

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



ECOLE NATIONALE POLYTECHNIQUE
Département de Génie Electrique

PROJET DE FIN D'ETUDES
En vue de l'obtention du diplôme
D'ingénieur d'état en Automatique
Promotion 2006

Thème

**CONCEPTION, REALISATION ET
COMMANDE D'UN ROBOT
SCARA**

Proposé et Dirigé par :

M. H.CHEKIREB

M. O.STIHI

Etudié par :

CHIKH Lotfi

ZIOUI Nadjet

E.N.P. 10, Avenue Hassen Badi, 16200 EL-Harrach, Alger.

Dédicaces

Je dédie ce travail à mon petit ange, à la lumière de mes yeux, à l'être que j'aime et que je chéris plus que tout au monde : à maman.

Je dédie aussi ce travail à mon père que j'adore, à mes frère et sœurs, surtout à mon amie, confidente et très chère sœur Abba. Que Dieu garde ma petite famille et la protège.

Aussi, je cite tous les professeurs et enseignants que j'ai eus depuis mes premières années du primaire jusqu'au jour d'aujourd'hui. Je ne leur exprimerais jamais assez mon estime, mes respects et ma gratitude.

Je ne saurais oublier mes amis (es), et en particulier notre belle et très belle promotion d'Automatique (2003-2006). Je leur souhaite à tous de très beaux succès dans la vie.

Je dédie enfin ce travail à toute personne ayant contribué de près ou de loin à sa concrétisation.

Nadjet ZIOUI.

*Je dédie ce travail
A tous ceux qui me sont chers, que j'aime et qui m'aiment*

Lotfi CHIKH

Remerciements

Nous tenons à remercier nos promoteurs M. H.Chekirib et M. O.Stihi pour leurs précieux conseils, leur aide inestimable et la confiance qu'ils nous ont accordée et qui nous a permis de mener à terme ce travail.

Nous remercions les membres du jury qui nous ont fait l'honneur d'examiner notre travail.

Nos vifs remerciements vont aussi à M. Y.Bounnah et M. Guergueb pour l'aide apportée lors de la conception mécanique et l'élaboration des plans du robot.

Nous ne remercierons jamais assez l'équipe fort sympathique de la division fonderie de la SNVI de Rouiba pour l'accueil exceptionnel qu'ils nous ont réservé. En particulier, M. Naili, M. Mabraḳ, Abdelmaleḳ, Smaïl, M. Alouane, M. Laouni et tous ceux dont nous ne connaissons pas le nom et qui nous ont aidé, orienté et encouragé.

Nous tenons à remercier tous nos professeurs du Primaire à l'Université.

Merci à toute personne ayant contribué de près ou de loin à la concrétisation de ce projet.

TABLE DES MATIERES

INTRODUCTION GENERALE.....	1
CHAPITRE I : CONCEPTION ET REALISATION.....	3
I.1 Introduction.....	3
I.2 Généralités sur les robots manipulateurs.....	4
I.2.1 Définitions de base.....	4
I.2.2 Caractérisation d'un robot.....	5
I.2.3 Classification des robots.....	6
I.3 Démarche adoptée et feuille de route.....	7
I.4 Conception du robot.....	10
I.4.1 Structures proposées et structure finale.....	11
I.4.2 Présentation des pièces du robot.....	12
I.4.3 Présentation des actionneurs utilisés.....	13
I.5 Réalisation du robot.....	14
I.5.1 Bref historique de la SNVI.....	15
I.5.2 Organisation du travail.....	16
I.5.3 Retouches et améliorations.....	16
I.5 Conclusion.....	18
CHAPITRE II : MODELISATION.....	20
II.1 Introduction.....	20
II.2 Modèles géométriques.....	21
II.2.1 Modèle géométrique direct.....	21
II.2.2 Modèle géométrique inverse.....	26
II.2.2.1 Propriétés générales des solutions.....	26
II.2.2.2 Calcul du MGI du robot SCARA.....	27
II.3 Modèles cinématiques.....	29
II.3.1 Modèle cinématique direct.....	29
II.3.2 Modèle cinématique inverse.....	29
II.4 Modèle dynamique.....	31
II.4.1 Equations générales du mouvement.....	31
II.4.2 Energie cinétique totale du manipulateur.....	32
II.4.2.1 Tenseur d'inertie.....	33
II.4.2.2 Vitesses de translation et de rotation.....	34
II.4.3 Energie potentielle.....	35

II.4.4 Force généralisée.....	36
II.4.5 Application de l’algorithme d’Euler Lagrange pour le robot SCARA.....	37
II.5 Modèle du moteur à courant continu.....	39
II.6 Conclusion.....	42
CHAPITRE III : COMMANDE ET SIMULATION.....	45
III.1 Introduction.....	45
III.2 Génération de trajectoire.....	46
III.3 Commande linéarisante par géométrie différentielle.....	50
III.3.1 Equations d’état et linéarisation du système.....	50
III.3.1.1 Calcul du degré relatif.....	51
III.3.1.2 Calcul de la commande linéarisante.....	53
III.3.1.3 Forme normale.....	54
III.3.3 Commande du système linéarisé.....	56
III.3.3.1 Retour d’état.....	57
III.3.3.2 Backstepping.....	66
III.3.3.3 Modes de glissement.....	69
III.4 Commande par couple calculé.....	76
III.5 Commande décentralisée.....	79
III.6 Analyse des résultats obtenus.....	83
III.7 Conclusion.....	85
CHAPITRE IV : PLANIFICATION DE TACHES ET IMPLEMENTATION.....	86
IV.1 Introduction.....	87
IV.2 Planification de tâches.....	88
IV.2.1 Tâche1: PICK AND PLACE.....	88
IV.2.2 Tâche 2 : Ecriture d’un mot entier.....	90
IV.2.3 Tâche 3 : Alphabet et écriture de phrases.....	97
IV.2.3.1 Principe.....	97
IV.2.3.2 Implémentation des lettres de l’alphabet et des chiffres.....	98
IV.2.3.3 Ecriture de phrases.....	100
IV.3 Présentation du contrôleur UMAC.....	101
IV.3.1 Constitution matérielle et fonctionnement.....	101
IV.3.1 Aspect logiciel de l’interface.....	102
IV.4 Implémentation des programmes de mouvement.....	104
IV.4.1 Elaboration d’un programme de mouvement.....	104
IV.4.2 Exécution d’un programme de mouvement.....	105
IV.4.3 Exemple de programme.....	106
IV.5 Conclusion.....	106
CONCLUSION GENERALE.....	107

ANNEXES	111
Annexe A	111
Annexe A. 1 Plans de coupe.....	111
Annexe A. 2 Espace de travail du robot SCARA.....	115
Annexe B	116
Annexe B. 1 Algorithme DH.....	116
Annexe B. 2 Huygens généralisé.....	117
Annexe B. 3 Calcul de la matrice Jacobienne $J(q)$	119
Annexe B. 4 Algorithme d'Euler Lagrange.....	124
 BIBLIOGRAPHIE	 126

LISTE DES FIGURES

Figure I.1 Structures proposées et structure finale.....	11
Figure I.2 Nomenclature des pièces du robot.....	13
Figure I.3 Différents moteurs Pittman.....	13
Figure I.4 Différentes vues du moteur GM9236 avec encodeur.....	14
Figure I.5 Division Fonderie de la SNVI à Rouiba.....	15
Figure I.6 Vue générale du robot.....	18
Figure II.1. Méthodologie adoptée pour le calcul du Modèle géométrique direct.....	22
Figure II.2. Schéma des différentes articulations et repères selon la norme D-H.....	23
Figure II.3. Modèle géométrique direct et inverse.....	26
Figure II.4 Mouvement du corps k	32
Figure II.5 Circuit de fonctionnement du moteur.....	40
Figure II.6 Schéma bloc du système réduit.....	42
Figure III.1 Profil des variables articulaires générées.....	48
Figure III.2 Profil des vitesses articulaires générées.....	49
Figure III.3 Consignes opérationnelles correspondantes.....	49
Figure III.4 Résultats de la poursuite après application d'un retour d'état simple.....	57
Figure III.5 Erreurs de poursuite pour le retour d'état simple.....	58
Figure III.6 Signaux de commande réels correspondant au retour d'état simple.....	58
Figure III.7 Résultats de la poursuite après application d'un retour d'état avec ajout du filtre précompensateur.....	59
Figure III.8 Erreurs de poursuite pour le retour d'état avec ajout du filtre précompensateur.....	60
Figure III.9 Signaux de commande réels correspondant au retour d'état avec ajout du filtre précompensateur.....	60
Figure III.10 Résultats de la poursuite après application d'un retour d'état avec intégration de l'erreur de poursuite.....	62
Figure III.11 Erreurs de poursuite pour le retour d'état avec intégration de l'erreur de poursuite.....	62
Figure III.12 Signaux de commande réels correspondant au retour d'état avec intégration de l'erreur de poursuite.....	63
Figure III.13 Résultats de la poursuite après application d'un retour d'état après modification du signal de référence.....	64

