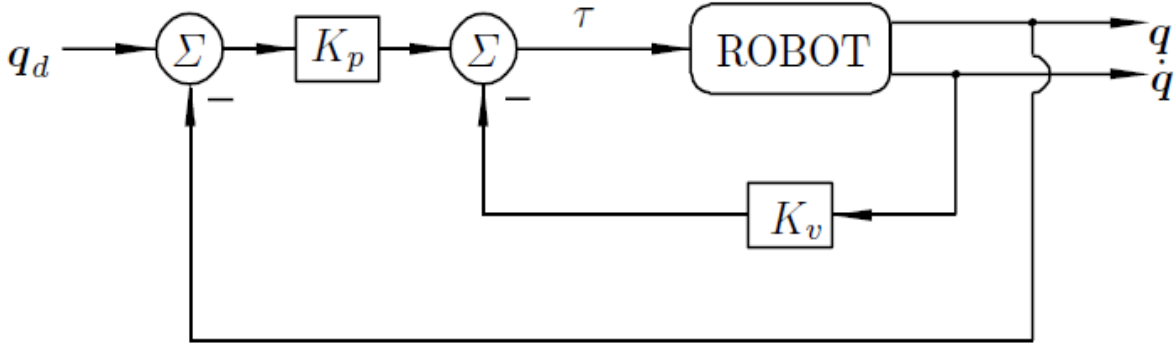
### V.2.Commandes classiques PD

#### V.2.1. Contrôle proportionnel avec rétroaction de vitesse

Le contrôle proportionnel muni d'une boucle de rétroaction vitesse est le contrôleur en boucle fermée le plus simple. Il peut être utilisé pour contrôler des robots manipulateurs. L'application conceptuelle de cette stratégie de contrôle la commande de position des moteurs à courant continu. Dans cette application, le dispositif de commande est également connu sous l'appellation de contrôle proportionnel avec retour tachymétrique. La loi de commande proportionnelle à rétroaction de vitesse rétroaction est donnée par :

$$\tau = K_p \tilde{q} - K_v \dot{q} \quad (\text{V.1.1})$$

Où  $K_p, K_v \in \mathbb{R}^{n \times n}$  sont des matrices symétriques définies positives présélectionnées par le praticien et sont communément appelé respectivement gain de position, et vitesse (ou dérivé). Le vecteur  $q_d \in \mathbb{R}^{n \times n}$  correspond à la position articulaire souhaitée et le vecteur  $\tilde{q} = q_d - q \in \mathbb{R}^{n \times n}$  est appelée erreur de position. La figure V.1 présente un schéma-bloc correspondant au système de commande formé par le robot sous contrôle proportionnel plus rétroaction de vitesse.



**Figure V.1.** Schéma-Bloc de la commande proportionnelle avec rétroaction de vitesse.

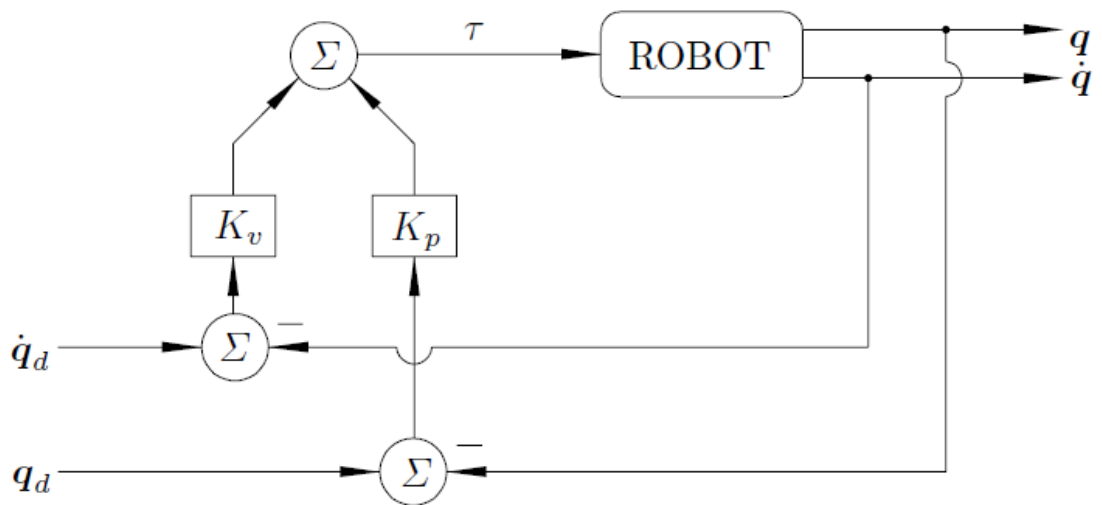
### V.2.2. Contrôle proportionnel-dérivé (PD)

La Commande proportionnelle-Dérivée (PD) est une extension immédiate de la commande proportionnelle avec retour de vitesse (V.1). Comme son nom l'indique, la loi de commande n'est pas seulement composée d'un terme proportionnel de l'erreur de position comme dans le cas d'une commande proportionnelle, mais aussi d'un autre terme qui est proportionnel à la dérivée de la position, i.e à l'erreur de vitesse  $\dot{\tilde{q}}$ .

La loi de commande PD est donnée par :

$$\tau = K_p \tilde{q} + K_v \dot{\tilde{q}} \quad (\text{V.1.2})$$

Où  $K_p, K_v \in \mathbb{R}^{n \times n}$  sont également symétriques définies positives et sélectionnées par le concepteur. Sur la figure 6.2, nous présentons le schéma-bloc correspondant au système de commande composé d'un régulateur PD et un robot.



**Figure V.2.** Schéma-Bloc de la commande PD

Jusqu'à présent, aucune restriction n'a été imposée sur le vecteur de position angulaire  $q_d$  souhaité pour définir les deux lois de commande précédentes. Ce est naturel, puisque le nom que nous



donnons à un contrôleur doit caractériser seulement sa structure et ne devrait pas être dépendant de la référence.

En dépit de la véracité de l'assertion ci-dessus, dans la littérature sur la commande des robots, on constate que les lois de contrôle (V,1) et (V,2) sont indistinctement appelés «contrôle PD ». L'argument commun en faveur de cette terminologie ambiguë est que dans le cas particulier où le vecteur de positions  $\mathbf{q}_d$  souhaitée est constant (consigne en échelon), alors il est clair de la définition de  $\tilde{\mathbf{q}}$  que  $\dot{\tilde{\mathbf{q}}} = -\dot{\mathbf{q}}$  et donc, les lois (V,1) et (V,2) deviennent identiques.

Dans le but d'éviter toute ambiguïté sur ces observations, et d'observer l'utilisation de la nomenclature commune à partir de maintenant, les deux lois de commande (V.1.1) et (V1.2), sont appelés dans ce qui suit par « **commande contrôle PD** ».

Dans les applications réelles, la régulation PD est locale en ce sens que le couple ou la force déterminée par un tel dispositif de commande lorsqu'il est appliqué à une articulation particulière, dépend seulement de la position et de la vitesse de l'articulation en question et non sur celles des autres articulations. Mathématiquement, cela se traduit par le choix d'une conception diagonale matrices  $\mathbf{K}_p$  et  $\mathbf{K}_v$ .

La commande Régulation PD, donnée par l'équation (V.1), nécessite la mesure de positions  $\mathbf{q}$  et les vitesses  $\dot{\mathbf{q}}$  ainsi que la spécification de la position articulaire  $\mathbf{q}_d$  souhaitée (cf. Figure V.1). Notez qu'il n'est pas nécessaire de préciser la vitesse souhaitée et l'accélération  $\ddot{\mathbf{q}}_d$ .

Dans ce qui suit, on présente une analyse de la régulation PD pour les manipulateurs de robots (à  $n$ -degrés de liberté). Le comportement d'un robot ( $n$ -degrés de liberté) en boucle fermée avec commande PD est déterminé en combinant le modèle de l'équation (V.3) à la loi de commande (V,1),

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} \quad (\text{V.1.3})$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{K}_p\tilde{\mathbf{q}} - \mathbf{K}_v\dot{\mathbf{q}} \quad (\text{V.14})$$

Ou de manière équivalente, en termes de vecteur d'état  $[\tilde{\mathbf{q}}^T \dot{\tilde{\mathbf{q}}}^T]^T$

$$\frac{d}{dt} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\mathbf{q}}} \\ \ddot{\mathbf{q}}_d - \mathbf{M}(\mathbf{q})^{-1}[\mathbf{K}_p\tilde{\mathbf{q}} - \mathbf{K}_v\dot{\mathbf{q}} - \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{G}(\mathbf{q})] \end{bmatrix} \quad (\text{V.1.5})$$

Qui est une équation non linéaire différentielle non autonome. Dans le reste de cette section, nous supposons que le vecteur des positions articulaires désirées  $\mathbf{q}_d$  est constant. Sous cette condition, l'équation en boucle fermée peut être réécrite en termes du nouveau vecteur d'état  $[\tilde{\mathbf{q}}^T \dot{\tilde{\mathbf{q}}}^T]^T$ , comme :

$$\frac{d}{dt} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} -\dot{\mathbf{q}} \\ \mathbf{M}(\mathbf{q})^{-1}[\mathbf{K}_p\tilde{\mathbf{q}} - \mathbf{K}_v\dot{\mathbf{q}} - \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{G}(\mathbf{q})] \end{bmatrix} \quad (\text{V.1.6})$$

Notez que l'équation différentielle en boucle fermée est encore non linéaire mais autonome. C'est parce que  $q_d$  est constante. L'équation précédente cependant, peut avoir des équilibres multiples. Si tel est le cas, ils sont donnés par  $[\tilde{q}^T \dot{\tilde{q}}^T]^T = [s^T \mathbf{0}^T]^T$  où  $s \in \mathbb{R}^n$  est solution de :

$$K_p s - G(q_d - s) = \mathbf{0} \quad (\text{V.1.7})$$

A l'équilibre, n régime établi, les termes de vitesse et d'accélération s'annulent ( $V(q, \dot{q})$  s'annule car il lui vient en produit le vecteur vitesse  $\dot{q}$ ).

Évidemment, si le modèle de manipulateur n'inclut pas les couples gravitationnels terme  $G(q)$ , le seul équilibre est l'origine de l'espace d'état, c'est à dire  $[\tilde{q}^T \dot{\tilde{q}}^T]^T = \mathbf{0} \in \mathbb{R}^{2n}$ . En outre, si  $G(q)$  est indépendant de  $q$ , c'est à dire si  $G(q) = g$  est constant, alors  $s = K_p^{-1} g$  est la seule solution.

Notez que l'équation (6.7) est en général non linéaire en  $s$  en raison du terme de la gravité  $G(q_d - s)$ . Pour cette raison, et étant donné la nature non linéaire de  $G(q_d - s)$ , la dérivation des solutions explicites de  $s$  est en général relativement complexe.

Dans les futures sections nous traitons séparément les cas dans lesquels le modèle de robot contient ou non le vecteur de couples gravitationnels  $G(q)$ .

### V.2.3. La gravité n'est pas prise en compte

Dans cette section, nous considérons robots dont le modèle dynamique ne contient pas le gravitationnelle  $G(q)$ ,

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = \tau \quad (\text{V.1.8})$$

Robots qui sont décrites par ce modèle sont ceux qui se déplacent uniquement sur le plan horizontal, ainsi que ceux qui sont mécaniquement conçu de moyen pratique spécifique.

En supposant que la position commune  $q_d$  souhaitée est constante. En boucle fermée, l'équation (V.6) devient (avec  $G(q) = \mathbf{0}$ ),

$$\frac{d}{dt} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} = \begin{bmatrix} -\dot{q} \\ M(q_d - \tilde{q})^{-1} [K_p \tilde{q} - K_v \dot{\tilde{q}} - C C(q_d - \tilde{q}, \dot{\tilde{q}}) \dot{\tilde{q}}] \end{bmatrix} \quad (\text{V.1.9})$$

$$\text{Avec : } V(q, \dot{q}) = C C(q_d - \tilde{q}, \dot{\tilde{q}}) \dot{\tilde{q}}$$

Qui, puisque  $q_d$  est constante, représente une équation différentielle autonome. De plus, l'origine  $[\tilde{q}^T \dot{\tilde{q}}^T]^T = \mathbf{0}$  est le seul équilibre de cette équation.

Considérons la fonction candidate de **Lyapunov** suivante :

$$P(\tilde{q}, \dot{\tilde{q}}) = \frac{1}{2} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix}^T \begin{bmatrix} K_p & \mathbf{0} \\ \mathbf{0} & M(q_d - \tilde{q}) \end{bmatrix} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} \quad (\text{V.1.10})$$

$$= \frac{1}{2} \dot{\tilde{q}}^T M(q) \dot{\tilde{q}} + \frac{1}{2} \tilde{q}^T K_p \tilde{q} \quad (\text{V.1.11})$$

Notons que cette fonction est définie positive puisque  $\mathbf{M}(\mathbf{q})$  ainsi que  $\mathbf{K}_p$  sont des matrices définies positives.

La dérivée totale de  $\mathbf{P}(\tilde{\mathbf{q}}, \dot{\mathbf{q}})$

$$\dot{\mathbf{P}}(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^T \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} + \tilde{\mathbf{q}}^T \mathbf{K}_p \dot{\tilde{\mathbf{q}}} \quad (\text{V.1.12})$$

En substituant  $\mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}$  de l'équation en boucle fermée (V.4), nous obtenons

$$\dot{\mathbf{P}}(\tilde{\mathbf{q}}, \dot{\mathbf{q}}) = -\dot{\mathbf{q}}^T \mathbf{K}_v \dot{\mathbf{q}} \quad (\text{V.1.13})$$

$$= - \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix}^T \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_v \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} \leq \mathbf{0}, \quad (\text{V.1.14})$$

Qui manifestement est une matrice définie semi-négative. La stabilité au sens de Lyapunov est donc vérifiée.

### V.3. Commandes par couple calculé

#### V.3.1. Loi de commande

Un robot manipulateur est généralement considéré comme un ensemble de corps rigides connectés en série par des articulations, avec une extrémité au sol, et l'autre libre (effecteur ou élément terminal).

Le modèle dynamique d'un tel mécanisme est habituellement décrit par l'équation matricielle suivante:

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) = \boldsymbol{\tau} \quad (\text{V.2.1})$$

Où :

$\boldsymbol{\tau}$  ( $n \times 1$ ): Vecteur des couples généralisés

$\mathbf{q}$  ( $n \times 1$ ): Vecteur des variables articulaires généralisées.

$\dot{\mathbf{q}}$  ( $n \times 1$ ): Vecteur des vitesses angulaires

$\ddot{\mathbf{q}}$  ( $n \times 1$ ): Vecteur des accélérations

$\mathbf{M}(\mathbf{q})$ : Matrice d'inertie de dimension ( $n \times n$ ) symétrique et définie positive

$\mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})$ : Vecteur de dimension ( $n \times 1$ ) spécifiant l'effet de centrifuge et de Coriolis

$\mathbf{G}(\mathbf{q})$ : Vecteur de dimension ( $n \times 1$ ) exprimant l'effet gravitationnel

$\mathbf{F}(\dot{\mathbf{q}})$ : Vecteur de dimension ( $n \times 1$ ) exprimant les frottements

$n$ : degré de liberté (nombre d'articulations). Dans la suite, on notera :

➤ Vecteur de l'erreur de position :  $\tilde{\mathbf{q}}(t) = \mathbf{q}_d(t) - \mathbf{q}(t)$

➤ Vecteur de l'erreur de vitesse :  $\dot{\tilde{\mathbf{q}}}(t) = \dot{\mathbf{q}}_d(t) - \dot{\mathbf{q}}(t)$

➤ Vecteur de l'erreur d'accélération :  $\ddot{\tilde{\mathbf{q}}}(t) = \ddot{\mathbf{q}}_d(t) - \ddot{\mathbf{q}}(t)$

Où

$q_d(t), \dot{q}_d(t)$  et  $\ddot{q}_d(t)$  représentent respectivement les vecteurs de position désirée, vitesse désirée et accélération désirée.

Nous allons maintenant introduire une loi de commande basée sur la linéarisation et le découplage du système régi par l'équation (V.1). La connaissance exacte du modèle de robot permet l'implémentation directe de la commande linéarisante à paramètres connus. La boucle de la linéarisation est réalisée par le choix d'un couple  $T$  à appliquer au robot, de la forme suivante [22] :

$$M(q)\bar{T} + V(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau \quad (\text{V.2.2})$$

En remplaçant  $\tau$  dans (V.2-1) on aura :

$$\ddot{q} = \bar{T} \quad (\text{V.2.3})$$

$M(q)$  Est une matrice non singulière, ceci transforme le problème de la commande du robot en celui de la commande de  $n$  doubles intégrateurs découplés. Ainsi l'entrée  $\bar{T}$  peut être interprétée comme une commande de la boucle externe qui devra asservir le système. Pour la commande en poursuite,  $\bar{T}$  est donné par :

$$\bar{T} = \ddot{q}_d + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t) \quad (\text{V.2.4})$$

Où

$K_v$  et  $K_p$  sont des matrices constantes, diagonales et positives de dimension  $(n \times n)$ .

$$K_v = \text{diag}\{K_{vi}\} = \text{diag}\{2\xi_i \omega_i\} \quad (\text{V.2.5})$$

$$K_p = \text{diag}\{K_{pi}\} = \text{diag}\{\omega_i^2\}; i=1 \dots n \quad (\text{V.2.6})$$

Avec :

$\xi$ : Facteur d'amortissement.

$\omega$ : Pulsation propre.

Les gains, donc les pulsations propres corrigées qui déterminent la dynamique (transitoires) du système sont limités par les flexibilités structurelles des liaisons, les retards purs introduits par les capteurs et les actionneurs et les périodes d'échantillonnage suivant les règles empiriques suivantes :

$$\omega_n \leq \frac{\omega_{res}}{2} \quad (\text{V.2.7})$$

$$\omega_n \leq \frac{\omega_{delay}}{3} \quad (\text{V.2.8})$$

$$\omega_n \leq \frac{\omega_s}{5} \quad (\text{V.2.9})$$

Le gain maximal est limité par la valeur minimale d'entre ces trois valeurs.

En remplaçant (V.2.4) dans (V.2.2), on obtient la loi de commande suivante :

$$\tau = M(q)[\ddot{q}_d + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t)] + V(q, \dot{q}) + G(q) + F(q, \dot{q}) \quad (\text{V.2.10})$$

L'équation de l'erreur traduisant le comportement du robot en boucle fermée se déduit en combinant l'équation (V.2.1) avec (V.2.7):

$$M(q)[\ddot{\tilde{q}}(t) + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t)] = 0 \quad (\text{V.2.11})$$

D'où

$$\ddot{\tilde{q}}(t) + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t) = 0 \quad (\text{V.2.12})$$

Ainsi, le comportement du robot bouclé avec cette loi de commande est identique à celui d'un système du second ordre. Cette dernière équation a pour solution un signal  $\tilde{q}(t)$  qui tend exponentiellement vers zéro. Le système en boucle fermée, avec cette loi de commande, dans le cas où le modèle du robot est connu avec exactitude, est asymptotiquement stable. Dans le cas d'une connaissance imprécise des paramètres du robot et/ou d'une présence de certaines dynamiques non modélisées, la commande linéarisante à paramètres constants montre ses limites. La solution adoptée consiste à utiliser un observateur de perturbations dont le rôle est d'estimer en ligne les incertitudes du modèle pour compenser leur influence dans la loi de commande.

En fait, toute sorte de perturbations intervient de façon non linéaire dans les équations du modèle. On doit alors introduire un terme caractérisant l'effet des incertitudes du modèle.

De ce fait, le modèle du robot est donné par :

$$\tau = M(q)\ddot{q} + \varphi(q, \dot{q}) + \Psi(t) \quad (\text{V.2.13})$$

Avec  $\varphi(\theta, \dot{\theta}) = V(\theta, \dot{\theta}) + G(\theta) + F(\dot{\theta})$

Où  $\Psi(t)$  est un vecteur ( $n \times 1$ ) spécifiant l'effet des incertitudes du modèle.

Dans ce qui suit, on suppose que  $M(T)$  et  $\varphi(\theta, \dot{\theta})$  sont connus alors que  $\Psi(t)$  est inconnu. Afin de compenser l'influence des termes non linéaires et les incertitudes dans l'équation (V.2.2), la loi de commande linéarisant devient :

$$\tau = M(\theta)\bar{T} + \varphi(\theta, \dot{\theta}) + \hat{\Psi}(t) \quad (\text{V.2.14})$$

Avec  $\bar{T}\ddot{q}_d + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t)$ , et  $\hat{\Psi}(t)$  représente l'estimée de  $\Psi(t)$ . En combinant (V.2.13) avec (V.2.14) on obtient l'équation de l'erreur dynamique en boucle fermée :

$$M(q)[\ddot{\tilde{q}}(t) + K_v \dot{\tilde{q}}(t) + K_p \tilde{q}(t)] = \Psi(t) - \hat{\Psi}(t) \quad (\text{V.2.15})$$

Ainsi le comportement du robot en boucle fermée est caractérisé par un choix approprié de  $K_v$  et  $K_p$ .

Le problème consiste alors à déterminer la fonction estimée  $\hat{\Psi}(t)$  dont l'écart :

$e(t) = \Psi(t) - \hat{\Psi}(t)$  Doit être minimal.

### V.3.2. Observateur de Perturbation

Un observateur est un système dynamique capable de reproduire une ou plusieurs grandeurs non mesurables d'un système à partir de la connaissance de ses entrées et ses sorties. Dans notre cas, l'observateur doit fournir à chaque instant une estimation de la fonction  $\Psi(t)$  spécifiant l'effet des perturbations agissant de façon non linéaire sur le modèle du robot manipulateur.

On définit l'erreur d'observation :  $e(t) = \Psi(t) - \hat{\Psi}(t)$

On désire réaliser un observateur de perturbations dont la dynamique de l'erreur d'observation sera décrite par l'équation :

$$\dot{e}(t) + R M^{-1}(q) e(t) = 0 \quad (\text{V.2.16})$$

Où ( $R$ ) est une matrice constante, diagonale et positive.

Sachant que la matrice d'inertie  $M(q)$  est définie positive son inverse l'est également, il est évident alors que la stabilité d'un tel observateur est assurée. En supposant que les variations des perturbations sont très faibles par rapport à la dynamique de l'observateur, on peut écrire :

$$\frac{dF(t)}{dt} \cong 0 \quad (\text{V.2.17})$$

En combinant (V.2.15) avec (V.2.16), on aura :

$$\frac{d\hat{F}(t)}{dt} = R M^{-1}(\theta) e(t) \quad (\text{V.2.18})$$

D'après les équations (VI-10) et (VI-13), l'estimation de la fonction  $F(t)$  est donnée par :

$$\frac{d\hat{F}(t)}{dt} = R \ddot{\tilde{q}}(t) + R K_v \dot{\tilde{q}}(t) + R K_p \tilde{q}(t) \quad (\text{V.2.19})$$

L'équation (V-14) représente la loi d'observation des perturbations liée à la commande linéarisante.

### V.3.3. Choix des paramètres PD

Les consignes de commande peuvent être soit du type consigne ou trajectoire dans l'espace articulaire. On peut cependant commander le système dans l'espace opérationnel comme, il apparaît dans le circuit ci-dessous. Le calcul de la géométrie et du jacobien inverse en amont des actionneurs dans la boucle de commande, la vitesse étant asservie, sont nécessaires :

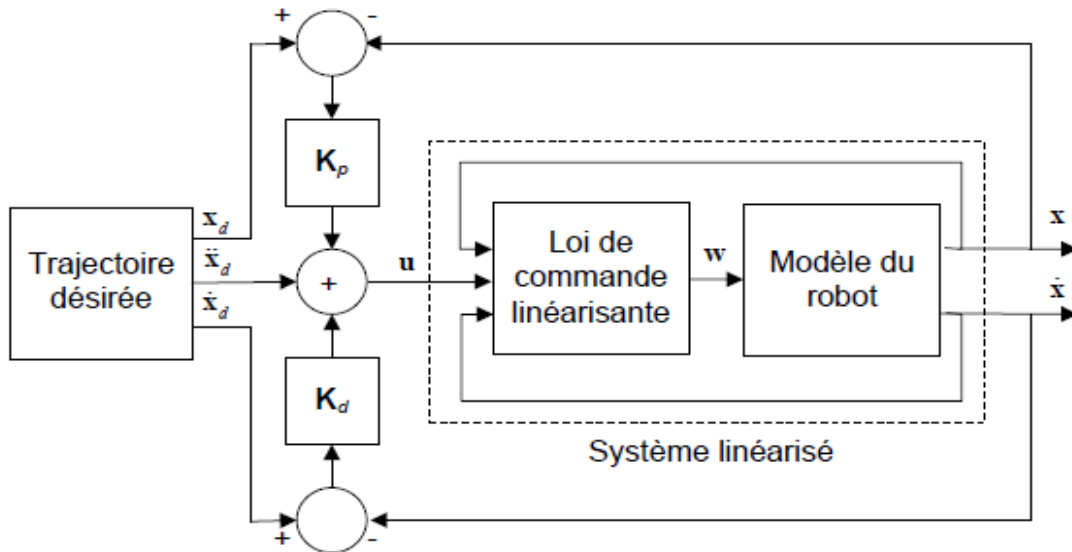


Figure V.3. Commande linéarisante PD dans l'espace cartésien

$$\ddot{\tilde{x}}_i + k_{di}\dot{\tilde{x}}_i + k_{pi}\tilde{x}_i = 0 \quad , i = 1, \dots, n \quad (\text{V.2.20})$$

Les gains de l'articulation peuvent alors être calculés de façon à imposer les pôles. Les gains suivants caractérisent par exemple un dépassement nul (régime critique) avec un temps de réponse  $T_r$ :

$$k_{pi} = \lambda^2 k_{di} = 2\lambda \quad (\text{V.2.21})$$

Où  $\lambda$  est un pôle de multiplicité 2 donné par :

$$\lambda = \frac{4.73}{T_r} \quad (\text{V.2.23})$$

#### V.4. Commande PID. Réjection des perturbations

##### V.4.1. Commande PID

Le système en boucle fermée avec correcteur PID prend la structure fonctionnelle montrée ci-dessous :

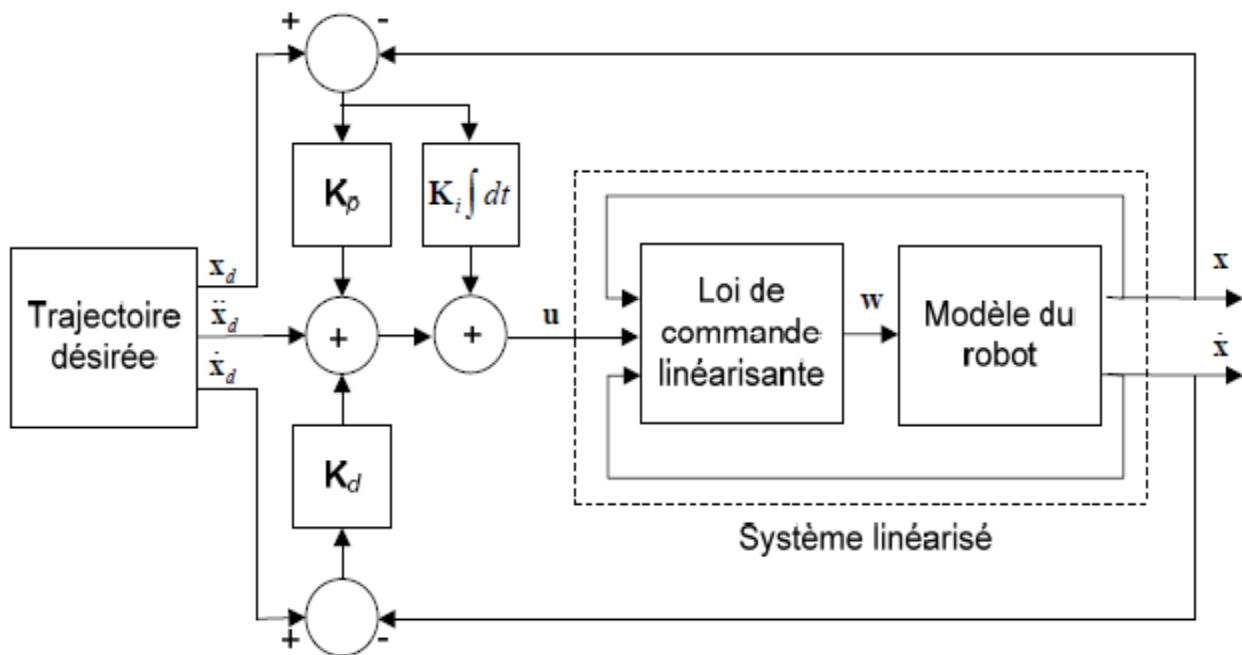


Figure V.4 : Commande linéarisante PID dans l'espace cartésien

Système linéarisé :

$$\ddot{\tilde{\mathbf{X}}} = \mathbf{U} \quad (\text{V.2.24})$$

Loi de commande PID :

$$\mathbf{U} = \ddot{\tilde{\mathbf{X}}}_d + \mathbf{K}_p \tilde{\mathbf{X}} + \mathbf{K}_i \int_0^t \tilde{\mathbf{X}} dt + \mathbf{K}_d \dot{\tilde{\mathbf{X}}} \quad (\text{V.2.25})$$

Dynamique de l'erreur de suivi :

$$\ddot{\tilde{\mathbf{X}}} + \mathbf{K}_d \dot{\tilde{\mathbf{X}}} + \mathbf{K}_p \tilde{\mathbf{X}} + \mathbf{K}_i \tilde{\mathbf{X}} = \mathbf{0} \quad (\text{V.2.26})$$

Si les matrices de gain  $\mathbf{K}_p$ ,  $\mathbf{K}_i$  et  $\mathbf{K}_d$  sont diagonales la dynamique de l'erreur devient:

$$\ddot{\tilde{x}}_i + k_{di} \dot{\tilde{x}}_i + k_{pi} \tilde{x}_i + k_{ii} \tilde{x}_i = 0, \quad i = 1, \dots, n \quad (\text{V.2.27})$$

Les gains de l'articulation  $i$  peuvent alors être calculés de façon à imposer les pôles. Les gains suivants caractérisent par exemple un dépassement nul avec un temps de réponse  $T_r$ :

$$k_{pi} = 3 \lambda^2 k_{ii} = \lambda^3 k_{di} = 3 \lambda \quad (\text{V.2.28})$$



Où  $\lambda$  est un pôle de multiplicité **3** donné par :

$$\lambda = \frac{6.73}{T_r} \quad (\text{V.2.29})$$

Dans notre travail, on s'est intéressé uniquement à la commande dans l'espace articulaire. On peut facilement démontrer que dans le cas d'un couple perturbateur  $\tau_d$ , provenant soit d'une incertitude sur le modèle, soit sur une force externe appliquée sur l'effecteur, l'erreur résultant stationnaire est égale à :

$$\tilde{q} = K_p^{-1} \tau_d \quad (\text{V.2.30})$$

Et s'annule donc le cas où une commande PID est adoptée.

***CHAPITRE VI***  
***FENETRE GRAPHIQUE MATLAB ET***  
***SIMULATION***

## VI.1 Description De La Fenêtre

### VI.1.1 Aperçu de la Fenêtre

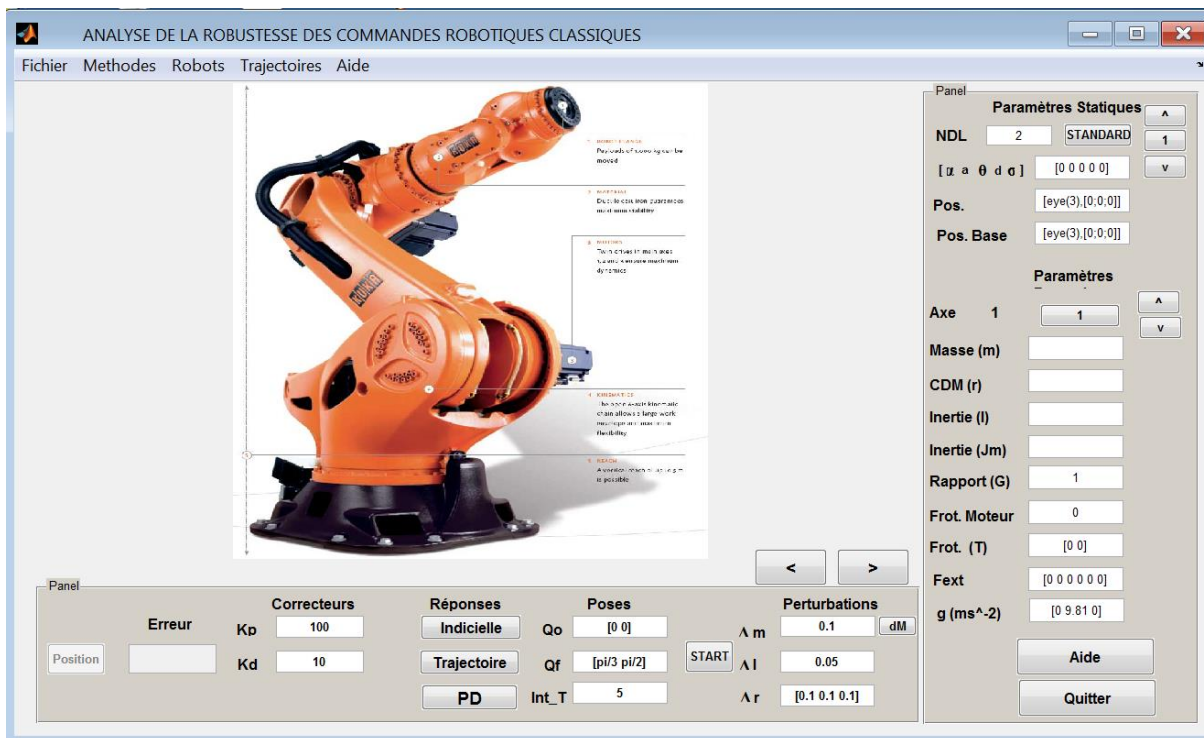


Figure 6.1

Ce logiciel écrit en code Matlab (Version 7.13) est constitué de deux modules : un programme de création de la fenêtre graphique " **maslek.fig** " et le programme principal " **maslek.m** " qui rassemble les différentes méthodes de commande des robots, le tracé des graphes des trajectoires et le calcul des erreurs pour des perturbations données sur le modèle dynamique du robot. La fenêtre graphique se présente sous la forme d'une interface interactive dynamique constituée d'une barre de menus et d'objets nécessaires à l'introduction des paramètres du robot, du type de correcteur, ses valeurs et les paramètres de la simulation. L'exécution sous la fenêtre de commande Matlab de la commande **belabbes** fait apparaître la fenêtre graphique suivante :

#### IV.1.2 Barre des menus

La barre des menus comprend les rubriques suivantes :

##### VI.1.2.1 Menu Fichier :

Il permet la création d'une session de simulation, correspondant à l'introduction d'un nouveau robot dont on introduira les paramètres statiques et dynamiques (**Nouveau**), la sauvegarde des résultats d'une session (**Enregistrer**), dans un fichier portant l'extension (\*.dat) dont le nom sera introduit par l'utilisateur et qui comportera toutes les informations concernant la simulation en cours. A savoir l'heure et la date de création du fichier, son nom, les paramètres DH définissant le robot ; le type de méthode de commande, les valeurs des paramètres du contrôleur ; les paramètres de

fonctionnement, ainsi que les valeurs des erreurs sur les trajectoires par rapport au cas non perturbé. C'est ce fichier qui permettra de rappeler une session de simulation précédente (**Ouvrir**). L'utilisateur peut également changer le nom du fichier de sauvegarde (**Enregistrer sous**). Enfin la commande (**Quitter**) ferme la fenêtre et supprime les variables créées pendant la session.