Figure III.14 Erreurs de poursuite pour le retour d'état avec signal de référence modifié.....	65
Figure III.15 Signaux de commande réels correspondant au retour d'état avec référence modifiée.....	65
Figure III.16 Résultats de la poursuite après application de la commande par Backstepping.....	68
Figure III.17 Erreurs de poursuite pour la commande par Backstepping.....	68
Figure III.18 Signaux de la commande par Backstepping.....	69
Figure III.19 Résultats de la poursuite après application de la commande par modes de glissement.....	71
Figure III.20 Erreurs de poursuite correspondant à l'application de la commande par modes de glissement.....	71
Figure III.21 Signaux de commande réels correspondant à l'application de la commande par modes de glissement.....	72
Figure III.22 Divergence des sorties après changement des paramètres de la commande par modes de glissement ($\lambda = 5$ et $K = 0.001$).....	73
Figure III.23 Divergence des sorties après changement des paramètres de la commande par modes de glissement ($\lambda = 2$ et $K = 1$).....	73
Figure III.24 Comparaison des résultats de poursuite entre les commandes de la géométrie différentielle.....	74
Figure III.25 Comparaison des erreurs de poursuite entre les commandes de la géométrie différentielle.....	75
Figure III.26 Comparaison entre les commandes de la géométrie différentielle.....	75
Figure III.27 Résultats de la poursuite dans l'espace articulaire après application de la commande par couple calculé.....	77
Figure III.28 Erreurs de poursuite correspondant à l'application de la commande par couple calculé.....	77
Figure III.29 Signaux de commande réels correspondant à l'application de la commande par couple calculé.....	78
Figure III.30 Résultats de la poursuite dans l'espace opérationnel après application de la commande par couple calculé.....	78
Figure III.31 Résultats de poursuite pour la commande PID.....	81
Figure III.32 Erreur de poursuite pour la commande PID.....	81
Figure III.33 Résultats de poursuite pour la commande PID après ajout du terme Feedforward.....	82
Figure III.34 Erreur de poursuite pour la commande PID après ajout de l'action Feedforward.....	82
Figure III.35 Comparaison des résultats de poursuite (sortie1) entre la commande par couple calculé et celle de la géométrie différentielle.....	83
Figure III.36 Comparaison des erreurs de poursuite (sortie1) entre la commande par couple calculé et le retour d'état avec référence modifiée.....	84
Figure III.37 Comparaison entre les commandes (sortie1) par couple calculé et par retour d'état avec référence modifiée.....	85
Figure IV.1 Tâche1: Pick and place.....	88
Figure IV.2 Configurations initiale, intermédiaire et finale du robot SCARA exécutant la tâche Pick and place.....	89
Figure IV.3 Trajectoire opérationnelle dans l'espace 3D.....	90
Figure IV.4 Segmentation du mot ENP.....	91

Figure IV.5 <i>Trajectoires opérationnelles empruntées par l'outil.....</i>	92
Figure IV.6 <i>Différents segments avec leur abscisse curviligne correspondante.</i>	93
Figure IV.7 <i>Projection de la trajectoire opérationnelle sur le plan d'écriture horizontal.....</i>	95
Figure IV.8 <i>Position de l'outil par rapport au plan d'écriture.....</i>	96
Figure IV.9 <i>Trajectoire opérationnelle de l'outil dans l'espace 3D.....</i>	96
Figure IV.10 <i>Droites à programmer afin d'exécuter toutes les lettres de l'alphabet.....</i>	98
Figure IV.11 <i>Ecriture de la lettre A.....</i>	98
Figure IV.12 <i>Cartes constituant l'UMAC.....</i>	101

LISTE DES TABLEAUX

Tableau I.1 <i>Classification des robots manipulateurs.....</i>	7
Tableau I.2 <i>Moyens disponibles et objectifs fixés.</i>	8
Tableau I.3 <i>Amélioration de la structure existante.....</i>	9
Tableau I.4 <i>Conception et réalisation d'un nouveau robot.....</i>	9
Tableau I.5 <i>Caractéristiques du moteur GM9236.....</i>	14
Tableau II.1. <i>Paramètres géométriques du robot.....</i>	23
Tableau IV.1 <i>Positionnement des différents tronçons par rapport au plan d'écriture.....</i>	92
Tableau IV.2 <i>Programmation de l'alphabet.....</i>	100
Tableau IV.3 <i>Programmation des chiffres.....</i>	100

NOMENCLATURE

I. Notations

$b(\dot{q})$	Fonction non linéaire qui modélise les forces de frottement
B	Vecteur qui caractérise les frottements
$C(q, \dot{q})$	Matrice des forces centrifuges et de Coriolis
Δc^k	Les coordonnées du centre de masse du corps k exprimé dans le repère R^k
D_k	La matrice d'inertie de dimension (3x3) du corps k autour de son centre de masse exprimée dans le repère absolu R_0
\bar{D}_k	Le tenseur d'inertie du corps k par rapport à son centre de masse exprimé dans R^k
F_i	Force généralisée agissant sur la $i^{\text{ème}}$ articulation
\dot{f}	Dynamique augmentée dans la méthode de l'intégration de l'erreur
g_0	Accélération gravitationnelle
$g(q)$	Forces gravitationnelles.
H_1	Matrice de passage des coordonnées homogènes aux coordonnées physiques
I	Variable d'accès au registre de contrôle et de configuration de l'interface pour les applications désirées
i_a	Courant de l'induit (en Ampères).
$J(q)$	Matrice jacobienne (THE MANIPULATOR JACOBIAN)
J_m	Moment d'inertie du moteur
J_a	Moment d'inertie de l'actionneur
J_g	Moment d'inertie du réducteur de vitesse
K	Matrice de retour d'état
K_b	Constante de la force contre électromotrice induite (en $V \cdot s / \text{rad}$)
K_m	Constante du couple (en $N.m / A$)
L	Inductance de l'induit (en Henri)
L_f	Opérateur de dérivée de Lie
$M(q)$	Matrice d'inertie
M_d	Matrice homogène désirée
m_i	Masse du corps i du robot
M	Variables de l'UMAC permettant de pointer des adresses mémoire
M_c	Matrice de Commandabilité
N	Filtre précompensateur
P	Variables de l'UMAC utilisées pour le traitement de données

\dot{p}	Vitesse instantanée de translation
ρ	Densité ou masse volumique du corps
$\phi(x)$	Difféomorphisme
$\dot{\phi}$	Vitesse instantanée de rotation
ϕ	Le flux magnétique généré par l'inducteur (en Weber)
q	Vecteur articulaire
Q	Variables de l'UMAC servant d'arguments pour les fonctions
q_k^i	Valeur initiale de la $k^{\text{ème}}$ variable articulaire
q^i	Configuration articulaire initiale
q^f	Configuration articulaire finale
R	La résistance de l'induit (en Ω)
$r(t)$	Fonction d'interpolation
\mathfrak{R}_q	Espace des variables articulaires
\mathfrak{R}_x	Espace des variables opérationnelles
\mathfrak{R}^n	Espace articulaire
\mathfrak{R}^6	Espace Opérationnel
s, n, a	Cosinus directeurs
S	Surface de glissement
$SO(3)$	Espace de représentation des orientations
T	Energie cinétique
T_s	Durée de simulation
τ_m	Couple généré par le moteur (en N.m)
τ_l	Couple de charge (en N.m)
T_{base}^{outil}	Matrice de transformation homogène représentant le modèle géométrique direct
$\theta_m(t)$	Position angulaire relative du rotor dans le domaine temporel (en radians)
$\Theta_m(s)$	La position angulaire relative du rotor dans l'espace de transfert (domaine de Laplace))
U	Energie potentielle
u	Vecteur de commande (vecteur couples)
v	Signal de commande dans l'espace linéarisé
V	Matrice jacobienne (THE TOOL CONFIGURATION JACOBIAN MATRIX)
\bar{v}^k	Vitesse de translation du centre de masse c par rapport à R_0 le repère de base.
$V(t)$	Tension d'alimentation (en Volt)
V_b	Force contre électromotrice induite (en Volt)
V_1	Fonction candidate de Lyapunov
w	Vecteur opérationnel
ω	Pulsation propre du système
$\bar{\omega}^k$	Vitesse de rotation autour du centre de masse c par rapport à R_0 le repère de base
x	Vecteur d'état
y	Sortie du système
Z	Vecteur d'état du système linéarisé