#### **VI.1.2.2 Menu Méthodes**

Il rassemble les méthodes de commande les plus connues utilisant un correcteur PD ou PID, à savoir la méthode linéarisante à couple calculé et la méthode non linéaire sous les deux versions : celles utilisant le calcul du modèle exact en temps réel, et celle dite par anticipation (feedforward) qui utilise les paramètres de la trajectoire de consigne pour le calcul du modèle dynamique du robot. Les consignes peuvent être soit du type échelon ; soit du type trajectoire (commande en poursuite)

#### **VI.1.2.3 Menu Robots**

Les robots les plus connus peuvent être visualisés dans la fenêtre. Cette action permet de charger le modèle DH du robot, dans le but d'en faire une simulation.

#### **VI.1.2.4 Menu Trajectoires**

Dans ce menu, on peut sélectionner le type de trajectoires à visualiser dans l'espace opérationnel ou articulaire. La position, la vitesse et l'accélération de la pince terminale (trajectoire opérationnelle dans l'espace cartésien) ou des variables d'articulations peuvent être ainsi visualisées.

#### **VI.1.2.5 Menu Aide**

La validation de ce menu donne accès à l'aide générale sur le logiciel et son utilisation. Tous les menus sont munis de raccourcis claviers (lettre soulignée accessible par la commande Alt sélection) et d'accélérateurs qui permettent d'exécuter directement la commande en utilisant la commande Ctrl +lettre sélectionnée.

### **VI.1.3 Barre des objets**

La barre des objets comprend les composants suivants : Un cadre comprenant les paramètres d'entrée des valeurs des paramètres DH statiques et dynamiques du robot étudié, le type de convention utilisé ( standard ou modifié), les coordonnées de la base et de l'outil (effecteur) relativement à la dernière articulation, la valeur des forces et moments appliqués à l'effecteur, Deux boutons poussoirs correspondant respectivement à l'aide sur la fenêtre et sa fermeture.

Un second cadre rassemble les valeurs des perturbations sur le modèle dynamique du robot, les valeurs initiales et finales des consignes, son type (indicielle ou trajectoire), la durée de la trajectoire, des éditeurs pour introduire les valeurs adoptées des paramètres du correcteur, un bouton poussoir de sélection sur son type (PD ou PID), deux poussoirs pour sélectionner le type d'entrée de commande (consigne indicielle ou trajectoire polynomiale). Enfin une fenêtre d'édition de l'erreur

globale par rapport aux consignes indicielles ou les trajectoires désirées, avec un bouton poussoir permettant de sélectionner le type de graphe, c'est à dire la position, la vitesse ou l'accélération.

#### VI.1.4 Déroulement d'une simulation

L'utilisateur devra suivre les étapes suivantes :

1. Ouvrir un fichier d'une session précédente, les éditeurs du cadre d'entrée des paramètres du robot sont alors remplis automatiquement.
2. Entrer les valeurs des positions initiales et finales et le type d'entrée (consigne ou trajectoire)
3. Choisir une méthode de commande et un type de correcteur (PD ou PID). Les éditeurs correspondants s'affichent ou sont masqués (Action intégrale I, dans le cas d'une sélection d'une action PD)
4. Introduire les valeurs des paramètres du correcteur, en s'aidant des règles établies précédemment.
5. Introduire les valeurs des incertitudes sur le modèle : position du centre de gravité d'une liaison, sa masse ou éventuellement sa matrice d'inertie. Si une seule valeur est introduite, alors cette valeur est adoptée pour l'ensemble des liaisons restantes.
6. Lancer la simulation avec le bouton d'exécution **START**. Une fenêtre de dialogue s'affiche indiquant que l'exécution est en cours. Le programme adresse alors le schéma simulink concerné et affiche une fenêtre de dialogue indiquant que le calcul est en cours. Une fois le calcul terminé, les graphes des trajectoires articulaires s'affichent sur la fenêtre. L'utilisateur pourra alors à sa convenance sélectionner le tracé des autres graphes pour les espaces : articulaire et cartésien
7. Sauvegarder le fichier en cours, sous un nom donné. Une variable portant le même nom que le fichier sera dans le même temps sauvegardé dans la perspective de l'ouverture d'une session. Cette variable sauvegardera la variable structurée « **handles** » qui rassemble toutes les variables créés durant la session, à savoir la date et l'heure de création du fichier, les paramètres du modèle DH, les paramètres dynamiques de chaque liaison, les valeurs des gains du correcteur, la méthode de correction, et la valeur trouvée pour les erreurs de position, vitesse et accélération
8. Dans le cas où on désire analyser un nouveau robot, on doit introduire dans l'ordre son nombre de liaisons, ses paramètres statiques, puis dynamiques et suivre les étapes énumérées ci-haut de 2 à 7.

## VI.2 Simulation

Pour pouvoir comparer avec la méthode à couple calculé, nous avons analysé au préalable étudié la commande classique PD (PID) avec compensation. Pour pouvoir visualiser les trois graphes correspondants aux trois trajectoires de position dans le cas d'une perturbation sur la masse, nous avons volontairement choisi des perturbations de grande amplitude pour pouvoir discriminer les trois courbes ( $m + \Delta m, m, m - \Delta m$ ). Le choix des paramètres a été fait pour avoir une sortie à dépassement nul (régime critique).  $K_p=100$   $K_d=10$ . La simulation du Puma560 exigeant des temps d'exécution assez élevés. Nous avons dans un premier temps étudié la robustesse de la commande sur le Robot plan à 2 ddl Pélikan.

Les circuits de simulation sont visualisés ci-dessous. Pour reconstruire le modèle du vecteur de gravité, nous avons implanté le bloc dynamique inverse qui utilise l'algorithme de Newton-Euler (fonction `rne` de la boîte à outil). Pour accélérer la vitesse d'exécution, deux versions de la commande étudiées : celle qui utilise la commande en temps réel du modèle et celle qui utilise les valeurs  $q_d$  de la position. (anticipation). Le schéma diffère suivant qu'on est en mode consigne (échelon) ou trajectoire.

### VI.2.1 Commande PD avec compensation de la gravité

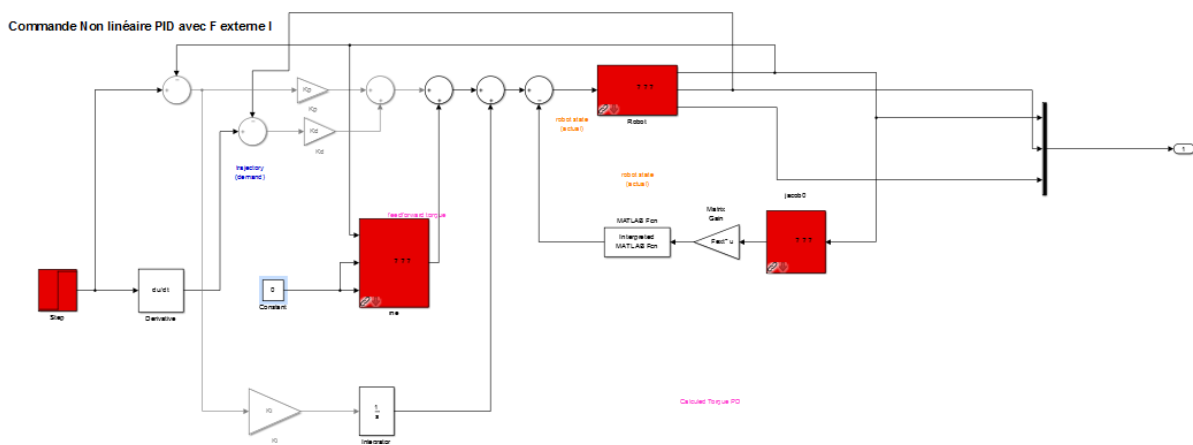
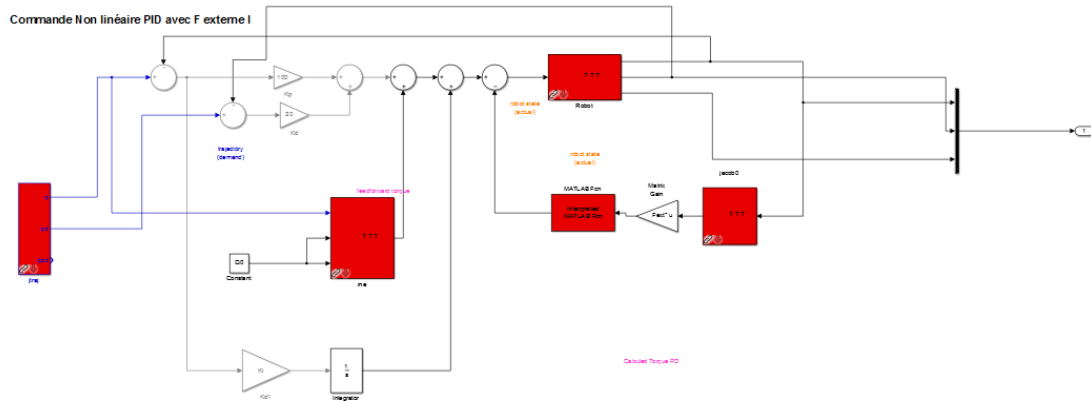
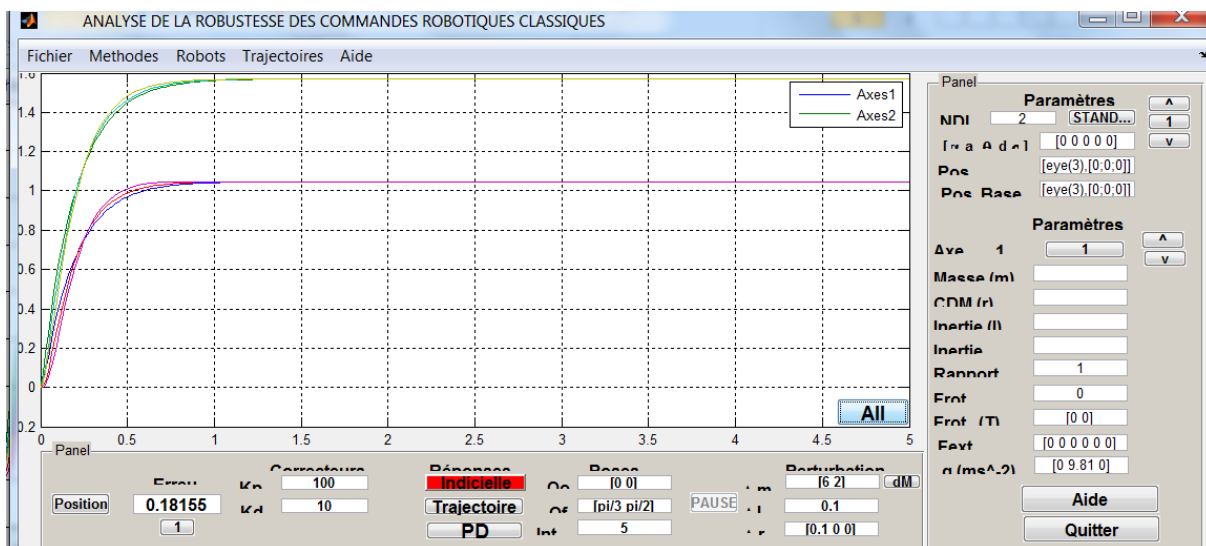


Figure VI.1 : Schéma de simulation de la commande PD avec compensation de la gravité



**Figure VI.2 :** Schéma de simulation de la commande PD avec compensation de la gravité  
Avec utilisation des valeurs de consignes (anticipation)



**Figure VI.3 :** Graphes temporels de l'évolution de la position articulaire  
pour les consignes  $[\pi/3 \ \pi/2]$  (Commande Non linéaire PD)

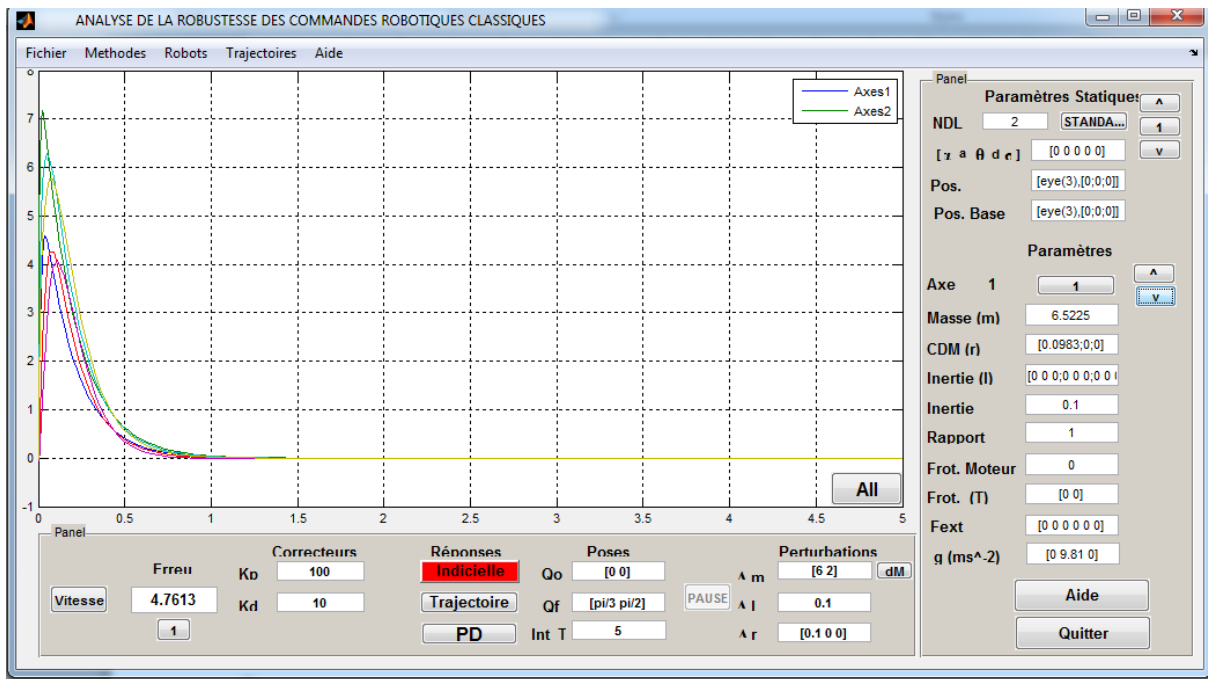


Figure VI.4 : Graphes temporelles de l'évolution de la vitesse  
(Commande Non linéaire PD)

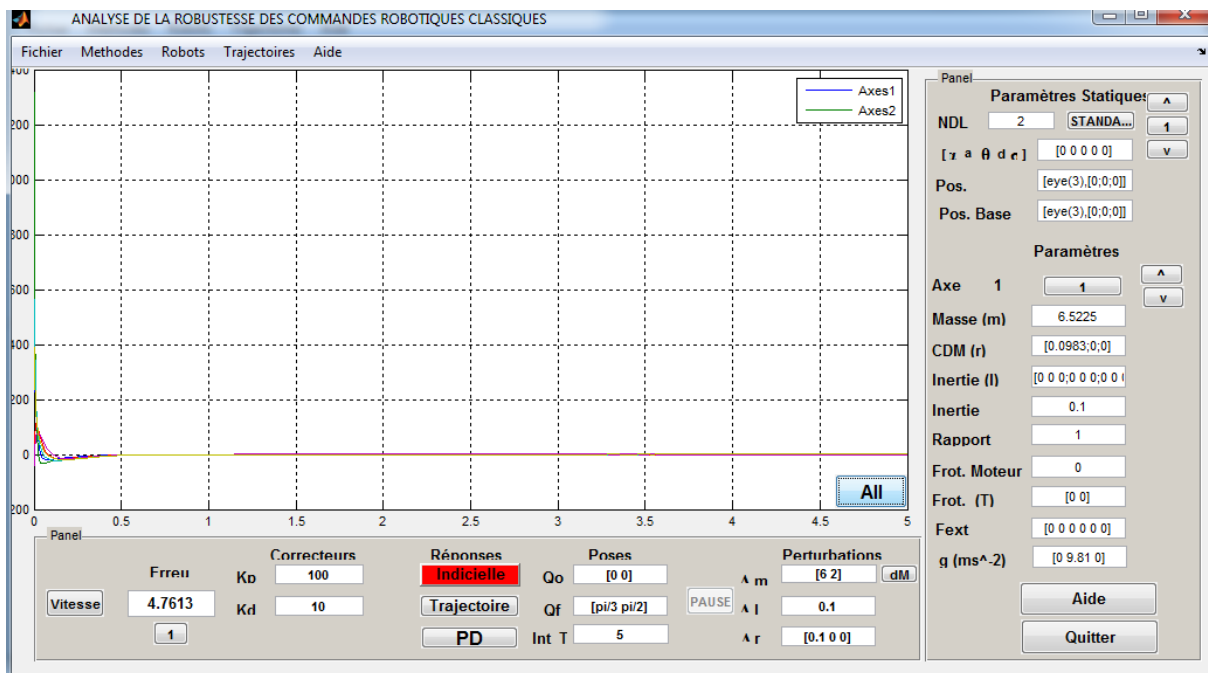


Figure VI.5 : Graphes temporelles de l'évolution de l'accélération  
(Commande Non linéaire PD)



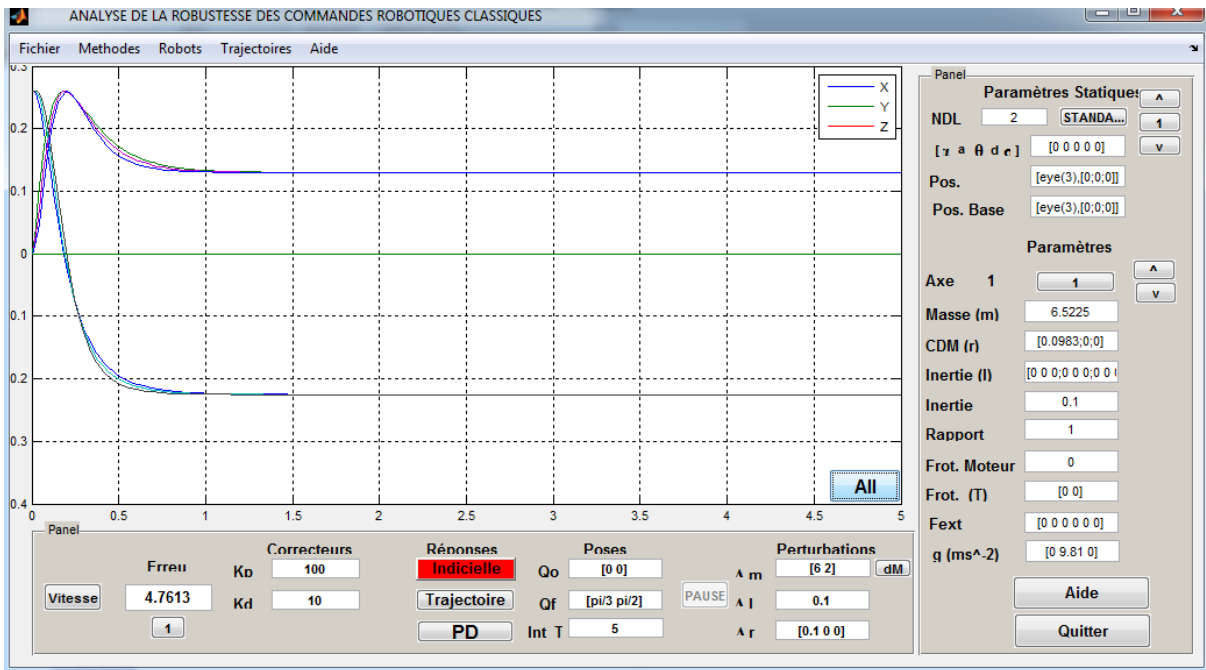


Figure VI.6 : Trajectoires cartésiennes position de l'effecteur pour les consignes [pi/3 pi/2]  
(Commande Non linéaire PD)

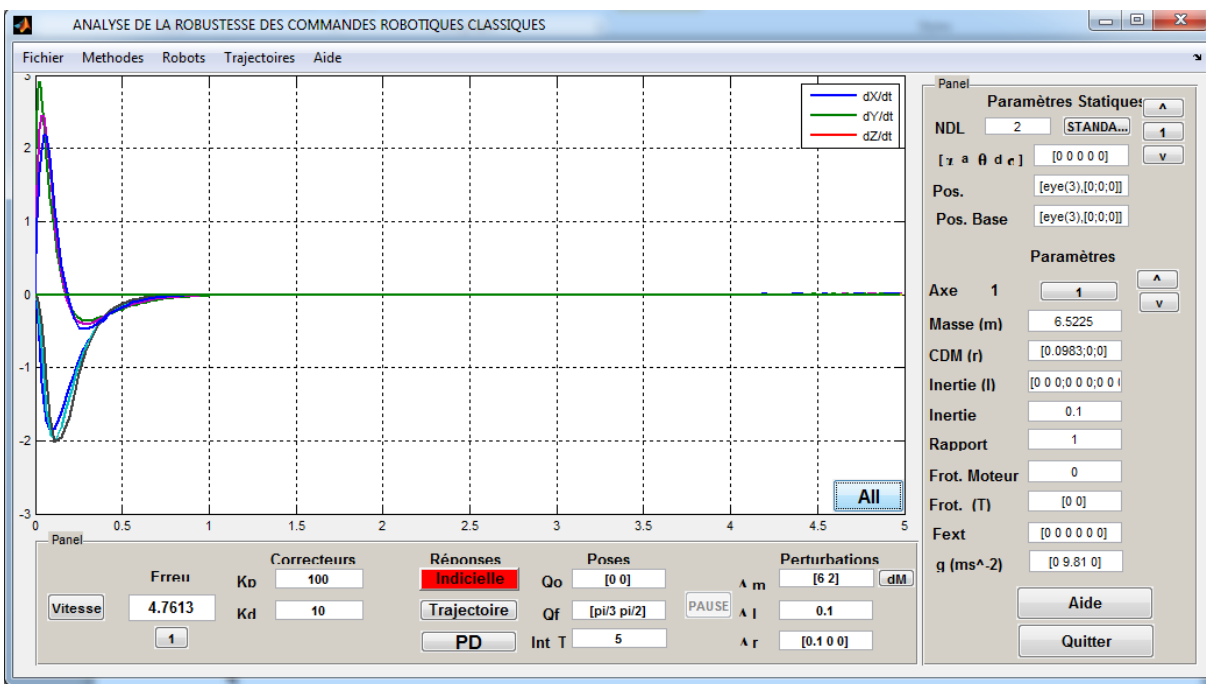


Figure VI.7 : Trajectoires cartésiennes vitesse  
(Commande Non linéaire PD)

Dans les commandes en trajectoires, on ne peut plus discerner les courbes, car l'erreur devient négligeable. On résumera dans le tableau récapitulatif la valeur des erreurs, pour les différentes configurations.

## VI.2.2 Commande à couple calculé

Dans cette structure de commande, l'ensemble du modèle est requis pour pouvoir élaborer la loi de commande. La présence du bloqueur d'ordre zéro est motivée pour stabiliser l'algorithme de résolution. Sa valeur est fonction de la valeur des gains adoptée. Nous avons adopté dans notre cas compte tenu des valeurs des gains une valeur de 0.05s. Les deux variantes sont étudiées : celles qui calculent le modèle en temps réel et celle qui calcule les matrices du modèle dynamique à partir des consignes de la trajectoire (anticipation)

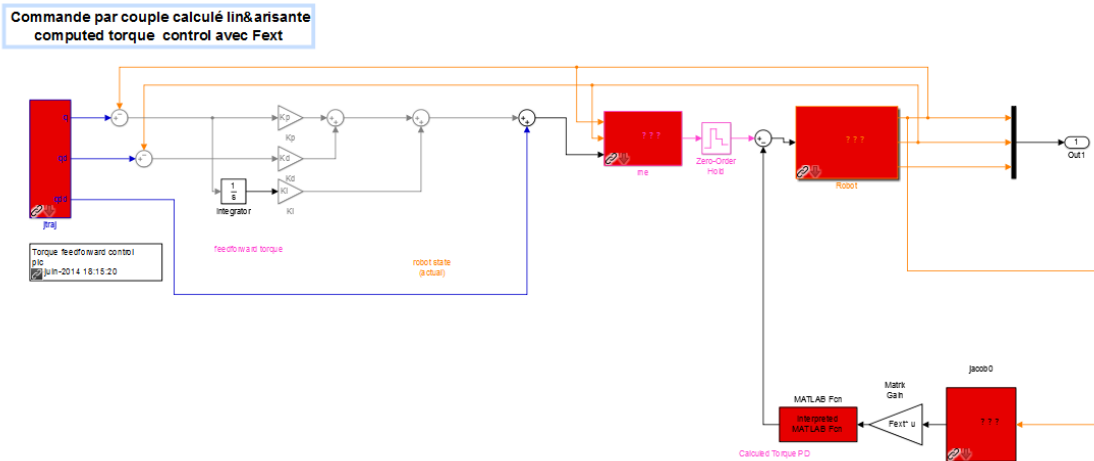


Figure VI.8 : Schéma de simulation de la commande PD à couple calculé

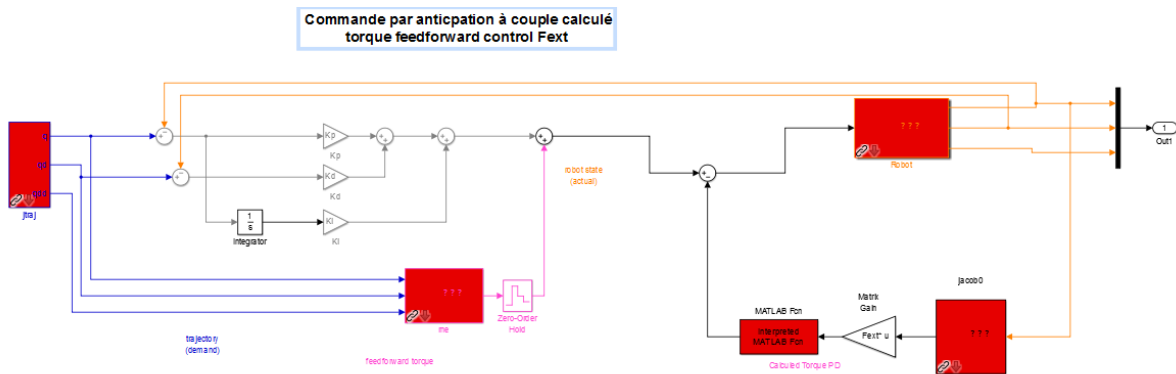


Figure VI.9 : Schéma de simulation de la commande PD à couple calculé avec anticipation

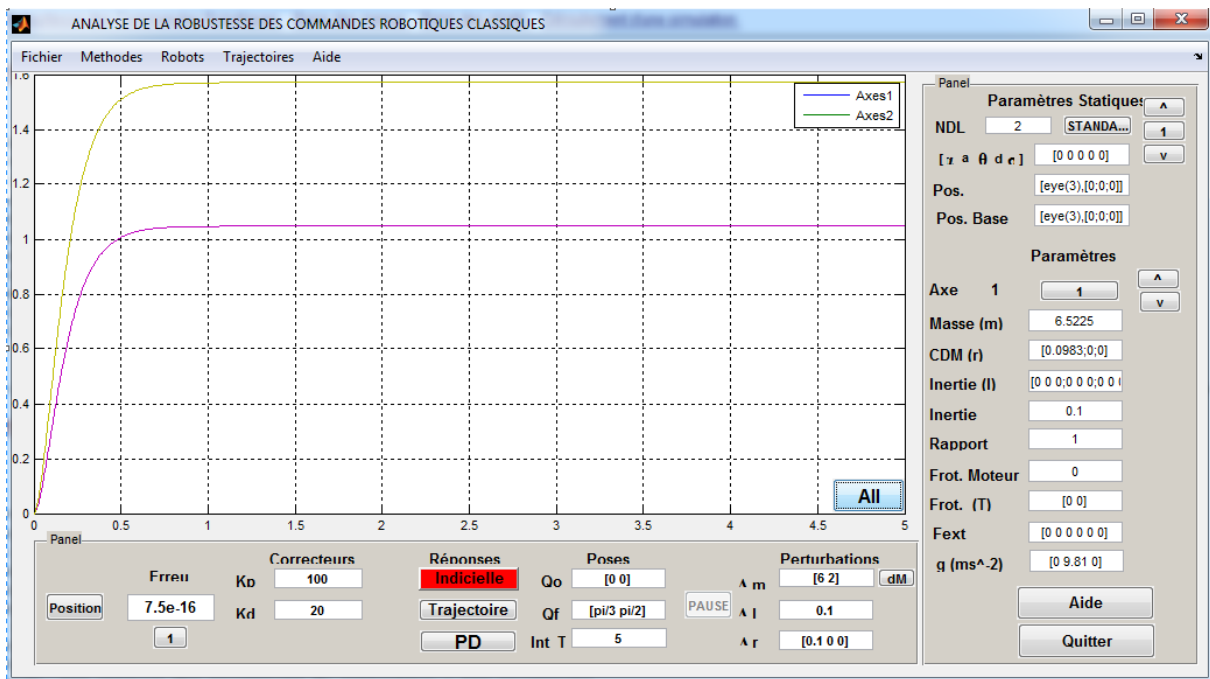


Figure VI.10 : Graphes temporelles de l'évolution de la position articulaire pour les consignes [pi/3 pi/2] Commande Couple Calculé