II. Abréviations

ALGAL	A lgérienne de l' A luminium
AMP-2	Carte d'amplification de l'UMAC
CEI	C ommission E lectrotechnique I nternationale
CPU	Unité de commande
CNA	C onversion N umérique/ A nalogique
D-H	D enavit- H artenberg
DSPGATE	Carte d'interface entre le CPU et les signaux d'E/S pour les moteurs
E/S	E ntées / S orties
IOGATE	Carte d'adaptation des signaux d'E/S digitaux
ISO	O rganisation I nternationale de N ormalisation
MIMO	M ultivariable I nterface M ultivariable O utput
MCD	M odèle C inématique D irect
MCI	M odèle C inématique I nverse
MGD	M odèle G éométrique D irect
MGI	M odèle G éométrique I nverse
MD	M odèle D ynamique
MLI	M odulation de L argeur d' I mpulsion
PD	Correcteur P roportionnel D érivé
PID	Correcteur P roportionnel I ntégral D érivé
PMAC 2-3U CPU	Carte contenant le DSP MOTOROLA 56303. Elle gère le système et contrôle les moteurs
PVT	P osition – V elocity – T ime
RTL	R oulis- T angage- L acet
SCARA	S elective C ompliance A rticulated R obot for A ssembly
SISO	S ingle I nterface S ingle O utput
<i>sign</i>	Fonction S igne
SNVI	S ociété N ationale des V éhicules I ndustriels
TIG	T ungstène I nerte G az
UBUS	Bus de l'UMAC
UMAC	U niversal M otion and A utomation C ontroller

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Conçus à l'origine afin de réaliser des tâches pénibles et dangereuses pour l'homme, les robots ont largement été utilisés en industrie afin d'augmenter la productivité (soudage, peinture, assemblage...etc.). A moyen terme, ils tendent à investir de plus en plus notre vie quotidienne.

De ce fait, la robotique constitue un marché porteur et très lucratif. A l'avenir, les experts annoncent une forte croissance du nombre de robots notamment dans le domaine de la robotique de service, l'assistance aux personnes handicapées, la robotique médicale ou encore pour les loisirs.

En Algérie comme dans le monde, le champ d'application des robots est immense. Il est donc clair que l'orientation de projets de recherche vers ce domaine est nécessaire, et constitue une stratégie de développement porteuse.

Depuis quelques années déjà, des projets de conception ont été menés au niveau du Laboratoire de Commande des Processus dans le but de réaliser des robots fiables.

L'objet de notre projet est la conception, la réalisation et la commande d'un robot SCARA.

PLAN DU PROJET

Le problème abordé au cours de ce travail est celui de la conception, de la réalisation puis de la commande d'un robot manipulateur à trois degrés de liberté.

Le chapitre I est consacré à la conception du robot et à sa réalisation. Nous justifierons nos choix de conception en mettant en évidence leur importance lors de l'étape de réalisation. Les diverses phases de celle-ci seront aussi détaillées.

Une fois le robot conçu et réalisé, le calcul de ses différents modèles mathématiques est indispensable. C'est l'objet du chapitre II où seront élaborés les modèles géométriques, cinématiques puis dynamique du robot. De même que ceux des actionneurs qui sont des moteurs à courant continu.

Le chapitre III quant à lui, traite de la stratégie de commande employée. Nos résultats de simulation seront exposés et analysés.

La planification de différentes tâches et leur implémentation sont abordées au chapitre IV qui constitue la suite logique des trois chapitres précédents. Nous mettrons l'accent sur les tâches que nous avons planifiées et donnerons un aperçu de l'interface utilisée pour l'implémentation des programmes de mouvement.

CHAPITRE I

CONCEPTION ET
REALISATION

CHAPITRE I

CONCEPTION ET REALISATION

I.1 INTRODUCTION

Réunir la conception et la réalisation dans un seul chapitre n'est pas le fruit du hasard. La conception de n'importe quel système –et qui est un robot dans notre cas- est faite dans le but de le réaliser. Elle doit donc impérativement tenir compte des contraintes du terrain et des moyens disponibles au risque de voir tout le processus de réalisation échouer.

Une fois réalisé, le robot en question sera mis en service et devra effectuer diverses tâches et qui sont généralement très laborieuses. La durée de vie du robot devra donc être la plus longue possible pour pouvoir amortir son coût de revient. Les étapes de conception et de réalisation dont il sera question dans ce chapitre, sont de ce fait très importantes et doivent être menées de façon à minimiser les défauts de la structure et à optimiser ses performances.

I.2 GENERALITES SUR LES ROBOTS MANIPULATEURS

I.2.1 DEFINITIONS DE BASE [16], [6]

La robotique étant une science à part entière, on ne peut traiter un sujet dans le domaine sans définir au préalable quelques notions de base :

Robot : C'est un système mécanique poly-articulé mû par des actionneurs et commandé par un ordinateur qui est destiné à effectuer une grande variété de tâches.

Articulation : Son rôle est de lier deux corps en limitant le nombre de degrés de liberté de l'un par rapport à l'autre. Le nombre de degré de liberté résultant m est appelé mobilité du robot. Il existe deux types d'articulations ; prismatique notée P et correspondant un à une translation et rotoïde qui correspond à une rotation, elle est notée R .

Espace articulaire : C'est l'espace dans lequel est représentée la situation de tous les corps du robot. Il est spécifié par l'espace \mathfrak{R}_q des variables articulaires qui est de dimension n . Il représente le nombre de degré de liberté de la structure mécanique.

Espace opérationnel : C'est l'espace dans lequel est définie la situation de l'outil. Il est spécifié par $\mathfrak{R}_x - \mathfrak{R}_3$ pour la position et $SO(3)$ pour l'orientation. Il est de dimension M , c'est le nombre de degré de liberté maximum que peut avoir l'outil.

Redondance : Elle est obtenue lorsque le nombre de degrés de liberté de l'outil est inférieur à celui de l'espace articulaire (nombre d'articulations motorisées). Elle permet d'augmenter le volume accessible et de préserver les capacités de déplacement de l'outil en présence d'obstacles.

Configuration singulière : Elle concerne les robots redondants ou pas. Elle est obtenue lorsque le nombre de degrés de liberté de l'outil est inférieur à la dimension de l'espace opérationnel.

I.2.2 CARACTERISATION D'UN ROBOT [9]

L'intérêt d'une caractérisation est de mettre à la disposition des différents intervenants dans le domaine de la robotique –industriels, chefs d'entreprises, ingénieurs...- un moyen efficace de comparaison des performances de plusieurs manipulateurs et de leur permettre ainsi d'effectuer le meilleur choix possible du robot.

Cependant, la versatilité (flexibilité d'utilisation), l'auto adaptabilité à l'environnement (capacité d'initiative) ou encore la complexité technologique des robots rendent leur caractérisation très difficile. Des travaux de normalisation (à l'image du groupe de travail TC 184/SC) développés par les deux principaux organismes de normalisation à l'échelle internationale l'ISO (Organisation Internationale de Normalisation) et la CEI (Commission Electrotechnique Internationale) ont été effectués.

Le but essentiel des ces travaux est de caractériser un robot de façon intrinsèque et ce indépendamment des applications. Plusieurs caractéristiques permettent de réaliser cet objectif :

La morphologie : Elles sont nombreuses et dépendent de la combinaison de deux paramètres ; le type d'articulation P ou R , et les angles que font deux axes successifs du robot (0 ou 90°).

L'espace de travail : appelé aussi *volume accessible*. C'est l'ensemble des points auxquels peut accéder l'organe terminal du robot.

Les performances (Précision et répétabilité) :

Précision : Les imprécisions ($>1\text{mm}$) sur le positionnement absolu sont dues essentiellement aux erreurs dans le modèle géométrique, aux erreurs de quantification de la mesure de position ou encore aux flexibilités.

Répétabilité : C'est l'erreur maximale de positionnement répété de l'outil en tout point de son espace de travail. Elle est en général inférieure à 1mm .

Les vitesses et accélérations max : Elles dépendent de la qualité des actionneurs utilisés. Il existe deux vitesses : la vitesse maximale de translation ou de rotation de chaque axe. Les constructeurs donnent souvent une vitesse maximale de l'organe terminal. L'accélération maximale est donnée pour chaque axe dans sa configuration la plus défavorable (charge maximale, inertie maximale). Elle dépend fortement de l'inertie et donc de la position du robot.

La résolution : Elle est spécifiée par la norme ISO 9283.

La charge utile : La charge utile est la charge maximale que peut porter le robot sans dégrader sa répétabilité. Elle est différente de la charge maximale que peut porter le robot qui dépend directement des actionneurs.

Les méthodes et les essais à effectuer afin de déterminer ces caractéristiques sont décrits de manière détaillée dans des normes spécifiques. Pour notre part, nous ne pouvons dans le cadre de ce travail faire une caractérisation du robot. D'une part, ceci n'est pas l'objet du projet. D'autre part, une caractérisation nécessite l'achat de normes et l'utilisation d'appareils de mesure perfectionnés et dont nous ne disposons malheureusement pas.

I.2.3 CLASSIFICATION DES ROBOTS [16]

Une classification se fait toujours suivant un paramètre. Pour les robots manipulateurs, trois paramètres sont généralement pris en considération : la géométrie de l'espace de travail, le type d'énergie des actionneurs mis en jeu et la stratégie de commande employée. Le tableau I.1 résume les classifications correspondantes.

Paramètre de classification	Classe	
	Morphologie	Désignation
Géométrie de l'espace de travail	RRP cylindrique	SCARA (Selective Compliance Articulated Robot for Assembly).
	RPP	cylindrique
	RRP	sphérique
	PPP axes perpendiculaires 2 à 2	cartésien
	RRR	anthropomorphe
	Architecture de la chaîne cinématique	Série (ou ouverte simple), fermée, arborescente ou encore parallèle.
Type d'énergie, des actionneurs	Electrique, pneumatique ou hydraulique	
Stratégie de commande	Commande en boucle ouverte ou en boucle fermée	

Tableau I.1 Classification des robots manipulateurs

I.3 DEMARCHE ADOPTEE ET FEUILLE DE ROUTE

Tout processus de conception et de réalisation –et en particulier pour les robots manipulateurs- se fait à partir d'un cahier des charges. Celui-ci spécifie les caractéristiques essentielles que le robot doit vérifier. Elles concernent notamment :

- L'espace de travail.
- Le nombre de degrés de liberté du robot.
- La morphologie.
- Le type d'énergie utilisée (électrique, pneumatique, hydraulique) ainsi que le type d'actionneurs mis en jeu.
- Les tâches que doit effectuer le robot.
- La méthode de commande (en boucle ouverte/en boucle fermée).

L'élaboration d'un cahier des charges est donc indispensable car il est à la base des processus de conception, de réalisation, de modélisation et de commande du robot. Il doit être fait de façon à assurer les meilleures caractéristiques au manipulateur. En l'absence d'un cahier des charges pour la réalisation du robot SCARA, nous proposons plutôt une feuille de route qui spécifie les exigences essentielles que le robot doit satisfaire, et ce au regard des moyens dont nous disposons et qui sont :

- Trois moteurs à courant continu. Ces moteurs qui sont munis d'encodeurs n'ont cependant pas de freins.
- Une structure existante qui a déjà été entamée l'année dernière dans le cadre d'un projet de fin d'étude (voir [17]). A noter que cette structure présente de nombreux défauts -jeux mécaniques importants, frottements très élevés du fait de l'absence de roulements- ce qui rend la commande de ce robot très difficile.
- Des moyens d'usinage au niveau de l'atelier de mécanique très limités, ce qui requiert probablement l'usinage à l'extérieur d'où la nécessité du financement de la part de l'école, ou encore une collaboration avec l'industrie mécanique.

Reste à énoncer de manière explicite les objectifs que nous nous assignons. Il s'agit de:

« Réaliser un robot rigide à trois degrés de liberté de morphologie simple, constitué de pièces faciles à usiner, fiable du point de vue mécanique, c'est-à-dire que les frottements doivent être négligeables et les jeux mécaniques limités »

Nous résumons moyens et objectifs dans le tableau récapitulatif I.2 :

Moyens	Objectifs
<ul style="list-style-type: none"> - Trois moteurs à courant continu (avec encodeurs mais sans freins). - Structure existante mais présentant de nombreuses imperfections. - Moyens d'usinage limités. 	<ul style="list-style-type: none"> - Robot rigide à 3 degrés de liberté. - Facile à usiner (pièces simples et morphologie adéquate). - Mécaniquement fiable.

Tableau I.2 Moyens disponibles et objectifs fixés.

Partant du tableau I. 2, deux alternatives se présentent à nous :

- *Solution1* : Amélioration de la structure existante.
- *Solution2* : Conception et réalisation d'un nouveau robot.

Les avantages et inconvénients de chacune des deux solutions sont explicités dans les tableaux I. 3 et I. 4 respectivement :

Solution 1 : Amélioration de la structure existante	
Avantages	Inconvénients
<ul style="list-style-type: none"> - Charge de travail a priori moins importante (conception déjà faite). - On se consacrera essentiellement à la commande. 	<p>Ne pas faire la conception veut dire :</p> <ul style="list-style-type: none"> - Ne pas se heurter aux difficultés de conception et de réalisation. - Accepter les choix antécédents. - Faire avec les défauts existants (jeux mécaniques, frottements absence de freins) ou essayer de les résorber ce qui est délicat (freins, roulements). - Difficultés de commande.

Tableau I.3 Amélioration de la structure existante

Solution 2 : Conception et réalisation d'un nouveau robot	
Avantages	Inconvénients
<ul style="list-style-type: none"> - Décider de choix de conception ce qui signifie : * Eviter les contraintes liées à la première structure. * Faire un choix de morphologie qui ne nécessite pas l'emploi de freins. (SCARA) * Eliminer les frottements sans devoir passer par les roulements. 	<ul style="list-style-type: none"> - Difficultés de réalisation car : * L'usinage des pièces devra se faire à l'extérieur de l'école. * Le matériau utilisé à savoir l'Aluminium plein est difficile à trouver. Il est soit importé par les industriels, soit commandé suivant besoin à l'entreprise ALGAL (Algérienne de l'Aluminium) de Oued Smar et ce à de grandes quantités.

Tableau I.4 Conception et réalisation d'un nouveau robot

En pesant les pour et les contre de chacune des deux solutions, nous avons choisi la seconde, car nous jugeons qu'elle correspond le mieux aux objectifs fixés auparavant.

Finalement, la feuille de route proposée est la suivante :

- 1- Robot à 3 degrés de liberté (car nous disposons de 3 moteurs).
- 2- Les pièces à usiner doivent être les plus simples possibles.
- 3- Les matériaux utilisés pour l'usinage doivent présenter deux caractéristiques essentielles et qui sont la rigidité et la légèreté.
- 4- L'arrêt de la structure doit se faire par programmation lors de la génération de mouvement.
- 5- Le robot une fois arrêté ne doit pas nécessiter l'emploi de freins pour être maintenu à sa position.
- 6- Les frottements doivent être diminuées au maximum sans pour autant utiliser des roulements.

I.4 CONCEPTION DU ROBOT

La conception du robot est une phase très importante. Elle doit se faire de façon à répondre aux spécificités de la feuille de route adoptée tout en assurant le minimum de contraintes aux phases qui en découleront - réalisation, modélisation et commande. Il est donc clair que le succès de celles-ci reposera sur la réussite de la phase de conception.

La première difficulté à laquelle nous nous sommes heurtés a été l'absence de documentation détaillée –les fabricants de robots ne dévoilant pas leurs techniques à cause de la concurrence. Aussi, le faible degré de pénétration des robots dans l'industrie en Algérie constitue un sérieux désavantage. En effet, les pièces sont quasiment inexistantes sur le marché (effecteurs, moteurs, articulations) de même pour les matières premières.

Ceci rend la phase de conception d'autant plus délicate, car en plus de respecter les dispositions de la feuille de route, elle doit prendre en compte les considérations du terrain.

Dans cette section, nous expliquerons la démarche de conception adoptée. Nous détaillerons les solutions envisagées, et nous justifierons la structure finale à laquelle nous avons abouti.

I.4.1 STRUCTURES PROPOSEES ET STRUCTURE FINALE

Avant d'aboutir à la structure finale, plusieurs propositions ont été faites. Nous en présentons quelques unes dans la figure I.1 :

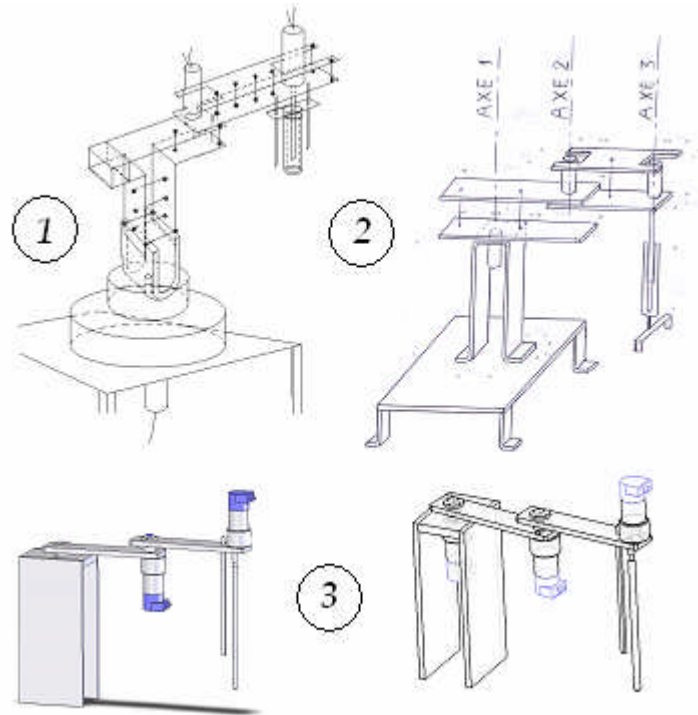


Figure I.1 Structures proposées et structure finale.