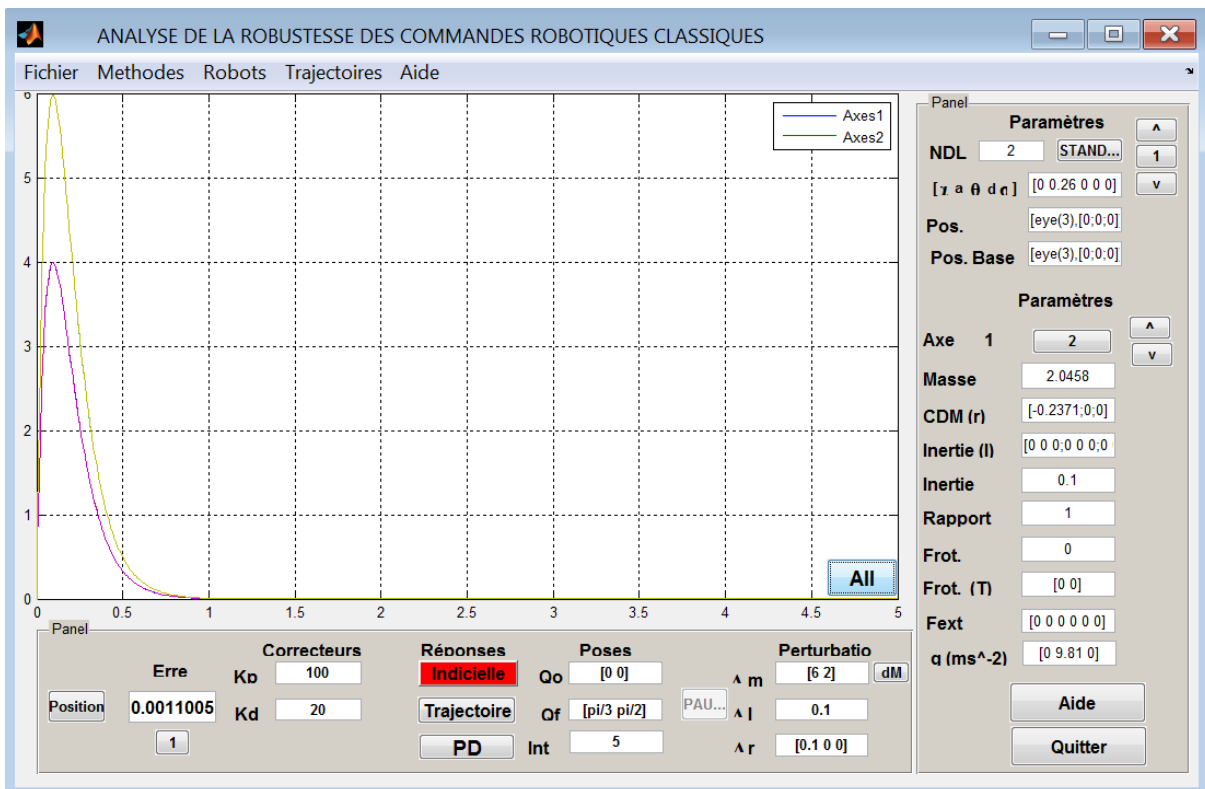


Figure VI.11 : Graphes temporelles de l'évolution vitesse

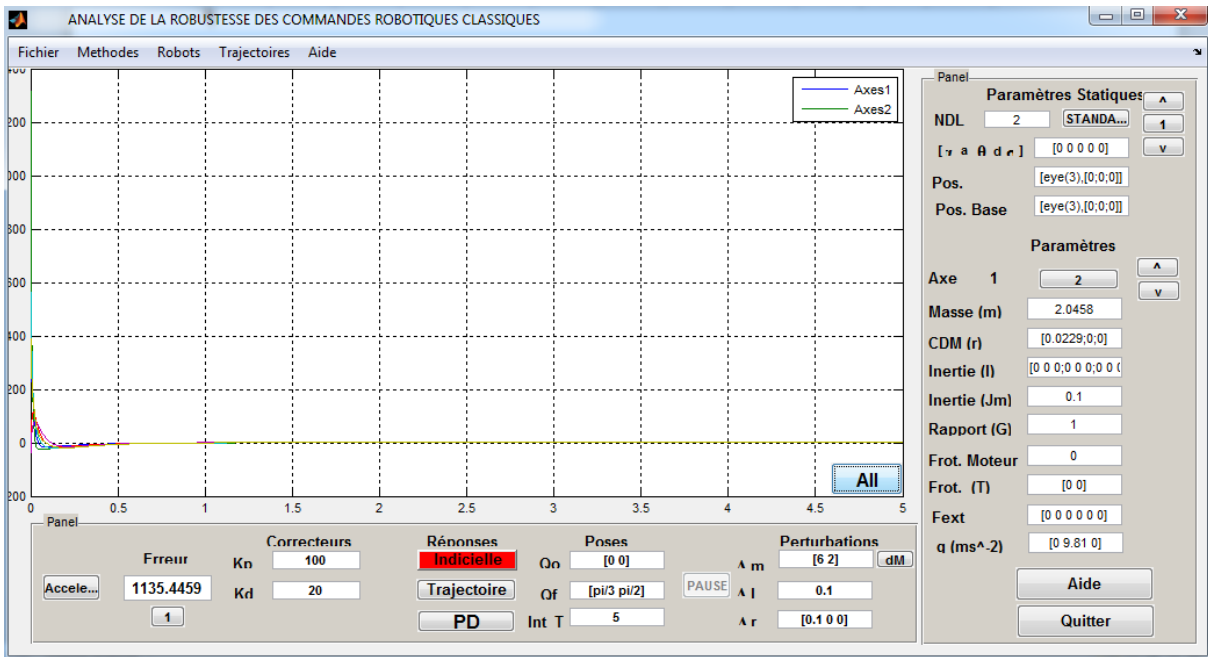


Figure VI.12 : Graphes temporels de l'évolution de l'accélération

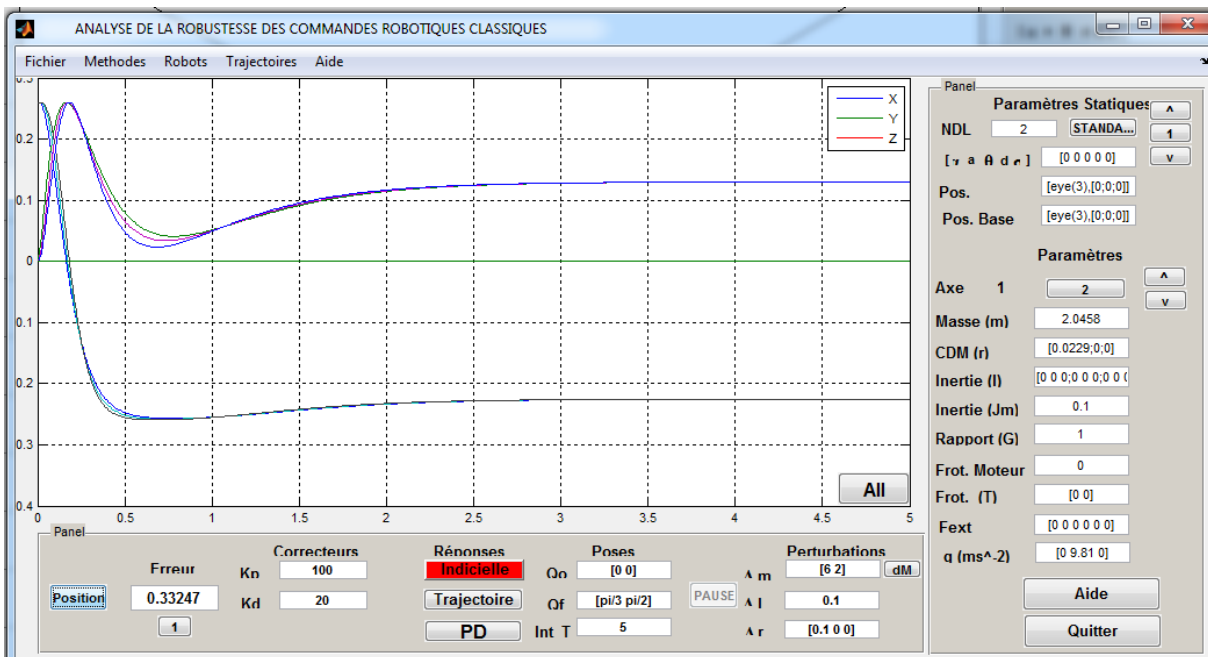


Figure VI.13 : Trajectoire cartésienne de l'effecteur

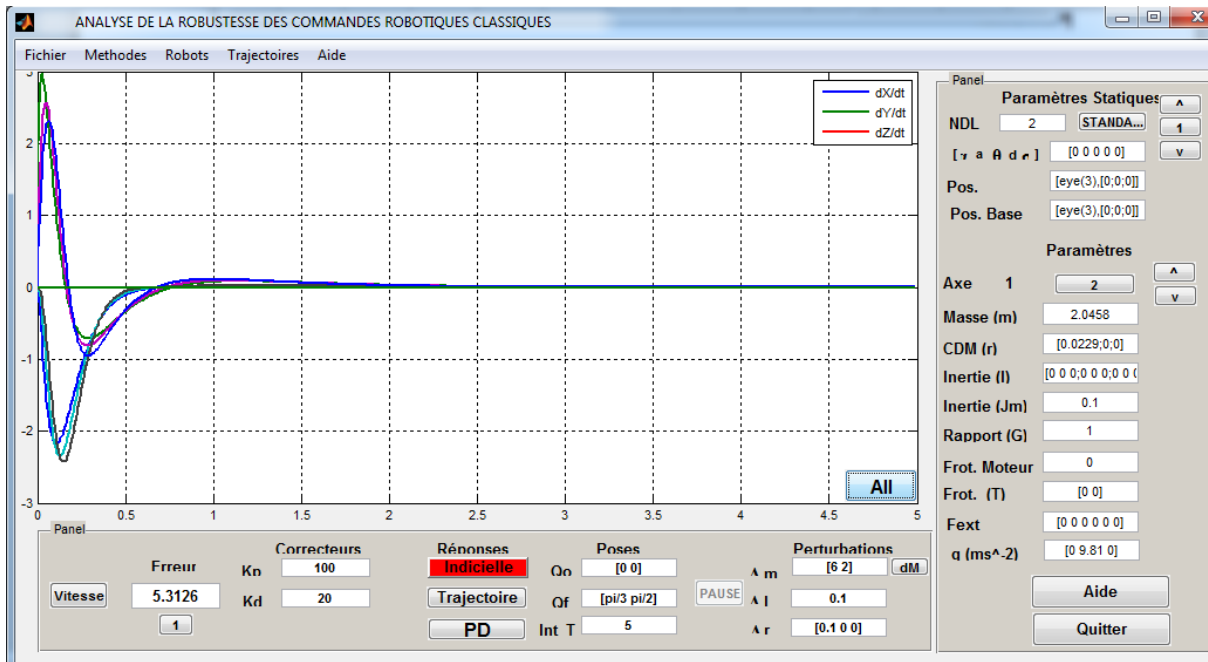


Figure VI.14 : Trajectoire vitesse cartésienne

### VI.2.3 Table récapitulative de l'évolution des erreurs en fonction de la nature et de l'amplitude de la perturbation pour le robot Pelikan

L'erreur qui s'affiche sur la fenêtre graphique correspond à la somme quadratique des écarts entre la trajectoire nominale non perturbée et la trajectoire perturbée la plus éloignée.

#### 1. Commande PD

##### 1.1. commande consignes PD

$K_p=100, K_v=10, K_a=20$

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_M$			$\Delta_M$			$\Delta_M$		
	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$
NL_PD	0.1	$5.31 \times 10^{-2}$	$2.53 \times 10^{-2}$	2.73	1.21	0.57	725	253	109
NL_PD ff	0.1	$5.31 \times 10^{-2}$	$2.53 \times 10^{-2}$	2.73	1.21	0.57	725	253	109
CC_PD	$1.10 \times 10^{-3}$	$3.85 \times 10^{-4}$	$1.8 \times 10^{-4}$	$1.10 \times 10^{-2}$	$4.15 \times 10^{-3}$	$1.94 \times 10^{-3}$	0.25	$9.15 \times 10^{-2}$	$4.28 \times 10^{-2}$
CC_PD ff	0.13	0.062	$3.44 \times 10^{-2}$	3.19	1.52	0.71	727	254	110

**Remarque :**  $K_v=10$  pour NL\_PD, NL\_PD ff

$K_v=20$  pour CC\_PD, CC\_PD ff

$K_p=100, K_v=10$

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_R$			$\Delta_R$			$\Delta_R$		
	0.15	0.1	0.05	0.15	0.1	0.05	0.15	0.1	0.05
NL_PD	0.28	0.1774	0.0813	4.98	3.4314	1.71	1017.4	914.21	482.0701
NL_PD ff	0.28	0.1774	0.0813	4.98	3.4314	1.71	1017.4	914.21	482.0701
CC_PD	0.04516 5	0.0304	0.01407	0.5642	0.37689	0.1710	7.2054	4.8334	2.2091
CC_PD ff	1.0247	0.61541	0.26936	13.0079	8.8164	4.274	1040.8	930.67	496.2001

**1.2. commande trajectoires PD**

**Kp=100,Kv=10,**

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_M$			$\Delta_M$			$\Delta_M$		
	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$
NL_PD	$6.15 \times 10^{-3}$	$3.2 \times 10^{-3}$	$1.5 \times 10^{-3}$	$8 \times 10^{-3}$	$4 \times 10^{-3}$	$2 \times 10^{-3}$	$5.4 \times 10^{-2}$	$4.38 \times 10^{-2}$	$3.2 \times 10^{-2}$
NL_PD ff	$6.15 \times 10^{-3}$	$3.2 \times 10^{-3}$	$1.5 \times 10^{-3}$	$8 \times 10^{-3}$	$4 \times 10^{-3}$	$2 \times 10^{-3}$	$5.4 \times 10^{-2}$	$4.38 \times 10^{-2}$	$3.2 \times 10^{-2}$
CC_PD	$5.58 \times 10^{-5}$	$2.82 \times 10^{-5}$	$1.41 \times 10^{-5}$	$5.47 \times 10^{-4}$	$2.04 \times 10^{-4}$	$9.71 \times 10^{-5}$	0.078	$2.8 \times 10^{-2}$	$1.22 \times 10^{-2}$
CC_PD ff	$5.58 \times 10^{-5}$	$2.82 \times 10^{-5}$	$1.41 \times 10^{-5}$	$5.47 \times 10^{-4}$	$2.04 \times 10^{-4}$	$9.71 \times 10^{-5}$	0.078	$2.8 \times 10^{-2}$	$1.22 \times 10^{-2}$

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_R$			$\Delta_R$			$\Delta_R$		
	0.15	0.1	0.05	0.15	0.1	0.05	0.15	0.1	0.05
NL_PD	0.0178	0.01106	0.00497	0.0229	0.0138	0.0063	0.0937	0.0652	0.054805
NL_PD ff	0.0178	0.01106	0.00497	0.0229	0.0138	0.0063	0.0937	0.0652	0.054805
CC_PD	$1.82 \times 10^{-4}$	$1.0645 \times 10^{-4}$	0,46066	$7.48 \times 10^{-4}$	$5.0942 \times 10^{-4}$	$2.71 \times 10^{-4}$	0.0784	0.0695	0.038545
CC_PD ff	$1.82 \times 10^{-4}$	$1.0645 \times 10^{-4}$	0,46074	$7.48 \times 10^{-4}$	$5.0942 \times 10^{-4}$	$2.71 \times 10^{-4}$	0.0784	0.0695	0.038527

## 2. Commande PID

### 2.1. commande consignes PID

**Kp=100, Kv=10, Ki=125**

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_M$			$\Delta_M$			$\Delta_M$		
	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$
NL_PID	<b>0.124</b>	<b>0.05</b>	<b>0.03</b>	<b>2.80</b>	<b>1.25</b>	<b>0.594</b>	<b>725</b>	<b>253</b>	<b>110</b>
NL_PID ff	$4 \times 10^{-3}$	$2 \times 10^{-3}$	$10^{-3}$	$7 \times 10^{-3}$	$3 \times 10^{-3}$	$1.7 \times 10^{-3}$	$7.7 \times 10^{-2}$	$4.4 \times 10^{-2}$	$3.45 \times 10^{-2}$
CC_PID	$1.5 \times 10^{-3}$	$5.8 \times 10^{-4}$	$2.6 \times 10^{-5}$	$1.15 \times 10^{-2}$	$5.79 \times 10^{-3}$	$2.5 \times 10^{-3}$	<b>0.352</b>	<b>0.124</b>	$4 \times 10^{-2}$
CC_PID ff	<b>0.16</b>	<b>0.081</b>	<b>0.040</b>	<b>3.54</b>	<b>1.56</b>	<b>0.74</b>	<b>728</b>	<b>254</b>	<b>110</b>

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_R$			$\Delta_R$			$\Delta_R$		
	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>
NL_PID	<b>0.7054</b>	<b>0.4278</b>	<b>0.1881</b>	<b>9.7963</b>	<b>6.5948</b>	<b>3.1795</b>	<b>1039.70</b>	<b>935.525</b>	<b>492.063</b>
NL_PID ff	<b>0.0124</b>	<b>0.0077</b> <b>955</b>	<b>0.003</b>	<b>0.020189</b>	<b>0.012435</b>	<b>0.00565</b>	<b>0.08193</b>	<b>0.06601</b>	<b>0.04766</b>
CC_PID	<b>0.0632</b>	<b>0.0426</b>	<b>0.0197</b>	<b>0.76096</b>	<b>0.51075</b>	<b>0.23453</b>	<b>9.3949</b>	<b>6.3209</b>	<b>2.9189</b>
CC_PID ff	<b>1.208</b>	<b>0.7108</b>	<b>0.30586</b>	<b>14.6411</b>	<b>9.6922</b>	<b>4.6216</b>	<b>1042.66</b>	<b>932.318</b>	<b>497.1229</b>

### 2.2. commande trajectoires PID

**Kp=100, Kv=10, Ki=125**

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_M$			$\Delta_M$			$\Delta_M$		
	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$	$\pm 100$	$\pm 50$	$\pm 25$
NL_PI D	<b>0.0044</b>	<b>0.0022</b>	<b>0.0011</b>	<b>0.0071</b>	<b>0.0035</b>	<b>0.00178</b>	<b>0.0772</b>	<b>0.0441</b>	<b>0.0345</b>
NL_PI D ff	<b>0.0044</b> <b>6</b>	<b>0.0022</b>	<b>0.0011</b>	<b>0.0071</b> <b>5</b>	<b>0.0035</b>	<b>0.00178</b>	<b>0.0772</b>	<b>0.0441</b>	<b>0.0345</b>
CC_PI D	<b>0,1065</b>	<b>0,03757</b>	<b>0,0164</b>	<b>0,2012</b>	<b>0,007101</b>	<b>0,030992</b>	<b>0.00136</b> <b>2</b>	<b>4.80</b> $\times$ <b>10<sup>-4</sup></b>	<b>0.00020</b>
CC_PI D ff	<b>0,4799</b>	<b>0,0238</b>	<b>0,1185</b>	<b>0.0005</b>	<b>2.0556</b> $\times$ <b>10<sup>-4</sup></b>	<b>0,09253</b>	<b>0.07878</b>	<b>0.0280</b>	<b>0.01224</b>