Les trois structures proposées sont à trois degrés de liberté et de type SCARA. Ce choix est important et correspond parfaitement à notre feuille de route. En effet, cette structure conserve sa position une fois arrêtée et ce en l'absence de freins. Ceci est dû au fait que les trois axes sont verticaux et donc parallèles à la force gravitationnelle.

En premier lieu, nous avons proposé les deux premières structures présentées. Elles sont identiques sauf que pour la deuxième, le premier moteur a été déplacé vers le haut de façon à réduire la charge qu'il entraîne (le poids du premier corps étant moins important).

Les plaques d'aluminium utilisées sont de faible épaisseur. Ce choix est fait en fonction de ce qui est disponible sur le marché. Les corps sont formés de deux plaques parallèles assemblées par des boulons.

Pour l'articulation prismatique, nous avons opté pour la solution vis sans fin (ou vis télescopique). La rotation de l'arbre du moteur engendre la translation d'un organe le long de

la vis. Cet organe est un écrou relié à deux guides parallèles à la tige filetée. Leur rôle est de limiter son mouvement à un seul degré de liberté (une translation en l'occurrence).

A noter que le système vis/écrou avec guide a été récupéré à partir d'un rhéostat défectueux du laboratoire de haute tension. Le pas de la tige filetée est de 20mm.

Après analyse de ces deux premières structures par un ingénieur en mécanique, nous avons décidé d'apporter quelques améliorations. Les plaques d'aluminium de faible épaisseur ont été remplacées par de l'aluminium plein, beaucoup plus rigide mais difficile à trouver sur le marché. De même que l'emplacement du deuxième moteur qui a été modifié pour accroître la rigidité du robot.

I.4.2 PRESENTATION DES PIÈCES DU ROBOT

La conception et le dimensionnement des pièces du robot ont été faits à l'aide de la suite logicielle Solidworks. Elle constitue un outil de CAO parfaitement adapté à ce type de structure.

En plus du dimensionnement précis des pièces du robot, il est possible de réaliser leur assemblage et de simuler les mouvements des axes les uns par rapport aux autres. Les plans de coupe peuvent aussi être générés. Le logiciel E-Drawings facilite l'analyse de la structure grâce aux différentes vues qui sont proposées (éclatée, ombrée, ...). D'autres fonctionnalités plus développées –que nous n'avons pas utilisées– peuvent aussi être utilisées comme le calcul automatique des moments d'inertie ou encore la génération d'animations pour le robot.

Une vue éclatée des différentes pièces constituant notre robot est donnée par la figure I.2 :

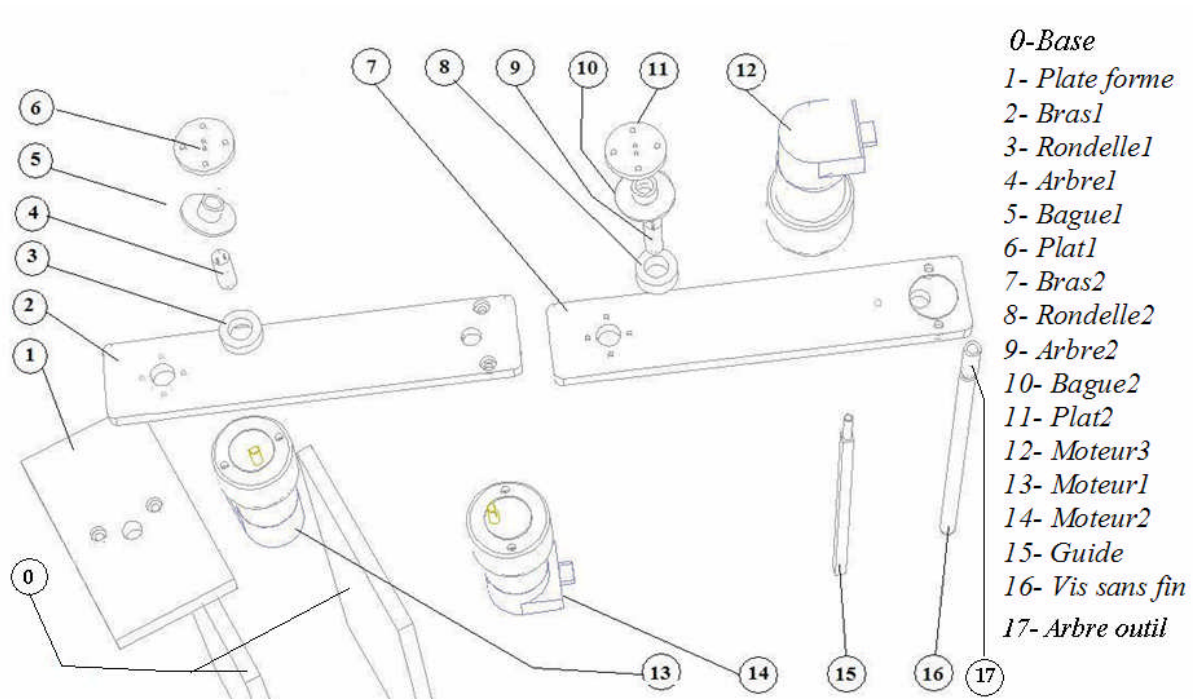


Figure I.2 Nomenclature des pièces du robot

Les plans de coupe des pièces du robot ainsi que les matériaux utilisés pour leur réalisation sont donnés en annexe A. 1.

I.4.3 PRESENTATION DES ACTIONNEURS UTILISES

Les actionneurs que nous utilisons sont des moteurs à courant continu à aimants permanents munis d'encodeurs incrémentaux optiques ayant une résolution de 500CPR (compte par tour). Ce sont les moteurs de référence GM9236 du fabricant américain Pittman.



Figure I.3 Différents moteurs Pittman

Les caractéristiques des moteurs que nous avons utilisées sont résumées dans le tableau suivant :

Paramètre	Notation	Unité	Valeur
Constante du couple	K_i	$N.m$	$48.8 \cdot 10^{-3}$
Constante de la force contre électromotrice	K_b	$V.s / rad$	$48.8 \cdot 10^{-3}$
Inertie du moteur	J_M	$Kg.m^2$	$7.06 \cdot 10^{-6}$
Constante de temps électrique	τ_E	ms	1.06
Constante de temps mécanique	τ_M	ms	8.5
Masse du moteur	W_M	gm	391.2
Résistance	R_a	Ω	2.49
Inductance	L_a	mH	2.63
Tension d'alimentation	V_a	$Volts$	24
Rapport de réduction	M	-	65.5

Tableau I.5 caractéristiques du moteur GM9236

Différentes vues du moteur avec l'encodeur sont données par la figure I. 4 :

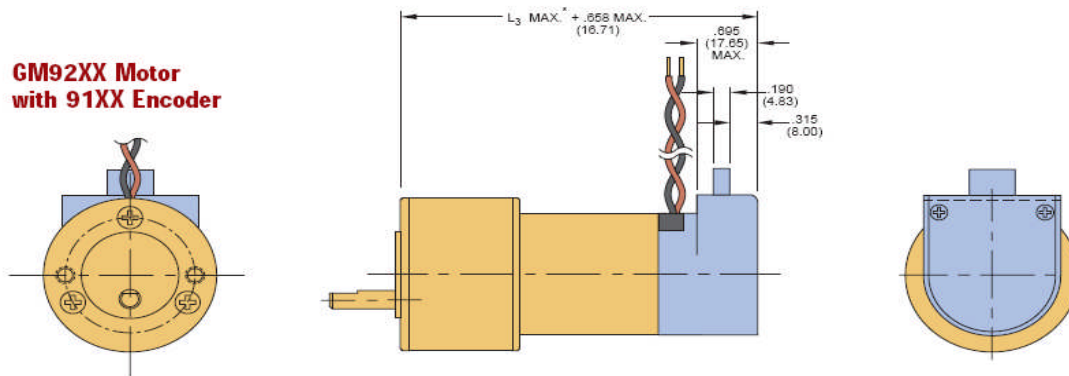


Figure I.4 Différentes vues du moteur GM9236 avec encodeur

I.5 REALISATION DU ROBOT

La réalisation du robot a été effectuée au cours d'un stage pratique d'une durée effective de huit jours (Samedi 29 Avril au Mardi 9 Mai 2006) au niveau de la division fonderie de la SNVI (Société Nationale des Véhicules Industriels) située dans la zone industrielle de Rouiba.



Figure I.5 Division Fonderie de la SNVI à Rouiba

I.5.1 BREF HISTORIQUE DE LA SNVI [15]

De 1957 à 1966 : Implantation de la société française **BERLIET** sur le territoire Algérien par la construction en juin 1957 d'une usine de montage de véhicules "poids lourd" à 30 km de l'est d'Alger, plus exactement à Rouïba.

De 1967 à 1980 : En 1967, fut créée la **SONACOME (SO**ciété **N**ationale de **CO**nstruction **ME**canique) le schéma d'organisation adopté pour la **SO.NA.CO.ME** regroupant en son sein dix (10) entreprises autonomes.

De 1981 à 1994 : La **S.N.V.I (Entreprise Nationale de Véhicules Industriels)** devient une entreprise publique socialiste (**EPS**). La **S.N.V.I** est née à l'issue de la restructuration de la **SO.NA.CO.ME** et le décret de sa création lui consacra un statut d'entreprise socialiste à caractère économique régis par les principes directifs de la **Gestion Socialiste des Entreprises (G.S.E)**.

De 1995 à ce jour : Le mois de **Mai 1995**, la **S.N.V.I** change de statut juridique pour devenir une **Entreprise Publique Économique** régie par le droit commun : la **S.N.V.I** est alors érigée en **Société Par Action (SPA)**, au capital social de **2,2 milliards de Dinars**. La **S.N.V.I** devenue groupe industriel.

I.5.2 ORGANISATION DU TRAVAIL

Les opérations d'usinage ont eu lieu dans deux ateliers de la fonderie :

- L'atelier aluminium où nous avons effectué les opérations de découpage, soudage, moulage, meulage.
- L'atelier central qui est lui-même divisé en deux ateliers ; « le modelage » où a eu lieu la plus grande partie du travail sous la conduite d'un technicien supérieur en mécanique, et l'atelier « maintenance et réparation » où ont été effectuées les opérations de tournage.

Les matières premières utilisées, qui ont été généreusement fournies par la SNVI, sont :

- 1 feuille d'Aluminium de 16mm d'épaisseur nécessaire à la réalisation de la base. (Du magasin car indisponible en atelier)
- 1 feuille d'Aluminium d'épaisseur 10 mm nécessaire à la réalisation des bras (disponible en atelier).
- Des cylindres pleins d'Aluminium et de Bronze pour la réalisation des arbres, plats, rondelles et bagues. (disponibles en atelier).

A noter que l'Aluminium utilisé est un alliage importé de France dont la référence est : AS10G. Il contient :

90% d'aluminium (Al), 10% de Silicium (Si) et du Magnésium (Mg) entre 0.17 et 0.34%.

Les caractéristiques de l'aluminium seul ne sont pas très intéressantes. L'ajout d'autres matériaux permet d'améliorer nettement ses propriétés (Rigidité, fluage, résilience, traction ...etc.).

I.5.3 RETOUCHES ET AMELIORATIONS

Une conception aussi parfaite soit-elle présente toujours des insuffisances. Ce n'est que lors de la réalisation que celles-ci peuvent être décelées et corrigées.

Le savoir-faire, l'assistance et les conseils de techniciens expérimentés nous ont permis d'apporter des améliorations auxquelles nous n'avions pas pensé lors de la conception, et de

concrétiser des solutions envisagées auparavant. Les principales modifications et améliorations sont donc :

- Dimensions des arbres :

Ceux-ci ont subi un agrandissement de leurs diamètres (14mm au lieu de 10.34), car ils étaient insuffisants pour recevoir les deux vis d'assemblage de diamètre M1 (par ailleurs indisponibles sur le marché), ainsi qu'une vis de pression (ou de blocage) de diamètre suffisamment grand pour une bonne fixation de l'arbre (Les jeux étaient trop importants auparavant).

Nous avons aussi préféré la solution vis/goupille à celle des deux vis trop encombrante et risquée compte tenu du diamètre des arbres.

- Contre poids :

Le centre de gravité de la structure étant excentré, celle-ci a tendance à basculer et les corps du robot fléchissent, ce qui conduit à l'arrachage des arbres et à des efforts de cisaillement très élevés sur l'ensemble vis/goupille (celle-ci joue d'ailleurs un rôle de sécurité en cas de charge importante).

De ce fait, le contre poids déjà envisagé lors d'une première solution de conception s'avère donc indispensable pour l'équilibre de la structure (centrage du centre de masse).

- Billes de maintien :

Le contre poids joue un rôle important dans l'équilibre de la structure mais introduit des contraintes dans le calcul du modèle dynamique. Ceci est dû à sa grande inertie et à sa forme géométrique complexe.

Nous avons donc pensé –avec l'aide précieuse des techniciens de la SNVI- à explorer d'autres solutions d'équilibrage qui seraient moins contraignantes et encombrantes.

La solution privilégiée est l'ajout de billes de maintien. Celles-ci permettent de garder les deux premiers corps du robot en permanence sur un plan horizontal, sans pour autant modifier sensiblement le robot. Pour la seconde articulation, nous avons dû usiner un disque servant de plateforme pour la seconde bille qui roule dessus. Les billes introduisent néanmoins des frottements mais qui sont négligeables car résultant de contacts ponctuels (principe du roulement), en plus du fait que les points de contact sont graissés.

- Taraudage des moteurs :

Les moteurs dont nous disposons étant d'origine américaine, nous avons été contraints de refaire leurs taraudages selon la norme européenne vu la disponibilité des vis correspondantes à cette dernière sur le marché.

- Goupillage des arbres et des moteurs :

Une fois le robot monté, nous avons constaté que les deux premiers arbres avaient tendance à s'arracher. De ce fait, l'agrandissement des vis de blocage envisagé en premier lieu s'est avéré insuffisant pour la fixation des arbres. Nous avons remédié à ce problème en optant pour le goupillage des arbres avec ceux des moteurs, ce qui permet une fixation définitive et fiable.

Finalement, et après l'assemblage des pièces, nous obtenons la structure finale du robot qui apparaît sur la photo suivante :

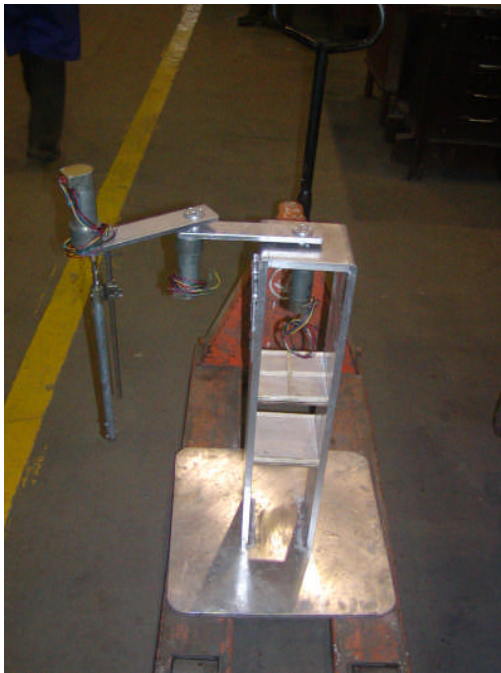


Figure I.6 *Vue générale du robot*

I.5 CONCLUSION

Ce chapitre nous a permis d'expliquer de façon détaillée la stratégie que nous avons suivie afin de réaliser le robot SCARA. La première étape a été de définir une feuille de route à partir de laquelle nous avons proposé la structure SCARA et dimensionné ses pièces. La morphologie SCARA nous a permis de palier à l'absence de freins. Les plans des pièces

donnés en annexe A.1 mettent en évidence leur simplicité. Celle-ci nous a permis de réaliser le robot dans des délais raisonnables ce qui est en fait l'une de nos priorités de conception.

Le stage pratique de réalisation nous a permis d'être au contact de professionnels. A l'aide de leurs conseils, d'autres solutions simples et efficaces ont pu être apportées. Elles nous ont permis d'améliorer sensiblement la qualité de la structure finale surtout du point de vue de la robustesse, de la fiabilité et de la précision du robot.