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	$\Delta_R$			$\Delta_R$			$\Delta_R$		
	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>	<b>0.15</b>	<b>0.1</b>	<b>0.05</b>
NL_PID	<b>0.0124</b>	<b>0.00779</b>	<b>0.0035</b>	<b>0.020189</b>	<b>0.012435</b>	<b>0.00565</b>	<b>0.08193</b>	<b>0.06601</b>	<b>0.047665</b>
NL_PID ff	<b>0.0124</b>	<b>0.00779</b>	<b>0.0035</b>	<b>0.020189</b>	<b>0.012435</b>	<b>0.00565</b>	<b>0.08193</b>	<b>0.06601</b>	<b>0.047665</b>
CC_PID	<b>0,8981</b>	<b>0,59565</b>	<b>0,2663</b>	<b>0.000169</b>	<b>0.000112</b>	<b>0,50331</b>	<b>0.01136</b>	<b>0.00756</b>	<b>0.003398</b>
CC_PID ff	<b>0.0001</b>	<b>0,9095</b>	<b>0,4799</b>	<b>0.000787</b>	<b>0.000525</b>	<b>0.00054</b>	<b>0.07854</b>	<b>0.06958</b>	<b>0.078781</b>

### 3. Effet de perturbation

#### Commande consignes PD

Fext 1=[0 0 0 0 0]

Fext 2=[0 10 0 0 0]

Fext 3=[0 20 0 0 0]

	$\varepsilon_p$			$\varepsilon_v$			$\varepsilon_a$		
	Fext			Fext			Fext		
	<b>Fext 1</b>	<b>Fext 2</b>	<b>Fext 3</b>	<b>Fext 1</b>	<b>Fext 2</b>	<b>Fext 3</b>	<b>Fext 1</b>	<b>Fext 2</b>	<b>Fext 3</b>
NL_PD	<b>0.0426</b>	<b>0.103</b>	<b>0.09976</b>	<b>0.51075</b>	<b>2.6022</b>	<b>2.4722</b>	<b>6.3209</b>	<b>690.10</b>	<b>654.876</b>
NL_PD ff	<b>0.1083</b>	<b>0.103</b>	<b>0.09976</b>	<b>2.7392</b>	<b>2.6022</b>	<b>2.4722</b>	<b>725.4025</b>	<b>690.10</b>	<b>654.876</b>
CC_PD	<b>0.0057</b>	<b>0.5504</b> <b>2</b>	<b>1.2823</b>	<b>0.06902</b>	<b>2.8593</b>	<b>6.0057</b>	<b>0.89407</b>	<b>45.822</b>	<b>92.5133</b>
CC_PD ff	<b>0.4314</b>	<b>0.3587</b> <b>7</b>	<b>0.34691</b>	<b>7.3345</b>	<b>6.543</b>	<b>6.1827</b>	<b>754.8131</b>	<b>715.80</b>	<b>678.7656</b>



## VI.3 Conclusion

- L'augmentation du gain réduit l'erreur due à une perturbation, mais accentue les écarts sur les trajectoires dues à une incertitude sur le modèle
- Les consignes constantes en échelon présentent les plus grands écarts par rapport à la trajectoire nominale non perturbée.
- Les écarts sont d'autant plus élevés que les grandeurs ne sont pas asservies, ce qui est le cas de l'accélération qui présente de erreurs transitoires élevées en mode échelon avant de se stabiliser (oscillations).
- La commande PID a pour effet d'annuler les effets des perturbations additionnelles sur le couple, mais augmente le temps de réponse.
- La commande par couple calculé linéarisante est celle qui présente la meilleure robustesse relativement aux incertitudes sur le modèle qu'on a analysées : masse, inertie et position du centre de masse.
- La commande par anticipation présente une complexité plus réduite dans sa mise en œuvre mais reste moins performante en terme de robustesse.
- Si le but poursuivi est la position finale, elle constitue une alternative séduisante à la précédente, particulièrement dans sa version PID.
- Bien qu'initialement, le sujet visait l'étude du robot Puma 560, les conclusions auxquelles on est parvenu avec le robot Pélikan peuvent être facilement étendu au robot Puma560 à 6 ddl. Les durées de simulation des circuits simulink étant particulièrement pénalisants.

# ***CONCLUSION***

Le but de ce travail était d'apprécier les performances des correcteurs standards PD et PID vis-à-vis des variations des paramètres du modèle. En effet ces commandes utilisent le modèle dynamique pour déterminer la loi de commande. Par ailleurs, dans le cas où des forces externes s'appliquent sur l'effecteur, il était important d'estimer leurs influences sur le suivi de trajectoire et la précision de pose.

Nous avons passé en revue toutes ces méthodes, et avons déterminé pour chaque type de perturbation, l'erreur quadratique globale des trajectoires par rapport aux consignes nominales non perturbées. Nous avons alors déterminé les lois les plus robustes et donné les sens des variations des paramètres permettant de réduire ces écarts. Nous avons également donné les justifications théoriques expliquant ce comportement.

Nous avons à cet effet, réalisé une fenêtre graphique sous Matlab, qui permet d'analyser la robustesse de ces commandes. Nous avons implanté pour chaque loi, le circuit Simulink correspondant auquel le logiciel fait appel. La boîte à outil Robotic de Matlab mise au point par P. I Corke depuis une vingtaine d'années et sans cesse améliorée par ses soins, très appréciée par la communauté universitaire pour les outils qu'elle offre, nous a été d'une grande utilité, particulièrement lors de la construction des circuits en utilisant les blocs du module Simulink livré avec le toolbox.

Néanmoins, les durées de la simulation de ces circuits calculs étant particulièrement longs pour le cas des robots ayant un nombre d'articulations supérieur à deux, nous avons écrit un programme sous Matlab qui implante la loi de commande désirée et calcule la trajectoire suivie dans les deux types de commande : consigne fixe ou variable (commande en poursuite). Une ébauche du code Matlab correspondant est donnée par le même auteur dans la version 6 (2001) pour le robot puma560.

Néanmoins les résultats obtenus pour le robot Pelikan sont transposables aux robots plus complexes, comme il a été établi par simulation.

Le choix des paramètres des correcteurs est primordial pour avoir un bon suivi des trajectoires et pour assurer des commandes stables au sens de Lyapunov. De nombreux travaux se sont penchés sur cet aspect et nous avons rappelé les principaux résultats auxquels ils sont parvenus.

Le modèle dynamique du robot doit être connu avec le maximum de précision afin de pouvoir établir la loi de commande qui exige dans le cas de la commande par couple calculé, l'inversion de la dynamique pour un calcul en temps réel. La complexité et le volume de calcul qui en résulte amène à déterminer le modèle dynamique à partir des valeurs de la trajectoire demandée (commande feedforward ou par anticipation) qui introduit une erreur résiduelle qu'il convient d'estimer et rechercher les moyens pour la réduire. Nous avons écrit un programme qui calcule analytiquement le modèle dynamique, en utilisant les équations de Lagrange. Le programme est donné en annexes A et B et détermine analytiquement les symboles de Christoffel à partir de la connaissance de la matrice de tenseur masse-inertie  $M$ .

En conclusion, dans le cas où des performances élevées ne sont pas recherchées, et qui nécessitent l'implantation de lois complexes utilisant des commandes du type adaptatif, la commande PID par

couple calculé avec compensation du couple de perturbation présente une bonne robustesse pour autant que les paramètres de réglage sont choisis soigneusement.

# ***ANNEXE***

**ANNEXE A-1 : PARAMETRES DH-MODIFIES DU ROBOT "PUMA560  
D'APRES PETER I. CORKE AVEC MODELE DYNAMIQUE OBJET ROBOT UTILISE DANS LA  
BOITE A OUTIL ROBOTIQUE VERSION**

```
%PUMA560AKB Load kinematic and dynamic data for a Puma 560 manipulator
%
% PUMA560AKB
% Armstrong, Khatib and Burdick 1986.
% "The Explicit Dynamic Model and Inertial Parameters of the Puma 560
Arm"
%
% Also define the vector qz which corresponds to the zero joint
% angle configuration, qr which is the vertical 'READY' configuration,
% and qstretch in which the arm is stretched out in the X direction.
%
% See also: ROBOT, PUMA560, STANFORD, TWOLINK.

% Copyright (C) 1993-2002, by Peter I. Corke
% CHANGES:
% 12/01 convert to object format
% $Log: puma560akb.m,v $
% Revision 1.3 2002/04/01 11:47:15 pic
% General cleanup of code: help comments, see also, copyright, remnant
dh/dyn
% references, clarification of functions.
%
% $Revision: 1.3 $

clear L
% alpha A theta D sigma
L{1} = link([0 0 0 0 0], 'mod');
L{2} = link([-pi/2 0 0 0.2435 0], 'mod');
L{3} = link([0 0.4318 0 -0.0934 0], 'mod');
L{4} = link([pi/2 -0.0203 0 .4331 0], 'mod');
L{5} = link([-pi/2 0 0 0 0], 'mod');
L{6} = link([pi/2 0 0 0 0], 'mod');

L{1}.m = 0;
L{2}.m = 17.4;
L{3}.m = 4.8;
L{4}.m = 0.82;
L{5}.m = 0.34;
L{6}.m = .09;

% rx ry rz
L{1}.r = [0 0 0];
L{2}.r = [0.068 0.006 -0.016];
L{3}.r = [0 -0.070 0.014];
L{4}.r = [0 0 -0.019];
L{5}.r = [0 0 0];
L{6}.r = [0 0 .032];

% Ixx Iyy Izz Ixy Iyz Ixz
L{1}.I = [0 0 0.35 0 0 0];
L{2}.I = [.13 .524 .539 0 0 0];
L{3}.I = [.066 .0125 .066 0 0 0];
L{4}.I = [1.8e-3 1.8e-3 1.3e-3 0 0 0];
L{5}.I = [.3e-3 .3e-3 .4e-3 0 0 0];
```

```

L{6}.I = [.15e-3 .15e-3 .04e-3 0 0 0];

L{1}.Jm = 291e-6;
L{2}.Jm = 409e-6;
L{3}.Jm = 299e-6;
L{4}.Jm = 35e-6;
L{5}.Jm = 35e-6;
L{6}.Jm = 35e-6;

L{1}.G = -62.6111;
L{2}.G = 107.815;
L{3}.G = -53.7063;
L{4}.G = 76.0364;
L{5}.G = 71.923;
L{6}.G = 76.686;

% viscous friction (motor referenced)
% unknown

% Coulomb friction (motor referenced)
% unknown

%
% some useful poses
%
qz = [0 0 0 0 0 0]; % zero angles, L shaped pose
qr = [0 -pi/2 pi/2 0 0 0]; % ready pose, arm up
qstretch = [0 0 pi/2 0 0 0]; % horizontal along x-axis

p560m = robot(L, 'Puma560-AKB', 'Unimation', 'AK&B');
clear L

```

**ANNEXE A-2 : PARAMETRES DH-STANDARDS DU ROBOT "PUMA560  
D'APRES PETER I. CORKE  
D'APRES PETER I. CORKE AVEC MODELE DYNAMIQUE OBJET ROBOT UTILISE DANS LA  
BOITE A OUTIL ROBOTIQUE VERSION**

```

%PUMA560 Load kinematic and dynamic data for a Puma 560 manipulator
%
% PUMA560
%
% Defines the object 'p560' in the current workspace which describes the
% kinematic and dynamic % characteristics of a Unimation Puma 560
manipulator
% using standard DH conventions.

```

```

% The model includes armature inertia and gear ratios.
%
% Also define the vector qz which corresponds to the zero joint
% angle configuration, qr which is the vertical 'READY' configuration,
% and qstretch in which the arm is stretched out in the X direction.
%
% See also: ROBOT, PUMA560AKB, STANFORD, TWOLINK.

%
% Notes:
%   - the value of m1 is given as 0 here.  Armstrong found no value for
it
% and it does not appear in the equation for tau1 after the substituiou
% is made to inertia about link frame rather than COG frame.
% updated:
% 2/8/95 changed D3 to 150.05mm which is closer to data from Lee, AKB86
and Tarn
% fixed errors in COG for links 2 and 3
% 29/1/91 to agree with data from Armstrong etal.  Due to their use
% of modified D&H params, some of the offsets Ai, Di are
% offset, and for links 3-5 swap Y and Z axes.
% 14/2/91 to use Paul's value of link twist (alpha) to be consistant
% with ARCL.  This is the -ve of Lee's values, which means the
% zero angle position is a righty for Paul, and lefty for Lee.
% Note that gravity load torque is the motor torque necessary
% to keep the joint static, and is thus -ve of the gravity
% caused torque.
%
% 8/95 fix bugs in COG data for Puma 560. This led to significant errors
in
% inertia of joint 1.
% $Log: puma560.m,v $
% Revision 1.3 2002/04/01 11:47:16 pic
% General cleanup of code: help comments, see also, copyright, remnant
dh/dyn
% references, clarification of functions.
%
% $Revision: 1.3 $

% Copyright (C) 1993-2002, by Peter I. Corke

clear L
L{1} = link([ pi/2 0 0      0      0], 'standard');
L{2} = link([ 0 .43180    0      0], 'standard');
L{3} = link([-pi/2 .0203  0      .15005    0], 'standard');
L{4} = link([pi/2 0 0      .4318 0], 'standard');
L{5} = link([-pi/2 0 0      0      0], 'standard');
L{6} = link([0 0 0      0      0], 'standard');

L{1}.m = 0;
L{2}.m = 17.4;
L{3}.m = 4.8;
L{4}.m = 0.82;
L{5}.m = 0.34;
L{6}.m = .09;

L{1}.r = [ 0      0      0 ];
L{2}.r = [ -.3638  .006  .2275];

```



```

L{3}.r = [ -.0203  -.0141  .070];
L{4}.r = [ 0      .019   0];
L{5}.r = [ 0      0       0];
L{6}.r = [ 0      0       .032];

L{1}.I = [ 0      0.35  0      0      0      0];
L{2}.I = [ .13   .524  .539  0      0      0];
L{3}.I = [ .066  .086  .0125  0  0  0];
L{4}.I = [ 1.8e-3  1.3e-3  1.8e-3  0  0  0];
L{5}.I = [ .3e-3   .4e-3   .3e-3   0  0  0];
L{6}.I = [ .15e-3  .15e-3  .04e-3  0  0  0];

L{1}.Jm = 200e-6;
L{2}.Jm = 200e-6;
L{3}.Jm = 200e-6;
L{4}.Jm = 33e-6;
L{5}.Jm = 33e-6;
L{6}.Jm = 33e-6;

L{1}.G = -62.6111;
L{2}.G = 107.815;
L{3}.G = -53.7063;
L{4}.G = 76.0364;
L{5}.G = 71.923;
L{6}.G = 76.686;

% viscous friction (motor referenced)
L{1}.B = 1.48e-3;
L{2}.B = .817e-3;
L{3}.B = 1.38e-3;
L{4}.B = 71.2e-6;
L{5}.B = 82.6e-6;
L{6}.B = 36.7e-6;

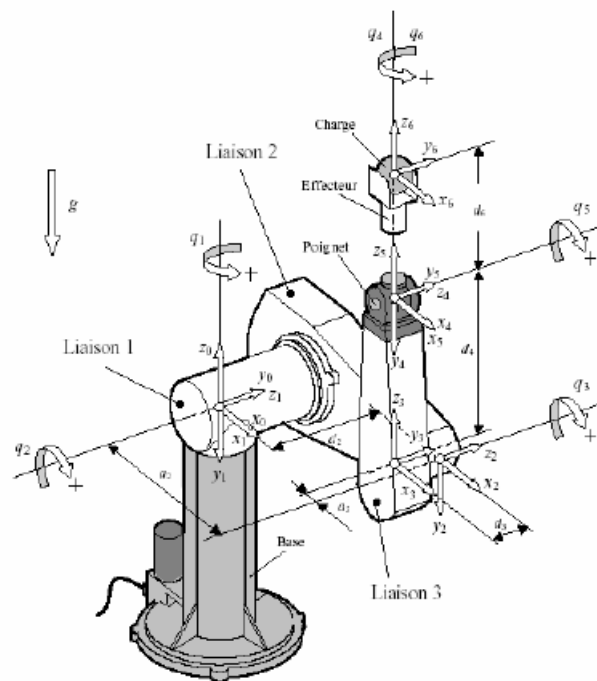
% Coulomb friction (motor referenced)
L{1}.Tc = [ .395-.435];
L{2}.Tc = [ .126-.071];
L{3}.Tc = [ .132-.105];
L{4}.Tc = [ 11.2e-3 -16.9e-3];
L{5}.Tc = [ 9.26e-3 -14.5e-3];
L{6}.Tc = [ 3.96e-3 -10.5e-3];

%
% some useful poses
%
qz = [0 0 0 0 0 0]; % zero angles, L shaped pose
qr = [0 pi/2 -pi/2 0 0 0]; % ready pose, arm up
qstretch = [0 0 -pi/2 0 0 0];

p560 = robot(L, 'Puma 560', 'Unimation', 'params of 8/95');
%clear L
p560.name = 'Puma 560';
p560.manuf = 'Unimation';