CHAPITRE II
MODELISATION

CHAPITRE II

MODELISATION

II.1 INTRODUCTION [2], [6]

Un modèle est une représentation de la réalité avec une complexité réduite [2]. Un modèle non représentatif avec des erreurs de modélisation élevées engendre des difficultés lors de la commande du système. En effet, la loi de commande, synthétisée sur sa base, et même si elle donne de bons résultats en simulation, peut déstabiliser voire détruire le système réel une fois appliquée.

Il apparaît donc clairement que la modélisation est l'une des étapes déterminantes - peut-être même la plus importante - lors de l'asservissement d'un système. Par conséquent, elle doit être menée avec beaucoup de minutie et d'attention car sa négligence en amont aurait pour conséquence éventuelle l'échec de tout le processus de régulation en aval.

Dans ce chapitre, nous calculons les modèles mathématiques nécessaires à la commande d'un robot. Ces modèles sont:

- Les modèles de transformation entre l'espace opérationnel (dans lequel est définie la situation de l'organe terminal) et l'espace articulaire (dans lequel est définie la configuration du robot). On distingue :
 - Les modèles géométriques direct et inverse qui expriment la situation de l'organe terminal en fonction des variables articulaires du mécanisme et inversement ;
 - Les modèles cinématiques direct et inverse qui expriment la vitesse de l'organe terminal en fonction des vitesses articulaires et inversement ;
- Les modèles dynamiques définissant les équations du mouvement du robot, qui permettent d'établir les relations entre les couples ou forces exercés par les actionneurs et les positions, vitesses et accélérations des articulations. [6].
- Les modèles des actionneurs qui, dans notre cas, sont des moteurs à courant continu à aimant permanent.

A noter que tout au long de cette modélisation, nous avons essayé de respecter deux règles fondamentales lors de la construction de nos modèles. La première est qu'il doit être le plus représentatif possible du système réel - car il traduit son fonctionnement. La seconde est la simplicité car si le modèle est trop compliqué, le système devient difficile – voire impossible - à commander.

II.2 MODELES GEOMETRIQUES

II.2.1 MODELE GEOMETRIQUE DIRECT [10], [6]

Le Modèle géométrique direct consiste à exprimer la situation de l'organe terminal en fonction des variables articulaires du robot.

La démarche que nous avons adoptée pour le calcul du modèle géométrique direct de notre robot manipulateur peut être résumée par les quatre principales étapes décrites dans la figure II.1 :

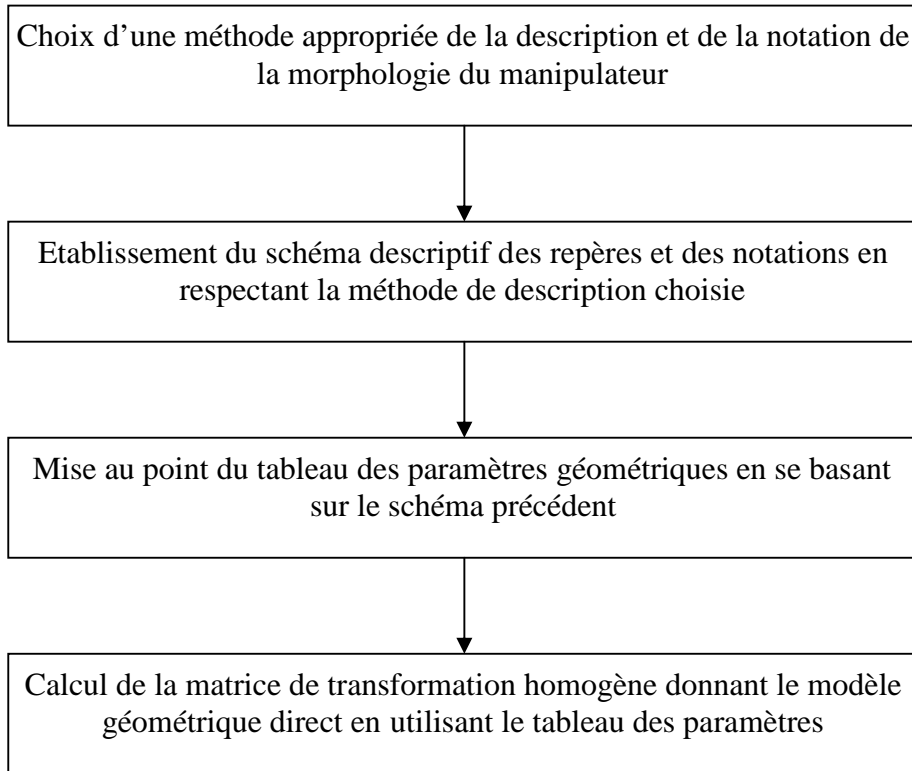


Figure II.1. *Méthodologie adoptée pour le calcul du Modèle géométrique direct.*

Appliquons donc cette même méthodologie au robot SCARA à 3 degrés de liberté

- Choix : Représentation D-H car elle très utilisée, facile à appliquer et est destinée aux chaînes cinématique ouvertes simples. Elle est donnée en annexe B. 1.
- Schéma de placement des repères et notations (Figure II.2) : (Etapas 0-7 de l'algorithme D-H) :

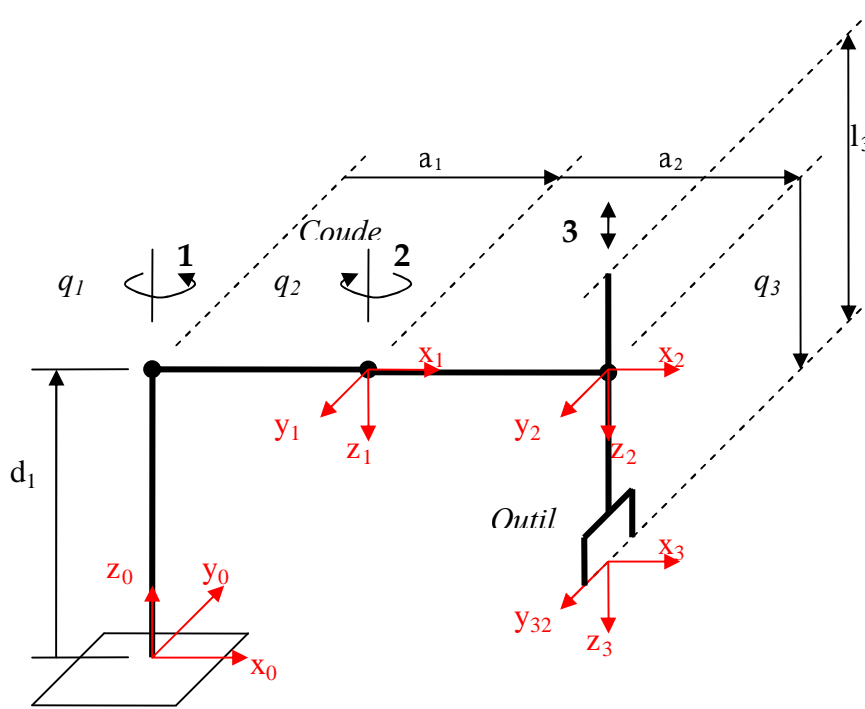


Figure II.2. Schéma des différentes articulations et repères selon la norme D-H

Remarque:

D'autres choix de repères sont possibles et peuvent être envisagés. Dans notre cas par exemple, nous avons choisi de placer le repère absolu au niveau de la base du robot car c'est sur celle-ci que reposera généralement l'objet à manipuler. L'avantage donc de cette solution est qu'elle permet de situer de façon intuitive la position relative de la pointe de l'outil par rapport à l'objet à manipuler.

- Tableau (II.1) des paramètres géométriques

Axes	ϑ	d	A	α	Repos
1	q_1	d_1	a_1	π	0
2	q_2	0	a_2	0	0
3	0	q_3	0	0	$d_3=l_3/2$

Tableau II.1. Paramètres géométriques du robot

- Calcul de T_{base}^{outil} :

$$\begin{aligned}
 T_{base}^{outil} &= \begin{pmatrix} c_1 & s_1 & 0 & a_1c_1 \\ s_1 & -c_1 & 0 & a_1s_1 \\ 0 & 0 & -1 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 T_{base}^{outil} &= \begin{pmatrix} C_{1-2} & S_{1-2} & 0 & a_1C_1 + a_2C_{1-2} \\ S_{1-2} & -C_{1-2} & 0 & a_1S_1 + a_2S_{1-2} \\ 0 & 0 & -1 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 T_{base}^{outil} &= \begin{pmatrix} C_{1-2} & S_{1-2} & 0 & a_1C_1 + a_2C_{1-2} \\ S_{1-2} & -C_{1-2} & 0 & a_1S_1 + a_2S_{1-2} \\ 0 & 0 & -1 & d_1 - q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{II.2.1}
 \end{aligned}$$

Le Modèle géométrique direct est donc donné par :

$$\left\{ \begin{array}{l} s_x = C_{1-2} \\ s_y = S_{1-2} \\ s_z = 0 \\ \\ n_x = S_{1-2} \\ n_y = -C_{1-2} \\ n_z = 0 \\ \\ a_x = 0 \\ a_y = 0 \\ a_z = -1 \\ \\ P_x = a_1C_1 + a_2C_{1-2} \\ P_y = a_1S_1 + a_2S_{1-2} \\ P_z = d_1 - q_3 \end{array} \right. \tag{II.2.2}$$

Dans cette représentation, neuf premiers paramètres représentent l'orientation alors qu'en pratique trois paramètres suffisent pour la décrire.