```

### A-3 GEOMETRIE DIRECTE DU ROBOT PUMA560



Représentation des articulations du Puma 560 ( modèle DH standard)

Table paramètres DH standards (à mettre dans une table)

dh=

[  $\pi/2$ , 0, 0, 0, 0 ]

$$[ 0, a_2, 0, 0, 0]$$

$$[-\pi/2, a_3, 0, d_3, 0]$$

$$[\pi/2, 0, 0, d_4, 0]$$

$$[-\pi/2, 0, 0, 0, 0]$$

$$[ 0, 0, 0, 0, 0]$$

$$T=[n \ o \ a \ P]$$

$$n_x = [(((\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) - \sin(t_1) \cdot \sin(t_4)) \cdot \cos(t_5) + (-\cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \cos(t_6) + (-\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) - \sin(t_1) \cdot \cos(t_4)) \cdot \sin(t_6)]$$

$$n_y = [(((\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) + \cos(t_1) \cdot \sin(t_4)) \cdot \cos(t_5) + (-\sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \cos(t_6) + (-\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) + \cos(t_1) \cdot \cos(t_4)) \cdot \sin(t_6)]$$

$$n_z = [(\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) \cdot \cos(t_5) + (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \cos(t_6) - (\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) \cdot \sin(t_6)]$$

$$o_x = -[(((\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) - \sin(t_1) \cdot \sin(t_4)) \cdot \cos(t_5) + (-\cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \sin(t_6) + (-\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) - \sin(t_1) \cdot \cos(t_4)) \cdot \cos(t_6)]$$

$$o_y = -[(((\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) + \cos(t_1) \cdot \sin(t_4)) \cdot \cos(t_5) + (-\sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \sin(t_6) + (-\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) + \cos(t_1) \cdot \cos(t_4)) \cdot \cos(t_6)]$$

$$o_z = [(\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) \cdot \cos(t_5) + (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \sin(t_6) - (\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) \cdot \cos(t_6)]$$

$$a_x = -[(\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) - \sin(t_1) \cdot \sin(t_4)) \cdot \sin(t_5) + (-\cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \cos(t_5)]$$

$$a_y = -[(\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) + \cos(t_1) \cdot \sin(t_4)) \cdot \sin(t_5) + (-\sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \cos(t_5)]$$

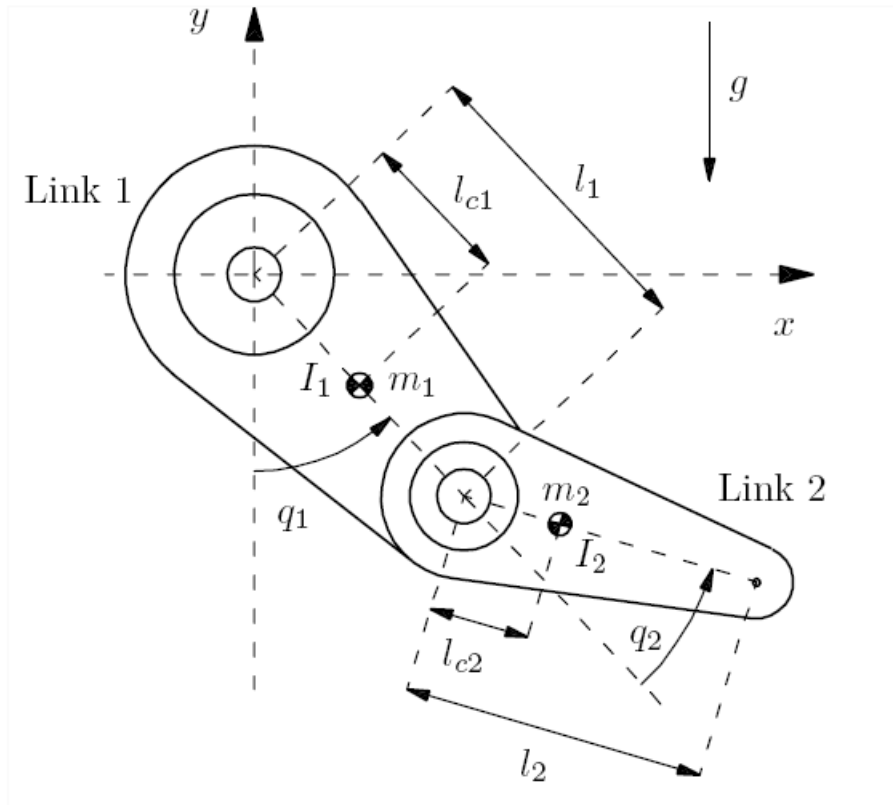
$$a_z = [(\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) \cdot \sin(t_5) + (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot \cos(t_5)]$$

$$P_x = [(-\cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot d_4 + \cos(t_1) \cdot \cos(t_2) \cdot a_3 \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot a_3 \cdot \sin(t_3) + \sin(t_1) \cdot d_3 + \cos(t_1) \cdot a_2 \cdot \cos(t_2)]$$

$$P_y = [(-\sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot d_4 + \sin(t_1) \cdot \cos(t_2) \cdot a_3 \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot a_3 \cdot \sin(t_3) - \cos(t_1) \cdot d_3 + \sin(t_1) \cdot a_2 \cdot \cos(t_2)]$$

$$P_z = [(-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot d_4 + \sin(t_2) \cdot a_3 \cdot \cos(t_3) + \cos(t_2) \cdot a_3 \cdot \sin(t_3) + a_2 \cdot \sin(t_2)]$$

ANNEXE B-1 : CALCUL ANALYTIQUE MODELE DYNAMIQUE ROBOT PELIKAN



Description	Notation	Value	Units
Length of Link 1	$l_1$	0.26	m
Length of Link 2	$l_2$	0.26	m
Distance to the center of mass (Link 1)	$l_{c1}$	0.0983	m
Distance to the center of mass (Link 2)	$l_{c2}$	0.0229	m
Mass of Link 1	$m_1$	6.5225	kg
Mass of Link 2	$m_2$	2.0458	kg
Inertia rel. to center of mass (Link 1)	$I_1$	0.1213	kg m <sup>2</sup>
Inertia rel. to center of mass (Link 2)	$I_2$	0.0116	kg m <sup>2</sup>
Gravity acceleration	$g$	9.81	m/s <sup>2</sup>

```

%alpha A theta D sigma m rx ry rz Ixx Iyy Izz Ixy Iyz
Ixz Jm G
syms l1 l2 lc1 lc2 m1 m2 Izz1 Izz2 t1 t2
mdyn] =
0 0 0 0 0 m1 lc1 0 0 0 0 Izz1 0 0 0 0 1
0 l1 0 0 0 m2 lc2 0 0 0 0 Izz2 0 0 0 0 1

```

**[M,C,B,G]=calcul\_fdyn(mdyn,[t1 d2],[0,-g,0])**

M =

[ m1\*lc1^2+Izz1+m2\*lc2^2+2\*m2\*lc2\*l1\*cos(t2)+m2\*l1^2+Izz2,  
m2\*lc2^2+m2\*lc2\*l1\*cos(t2)+Izz2]

[m2\*lc2^2+m2\*lc2\*l1\*cos(t2)+Izz2,

m2\*lc2^2+Izz2]

C =

[0 -m2\*lc2\*l1\*sin(t2)]

[m2\*lc2\*l1\*sin(t2) 0]

B =

[-2\*m2\*lc2\*l1\*sin(t2)]

[ 0]

G =

[ m1\*cos(t1)\*lc1\*g+m2\*(cos(t1+t2)\*lc2+cos(t1)\*l1)\*g]

[ m2\*cos(t1+t2)\*lc2\*g]

ANNEXE B-2 : PROGRAMME DE ALCUL ANALYTIQUE DU MODELE DYNAMIQUE D'UN ROBOT

```

function [M,C,B,G,Jv]=calcul_fdyn(dyn,q,gravity)
% forme = 3 : [M,G,C,B]    M(q)qdd+C[qd^2] +B(qdq)+G(q) Cc matrice de coriolis B(q):
forces centriguges
% definition du vecteur de gravitation par défaut g=[0 -g 0]
% dhy matrice dh modifiée avec paramètres dynamiques
% q : vecteur de position

if nargin==2
    syms g
    gravity=[-g; 0;0];
else
    gravity=gravity(:);
end
nn=size(dyn,1);
dhm=dyn(:,1:5);
m=dyn(:,6);
dec=dyn(:,7:9);
E=zeros(nn,nn);
G=zeros(nn,1);
for ii=1:nn
    [JJ,T,Tf]=calcul_Jv(ii,dhm,q,dec(ii,:));
    Ic=[dyn(ii,10) dyn(ii,13) dyn(ii,15)
        dyn(ii,13) dyn(ii,11) dyn(ii,14)
        dyn(ii,15) dyn(ii,14) dyn(ii,12)];
    Jv=JJ(1:3,:);Jw=JJ(4:6,:);
    E=E+transpose(Jv)*Jv*m(ii)+transpose(Jw)*Ic*Jw;
    G=G-transpose(Jv)*m(ii)*gravity;
end
[M,h]=simple(E);
for ii=1:nn
    for jj=1:nn
        C(ii,jj)=christoffel(M,ii,jj,jj,q)
    end
end
for ii=1:nn
    ll=1;
    for jj=1:nn-1
        for kk=jj+1:nn
            B(ii,ll)=2*christoffel(M,ii,jj,kk,q);
            ll=ll+1;
        end
    end
end
end

%*****
function b=christoffel(M,i,j,k,q);
b=0.5*(diff(M(i,j),q(k))+diff(M(i,k),q(j))-diff(M(j,k),q(i)));
%*****

```

**ANNEXE B-3 : FONCTION QUI CALCULE LES JACOBIENS ELEMENTAIRES ARTICULAIRES**

```

function [J,T,Tf]=calcul_Jv(nn,dhm,q,dec)
%FONCTION CALCUL_JAC1 : CALCULE LE JACOBIEN MANIPULATEUR PAR DERIVATION
%PARAMETRES ENTREES
%n : NUMERO DE REPERE DE 0 (BASE) A n (PINCE TERMINALE)
%dhm : MATRICE DES PARAMETRES DH MODIFIES
%q : VECTEUR COORDONNES ARTICULAIRES
%dec : VECTEUR DEPLACEMENT POUR POSITIONNER EFFECTEUR ( 3 VALEURS X Y Z)
% OU MATRICE 3X4 CONTENANT MATRICE ORIENTATION [nx ox ax] et POSITION
%PARAMETRES SORTIE
%J : MATRICE JACOBIEENNE RELATIVE AU REPERE INDICE n
%Tf : MATRICE HOMOGENE GENERALISEE FINALE RELATIVE A EFFECTEUR
%T : MATRICES HOMOGENES DIMENSION 3 DE CHAQUE ARTICULATION / AU REPERE
DE
%BASE
%nn=size(dhm,1);
if nargin==4
    if prod(size(dec))==12
        dec=[dec; 0 0 0 1];
    else
        dec=[eye(3),dec(:);0 0 0 1];
    end
else
    dec=[[eye(3),zeros(3,1)];0 0 0 1];
end
J=[];
for jj=1:nn %size(q,2)
    if jj==1
        T(:,:,jj)=mlinktra(dhm(jj,:),q(1));
    else
        T(:,:,jj)=T(:,:,jj-1)*mlinktra(dhm(jj,:),q(jj));
    end
end
    TT(:,:,nn)=T(:,:,nn)*dec;

for ii=1:size(q,2)
    xx=TT(1,4,nn);
    yy=TT(2,4,nn);

```

```

zz=TT(3,4,nn);
if ii<=nn
dP=[diff(xx,q(ii));diff(yy,q(ii));diff(zz,q(ii))];
  if dhm(ii,5)==0
    z=T(1:3,3,ii);
    J=[J,[dP;z]];
  else
    J=[J,[dP;zeros(3,1)]];
  end
else
  J=[J,zeros(6,1)];
end

```

end

```

if ~isnumeric(dhm)
  [J,who]=simple(J);
  [Tf,who]=simple(TT(:,:,nn));
  for ii=1:nn
    [T(:,:,ii),who]=simple(T(:,:,ii));
  end
end

```



FICHER : Tofik

-----

Fichier crée le : 20-Oct-2014 à : 11h:40mn:47s

\*\*\*\*\*

NOM ROBOT : Simple two link

\*\*\*\*\*

NOMBRE DE DEGRES DE LIBERTE : 2

\*\*\*\*\*

CONVENTION : STANDARD

\*\*\*\*\*

PARAMETRES STATIQUES : alpha a theta d sigma :

\*\*\*\*\*

[0 0 0 0 0]

[0 0.26 0 0 0]

\*\*\*\*\*

PARAMETRES DYNAMIQUES :

LIAISON N° 1

Masse : 6.5225  
Position Centre de Masse : [0.0983 0 0]  
Matrice Inertie Liaison : [0 0 0;0 0 0;0 0 0.1213]  
Moment Inertie Moteur : 0.1  
Rapport réduction Moteur : 1  
Frottement Moteur : 0  
Coefficient Frottement manipulateur : [0 0]

LIAISON N° 2

Masse : 2.0458  
Position Centre de Masse : [0.0229 0 0]  
Matrice Inertie Liaison : [0 0 0;0 0 0;0 0 0.0116]  
Moment Inertie Moteur : 0.1  
Rapport réduction Moteur : 1  
Frottement Moteur : 0  
Coefficient Frottement manipulateur : [0 0]

FORCE EXTERNE APPLIQUEE SUR EFFECTEUR : [0 0 0 0 0 0]

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

PARAMETRES CORRECTEUR :

\*\*\*\*\*

```
TYPE CORRECTEUR                : PD
Kp                              : 100
Kd                              : 10
*****
DUREE PARCOURS TRAJECTOIRE     : 5
*****
TYPE ENTREE                    : TRAJECTOIRE
*****
POSITIONS ARTICULAIRES INITIALES : [0 0]
POSITIONS ARTICULAIRES FINALES  : [1.0471 1.570]
*****
ERREUR POSITION                  : 0.
ERREUR VITESSE                 :
ERREUR ACCELERATION            :
```

**ANNEXE D-1 : PARAMETRES DH-MODIFIES DU ROBOT "PUMA560  
D'APRES PETER I. CORKE AVEC MODELE DYNAMIQUE OBJET ROBOT UTILISE DANS LA  
BOITE A OUTIL ROBOTIQUE VERSION**

```
%PUMA560AKB Load kinematic and dynamic data for a Puma 560 manipulator
%
% PUMA560AKB
% Armstrong, Khatib and Burdick 1986.
% "The Explicit Dynamic Model and Inertial Parameters of the Puma 560
Arm"
%
% Also define the vector qz which corresponds to the zero joint
% angle configuration, qr which is the vertical 'READY' configuration,
% and qstretch in which the arm is stretched out in the X direction.
%
% See also: ROBOT, PUMA560, STANFORD, TWOLINK.

% Copyright (C) 1993-2002, by Peter I. Corke
% CHANGES:
% 12/01 convert to object format
% $Log: puma560akb.m,v $
% Revision 1.3 2002/04/01 11:47:15 pic
% General cleanup of code: help comments, see also, copyright, remnant
dh/dyn
% references, clarification of functions.
%
% $Revision: 1.3 $

clear L
%          alpha   A      theta   D sigma
L{1} = link([0      0      0      0      0], 'mod');
L{2} = link([-pi/2  0      0      0.2435 0], 'mod');
L{3} = link([0      0.4318 0      -0.0934 0], 'mod');
L{4} = link([pi/2  -0.0203 0      .4331  0], 'mod');
L{5} = link([-pi/2  0      0      0      0], 'mod');
L{6} = link([pi/2  0      0      0      0], 'mod');

L{1}.m = 0;
L{2}.m = 17.4;
L{3}.m = 4.8;
L{4}.m = 0.82;
L{5}.m = 0.34;
L{6}.m = .09;

%          rx      ry      rz
L{1}.r = [0  0  0  ];
L{2}.r = [0.068  0.006  -0.016];
L{3}.r = [0  -0.070  0.014  ];
L{4}.r = [0  0  -0.019];
L{5}.r = [0  0  0  ];
L{6}.r = [0  0  .032  ];

%          Ixx      Iyy      Izz      Ixy      Iyz      Ixz
L{1}.I = [0  0  0.35  0  0  0];
L{2}.I = [.13  .524  .539  0  0  0];
L{3}.I = [.066  .0125  .066  0  0  0];
L{4}.I = [1.8e-3  1.8e-3  1.3e-3  0  0  0];
L{5}.I = [.3e-3  .3e-3  .4e-3  0  0  0];
L{6}.I = [.15e-3  .15e-3  .04e-3  0  0  0];

L{1}.Jm = 291e-6;
```

```

L{2}.Jm = 409e-6;
L{3}.Jm = 299e-6;
L{4}.Jm = 35e-6;
L{5}.Jm = 35e-6;
L{6}.Jm = 35e-6;

L{1}.G = -62.6111;
L{2}.G = 107.815;
L{3}.G = -53.7063;
L{4}.G = 76.0364;
L{5}.G = 71.923;
L{6}.G = 76.686;

% viscous friction (motor referenced)
% unknown

% Coulomb friction (motor referenced)
% unknown

%
% some useful poses
%
qz = [0 0 0 0 0 0]; % zero angles, L shaped pose
qr = [0 -pi/2 pi/2 0 0 0]; % ready pose, arm up
qstretch = [0 0 pi/2 0 0 0]; % horizontal along x-axis

p560m = robot(L, 'Puma560-AKB', 'Unimation', 'AK&B');
clear L

```

**ANNEXE D-2 : PARAMETRES DH-STANDARDS DU ROBOT "PUMA560  
D'APRES PETER I. CORKE  
D'APRES PETER I. CORKE AVEC MODELE DYNAMIQUE OBJET ROBOT UTILISE DANS LA  
BOITE A OUTIL ROBOTIQUE VERSION**

```

%PUMA560 Load kinematic and dynamic data for a Puma 560 manipulator
%
% PUMA560
%
% Defines the object 'p560' in the current workspace which describes the
% kinematic and dynamic % characteristics of a Unimation Puma 560 manipulator
% using standard DH conventions.
% The model includes armature inertia and gear ratios.
%
% Also define the vector qz which corresponds to the zero joint
% angle configuration, qr which is the vertical 'READY' configuration,
% and qstretch in which the arm is stretched out in the X direction.
%
% See also: ROBOT, PUMA560AKB, STANFORD, TWOLINK.

```

```

%
% Notes:
% - the value of m1 is given as 0 here.  Armstrong found no value for it
% and it does not appear in the equation for tau1 after the substitution
% is made to inertia about link frame rather than COG frame.
% updated:
% 2/8/95  changed D3 to 150.05mm which is closer to data from Lee, AKB86 and
Tarn
% fixed errors in COG for links 2 and 3
% 29/1/91 to agree with data from Armstrong et al.  Due to their use
% of modified D&H params, some of the offsets Ai, Di are
% offset, and for links 3-5 swap Y and Z axes.
% 14/2/91 to use Paul's value of link twist (alpha) to be consistent
% with ARCL.  This is the -ve of Lee's values, which means the
% zero angle position is a righty for Paul, and lefty for Lee.
% Note that gravity load torque is the motor torque necessary
% to keep the joint static, and is thus -ve of the gravity
% caused torque.
%
% 8/95 fix bugs in COG data for Puma 560. This led to significant errors in
% inertia of joint 1.
% $Log: puma560.m,v $
% Revision 1.3 2002/04/01 11:47:16 pic
% General cleanup of code: help comments, see also, copyright, remnant
dh/dyn
% references, clarification of functions.
%
% $Revision: 1.3 $

% Copyright (C) 1993-2002, by Peter I. Corke

clear L
L{1} = link([ pi/2 0 0 0 0 ], 'standard');
L{2} = link([ 0 .4318 0 0 0 ], 'standard');
L{3} = link([-pi/2 .0203 0 .15005 0 ], 'standard');
L{4} = link([pi/2 0 0 .4318 0 ], 'standard');
L{5} = link([-pi/2 0 0 0 0 ], 'standard');
L{6} = link([0 0 0 0 0 ], 'standard');

L{1}.m = 0;
L{2}.m = 17.4;
L{3}.m = 4.8;
L{4}.m = 0.82;
L{5}.m = 0.34;
L{6}.m = .09;

L{1}.r = [ 0 0 0 ];
L{2}.r = [ -.3638 .006 .2275];
L{3}.r = [ -.0203 -.0141 .070];
L{4}.r = [ 0 .019 0];
L{5}.r = [ 0 0 0];
L{6}.r = [ 0 0 .032];

L{1}.I = [ 0 0.35 0 0 0 0];
L{2}.I = [ .13 .524 .539 0 0 0];
L{3}.I = [ .066 .086 .0125 0 0 0];
L{4}.I = [ 1.8e-3 1.3e-3 1.8e-3 0 0 0];
L{5}.I = [ .3e-3 .4e-3 .3e-3 0 0 0];
L{6}.I = [ .15e-3 .15e-3 .04e-3 0 0 0];

```

```

L{1}.Jm = 200e-6;
L{2}.Jm = 200e-6;
L{3}.Jm = 200e-6;
L{4}.Jm = 33e-6;
L{5}.Jm = 33e-6;
L{6}.Jm = 33e-6;

L{1}.G = -62.6111;
L{2}.G = 107.815;
L{3}.G = -53.7063;
L{4}.G = 76.0364;
L{5}.G = 71.923;
L{6}.G = 76.686;

% viscous friction (motor referenced)
L{1}.B = 1.48e-3;
L{2}.B = .817e-3;
L{3}.B = 1.38e-3;
L{4}.B = 71.2e-6;
L{5}.B = 82.6e-6;
L{6}.B = 36.7e-6;

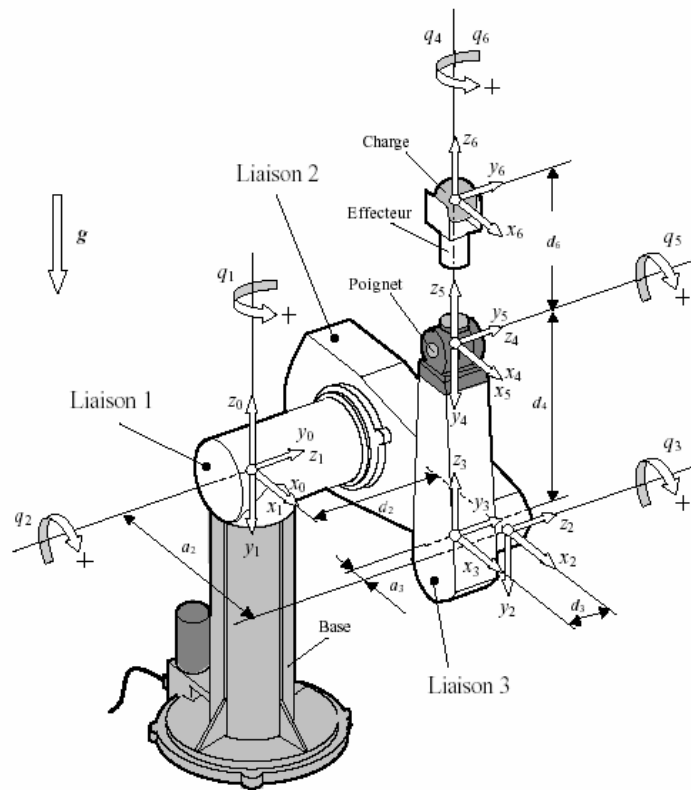
% Coulomb friction (motor referenced)
L{1}.Tc = [ .395    -.435];
L{2}.Tc = [ .126    -.071];
L{3}.Tc = [ .132    -.105];
L{4}.Tc = [ 11.2e-3 -16.9e-3];
L{5}.Tc = [ 9.26e-3 -14.5e-3];
L{6}.Tc = [ 3.96e-3 -10.5e-3];

%
% some useful poses
%
qz = [0 0 0 0 0 0]; % zero angles, L shaped pose
qr = [0 pi/2 -pi/2 0 0 0]; % ready pose, arm up
qstretch = [0 0 -pi/2 0 0 0];

p560 = robot(L, 'Puma 560', 'Unimation', 'params of 8/95');
%clear L
p560.name = 'Puma 560';
p560.manuf = 'Unimation';

```

### D-3 GEOMETRIE DIRECTE DU ROBOT PUMA560



**Représentation des articulations du Puma 560 ( modèle DH standard)**

Table paramètres DH standards (à mettre dans une table)

dh=

[ pi/2,	0,	0,	0,	0]
[ 0,	a2,	0,	0,	0]
[ -pi/2,	a3,	0,	d3,	0]
[ pi/2,	0,	0,	d4,	0]
[ -pi/2,	0,	0,	0,	0]
[ 0,	0,	0,	0,	0]

T=[n o a P]

$$nx = [ ((\cos(t1) \cdot \cos(t2) \cdot \cos(t3) - \cos(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \cos(t4) - \sin(t1) \cdot \sin(t4)) \cdot \cos(t5) + (-\cos(t1) \cdot \cos(t2) \cdot \sin(t3) - \cos(t1) \cdot \sin(t2) \cdot \cos(t3)) \cdot \sin(t5)) \cdot \cos(t6) + (-\cos(t1) \cdot \cos(t2) \cdot \cos(t3) - \cos(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \sin(t4) - \sin(t1) \cdot \cos(t4)) \cdot \sin(t6)]$$

$$ny = [ ((\sin(t1) \cdot \cos(t2) \cdot \cos(t3) - \sin(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \cos(t4) + \cos(t1) \cdot \sin(t4)) \cdot \cos(t5) + (-\sin(t1) \cdot \cos(t2) \cdot \sin(t3) - \sin(t1) \cdot \sin(t2) \cdot \cos(t3)) \cdot \sin(t5)) \cdot \cos(t6) + (-\sin(t1) \cdot \cos(t2) \cdot \cos(t3) - \sin(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \sin(t4) + \cos(t1) \cdot \cos(t4)) \cdot \sin(t6)]$$

$$nz = [ (\sin(t2) \cdot \cos(t3) + \cos(t2) \cdot \sin(t3)) \cdot \cos(t4) \cdot \cos(t5) + (-\sin(t2) \cdot \sin(t3) + \cos(t2) \cdot \cos(t3)) \cdot \sin(t5)) \cdot \cos(t6) - (\sin(t2) \cdot \cos(t3) + \cos(t2) \cdot \sin(t3)) \cdot \sin(t4) \cdot \sin(t6)]$$

$$ox = [ -(((\cos(t1) \cdot \cos(t2) \cdot \cos(t3) - \cos(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \cos(t4) - \sin(t1) \cdot \sin(t4)) \cdot \cos(t5) + (-\cos(t1) \cdot \cos(t2) \cdot \sin(t3) - \cos(t1) \cdot \sin(t2) \cdot \cos(t3)) \cdot \sin(t5)) \cdot \sin(t6) + (-\cos(t1) \cdot \cos(t2) \cdot \cos(t3) - \cos(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \sin(t4) - \sin(t1) \cdot \cos(t4)) \cdot \cos(t6)]$$

$$oy = [ -(((\sin(t1) \cdot \cos(t2) \cdot \cos(t3) - \sin(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \cos(t4) + \cos(t1) \cdot \sin(t4)) \cdot \cos(t5) + (-\sin(t1) \cdot \cos(t2) \cdot \sin(t3) - \sin(t1) \cdot \sin(t2) \cdot \cos(t3)) \cdot \sin(t5)) \cdot \sin(t6) + (-\sin(t1) \cdot \cos(t2) \cdot \cos(t3) - \sin(t1) \cdot \sin(t2) \cdot \sin(t3)) \cdot \sin(t4) + \cos(t1) \cdot \cos(t4)) \cdot \cos(t6)]$$

$$\begin{aligned}
oz &= ((\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) \cdot \cos(t_5) + (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot \sin(t_5)) \cdot \sin(t_6) - \\
& (\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \sin(t_4) \cdot \cos(t_6) \\
ax &= -((\cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) - \sin(t_1) \cdot \sin(t_4)) \cdot \sin(t_5) + (- \\
& \cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \cos(t_5) \\
ay &= -((\sin(t_1) \cdot \cos(t_2) \cdot \cos(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) + \cos(t_1) \cdot \sin(t_4)) \cdot \sin(t_5) + (- \\
& \sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot \cos(t_5) \\
az &= (\sin(t_2) \cdot \cos(t_3) + \cos(t_2) \cdot \sin(t_3)) \cdot \cos(t_4) \cdot \sin(t_5) + (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot \cos(t_5) \\
Px &= (-\cos(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \cos(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot d_4 + \cos(t_1) \cdot \cos(t_2) \cdot a_3 \cdot \cos(t_3) - \\
& \cos(t_1) \cdot \sin(t_2) \cdot a_3 \cdot \sin(t_3) + \sin(t_1) \cdot d_3 + \cos(t_1) \cdot a_2 \cdot \cos(t_2) \\
Py &= (-\sin(t_1) \cdot \cos(t_2) \cdot \sin(t_3) - \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)) \cdot d_4 + \sin(t_1) \cdot \cos(t_2) \cdot a_3 \cdot \cos(t_3) - \\
& \sin(t_1) \cdot \sin(t_2) \cdot a_3 \cdot \sin(t_3) - \cos(t_1) \cdot d_3 + \sin(t_1) \cdot a_2 \cdot \cos(t_2) \\
Pz &= (-\sin(t_2) \cdot \sin(t_3) + \cos(t_2) \cdot \cos(t_3)) \cdot d_4 + \sin(t_2) \cdot a_3 \cdot \cos(t_3) + \cos(t_2) \cdot a_3 \cdot \sin(t_3) + a_2 \cdot \sin(t_2)
\end{aligned}$$



# ***Bibliographie***

## REFERENCES BIBLIOGRAPHIQUES

- [1] Jean-Louis Boimond ISTIA, *monographie sur la robotique* Université Angers
- [2] W. Khalil, E. Dombre, *Modeling, Identification & Control of Robots*, HermesPenton Science 2002,
- [3] André Preumont *Théorie Générale des Systèmes articulées. Introduction à la Robotique* 2nde Edition. Université Libre de Bruxelles.2001
- [4] W.Khalil, G. Lebret *Modélisation et commande des robots*, Cours EI3 Automatique de l'ECN 94
- [5] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar *Robot Modeling and Control*.First Edition John Wiley & Sons, Inc.
- [6] Rosheim, M.E., *Robot Evolution -The Development of Anthrobotics*, John Wiley & Sons, 1994.
- [7] Denavit, J., et Hartenberg, R.S., < *A kinematic notation for lower pair mechanisms based on matrices* >, ASME Journal of Applied Mechanics, Vol. 6, pp. 215{221, 1955.
- [8] Lipkin, H., < *A note on Denavit-Hartenberg notation in robotics* >, Proceedings of IDETC/CIE 2005 ASME 2005 International Design Engineering Technical Conferences, Long Beach, California, \_Etats-Unis, 24{28 septembre 2005.
- [9] Craig, J.J., *Introduction to Robotics : Mechanics and Control, 3 rd edition*, Pearson Education, 2005.
- [10] Khalil W. Khalil et J. Kleinfinger. *A New Geometric Notation for Open and Closed Loop Robots*.ICRA'86, pages 75–79, avril 1986.
- [11] R. Kelly, V. Santibáñez A. Loría *Control of robot manipulators in joint space* .Springer\_2005
- [12] John J. Craig *Introduction to robotic. Mechanics and control*. 3rd Edition. Perason. Prentice Hall. 2005
- [13] J.Y.S. Luh, M.W. Walker, and R.P. Paul, "On-Line Computational Scheme for Mechanical Manipulators," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, 1980.
- [14] Joos, G. & al. 1987 *Entraînement à Courant Continu*, Canada, Presses de l'Université du Québec, 1987.
- [15] Khalil W., Liègois A., Fournier A., "Commande dynamique de robots," *RAIRO Automatique/Systems Analysis and Control*, Vol. 13, 1979, p. 189-201.
- [16] Egeland O., " *On the robustness of the computed torque technique in manipulator Control* ", Proc.IEEE. intern. Conf. Robotics and Automation, 1986, p. 1203-1208.
- [17] Craig J.-J., Hsu P., and SastryS.S., "Adaptive control of mechanical manipulators," *The International Journal of Robotics Research*, Vol. 6, 1987, p. 16-28.

- [18] Middleton R H., Goodwin G C., " *Adaptive computed torque control for rigide link manipulators*, ". Systems and Control Letters, Vol. 10, n° 10, 1988, p. 9-16.
- [19] Ortega R., Spong M W. " *Adaptive motion control of rigid robots : a tutorial* ", Automatica, Vol. 25, 1989, p. 877-888
- [20] Liu C.-S., Peng H., " *Disturbance observer based tracking control*, " ASME Journal of Dynamic Systems, Measurement, and Control , Vol. 122, 2000, p. 332-335.
- [21] Chen W-H., D. J. Balance, P. J. Gawthrop, J.J.Gribble and J. O'Reilly, " *A nonlinear disturbance observer for two-link robotic manipulators*, ". IEEE Trans. onIndustrialElectronicsVol. 47, 2000, 932–938.
- [22] Khalil W., Liègois A., Fournier A., " *Commande dynamique de robots*, " RAIRO Automatique/Systems analysis and control, Vol. 13, 1979, p. 189-201.
- [23] Peter I. Corke. *Robotic Toolbox.Version 3*.February 1996
- [24] Peter I Corke: *Robotic toolbox version 7* .Simulink circuits. CSIRO Manufacturing Science and Technology Pullenvale, AUSTRALIA, 4069. 2002 <http://www.cat.csiro.au/cmst/staff/pic/robot>