Plusieurs méthodes pour représenter l'orientation existent, par exemple les angles d'Euler, les angles de **Roulis-Tangage-Lacet** (RTL) ou encore les quaternions -qui sont une représentation redondante car utilisant quatre paramètres au lieu de trois.

Pour notre part, nous choisissons d'utiliser une notation particulière de l'orientation (consulter pour plus de détails [10]).

Partant de là, le vecteur des coordonnées opérationnelles est :

$$\begin{aligned}
 w(q) &= (w_1 \ w_2 \ w_3 \ w_4 \ w_5 \ w_6)^T \\
 &= \left(P_x \ P_y \ P_z \ a_x \exp\left(\frac{q_n}{\pi}\right) \ a_y \ exp\left(\frac{q_n}{\pi}\right) \ a_z \ exp\left(\frac{q_n}{\pi}\right) \right)^T \\
 &= \left(P \ a \ exp\left(\frac{q_n}{\pi}\right) \right)^T
 \end{aligned} \tag{II.2.3}$$

Ainsi le Modèle géométrique direct s'écrit sous la forme :

$$w(q) = \begin{bmatrix} a_1 C_1 + a_2 C_{1-2} \\ a_1 S_1 + a_2 S_{1-2} \\ \hline d_1 - q_3 \\ 0 \\ 0 \\ -1 \end{bmatrix} \tag{II.2.4}$$

Remarque :

A noter que dans cette représentation, nous obtenons les trois composantes de l'orientation en considérant $q_n = 0$, ceci est vrai pour $n = 4$ (quatre ddl) avec la quatrième articulation qui est rotoïde et qui sert à orienter l'outil (d'où $q_4 = 0$), articulation dont nous ne disposons pas.

Nous avons donc obtenu le modèle géométrique direct sous sa forme minimale. Cette dernière sera utilisée dans la prochaine section, où sera abordé le problème inverse à savoir le calcul du modèle géométrique inverse.

II.2.2 MODELE GEOMETRIQUE INVERSE

Le problème dans cette section consiste à calculer les coordonnées articulaires correspondant à une situation donnée de l'organe terminal.

Ce qui est montré par la figure II.3 :

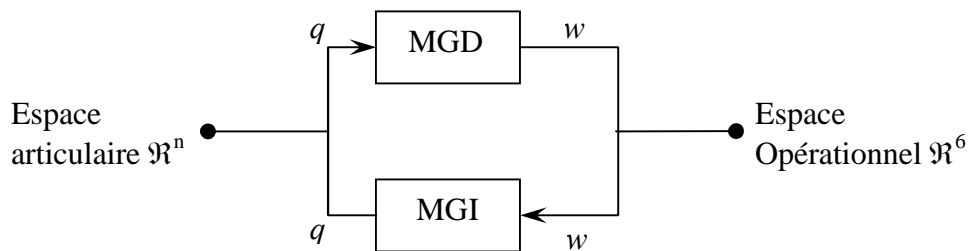


Figure II.3. *Modèle géométrique direct et inverse*

II.2.2.1 Propriétés générales des solutions [10], [6]

Bien que les solutions du modèle géométrique inverse dépendent de la classe et du type de robot utilisé, certaines propriétés des solutions sont générales et restent valables quelque soit le type de manipulateur utilisé.

Existence de solutions

Si la position désirée de l'outil se trouve en dehors de l'espace de travail, il ne peut y avoir de solutions.

De plus, même si $P \in E$, il pourrait y avoir certaines orientations pour lesquelles les variables articulaires trouvées violeraient les limites des débattements articulaires. D'ailleurs, si le robot possède moins de trois degrés de liberté pour orienter l'outil, il est clair qu'un certain nombre d'orientations serait irréalisable.

Mais afin de mieux cerner ce problème considérons une autre approche de celui-ci : La recherche de solutions pour le modèle géométrique inverse consiste en la résolution du système d'équations : $T_{base}^{outil}(q) = M_d$, telle que M_d est la matrice homogène désirée.

A priori, il s'agit de résoudre un système de douze équations algébriques non linéaires -la dernière ligne étant toujours constante- à n inconnues. Or, $R^{-1} = R^T$ et donc $R^T R = I$, nous obtenons ainsi les six contraintes suivantes :

$$\begin{cases} s \cdot n = 0 \\ s \cdot a = 0 \\ n \cdot a = 0 \end{cases} \quad (\text{II.2.5})$$

$$\|s\| = \|n\| = \|a\| = 1 \quad (\text{II.2.6})$$

Ceci revient à résoudre un système de six équations à n inconnues. Nous pouvons donc conclure qu'il faut que le nombre de variables articulaires q soit au moins égal au nombre de contraintes indépendantes pour obtenir une solution générale au problème du modèle géométrique inverse, d'où la condition nécessaire mais non suffisante : $n \geq 6$.

Unicité de la solution

L'existence de la solution n'est pas le seul problème à considérer dans le calcul du modèle géométrique inverse. Quand les solutions existent, elles ne sont généralement pas uniques. La multiplicité de solutions peut subvenir dans plusieurs cas d'ailleurs.

Par exemple lorsque $n > 6$, le robot est dit *redondant* car il possède plus de degrés de liberté que ce dont il a besoin pour amener l'outil aux configurations arbitraires désirées. Pour ce type de robots, il existe une infinité de solutions. L'intérêt de ce type de structures est leur flexibilité qui fait qu'elles soient très utilisées en environnement encombré. Ainsi, un certain nombre de degrés de liberté sert à éviter l'obstacle alors que les degrés de liberté restants servent à positionner et orienter l'outil.

L'existence d'une infinité de solutions n'est pas caractéristique des robots possédant plus de six degrés de liberté et qui sont *redondant par nature*, ce cas peut se présenter dans certaines configurations singulières, ou alors dans certains cas où un robot non redondant dans le cas général peut être redondant vis-à-vis de la tâche.

II.2.2.2 Calcul du MGI du robot SCARA

La condition ($n \geq 6$) nécessaire pour l'existence de solutions n'étant pas réalisée -notre robot ne possède que trois degrés de liberté- nous faisons le choix délibéré de fixer uniquement la position de l'outil car il est impossible de fixer à la fois la position et l'orientation (on aurait alors plus d'équations que d'inconnus).

Les variables opérationnelles désirées correspondent donc aux paramètres du vecteur position désiré à savoir : $P^d = (Px \quad Py \quad Pz)^T$, dans ce cas on a le système d'équations :

$$\begin{cases} a_1 C_1 + a_2 C_{1-2} = P_x = w_1 \\ a_1 S_1 + a_2 S_{1-2} = P_y = w_2 \\ d_1 - q_3 = P_z = w_3 \end{cases} \quad (\text{II.2.7})$$

Calcul de q_3 :

A partir de la dernière équation du système, on a :

$$q_3 = d_1 - P_z \quad (\text{II.2.8})$$

Calcul de q_2 :

A partir de la somme des deux premières équations élevées au carré, on obtient q_2 :

$$q_2 = \pm \arccos\left(\frac{P_x^2 + P_y^2 - a_1^2 - a_2^2}{2 a_1 a_2}\right) \quad (\text{II.2.9})$$

Calcul de q_1 :

Les deux premières équations du système peuvent s'écrire sous la forme suivante :

$$\begin{cases} [a_1 + a_2 C_2]C_1 + [a_2 S_2]S_1 = P_x & (1) \\ -[a_2 S_2]C_1 + [a_1 + a_2 C_2]S_1 = P_y & (2) \end{cases}$$

C'est un système de deux équations à deux inconnues dont les solutions sont :

$$S_1 = \frac{a_2 S_2 P_x + [a_1 + a_2 C_2]P_y}{[a_2 S_2]^2 + [a_1 + a_2 C_2]^2} \quad \text{Et} \quad C_1 = \frac{[a_1 + a_2 C_2]P_x - a_2 S_2 P_y}{[a_2 S_2]^2 + [a_1 + a_2 C_2]^2}$$

D'où :

$$q_1 = A \tan 2 \left[a_2 S_2 P_x + (a_1 + a_2 C_2)P_y, (a_1 + a_2 C_2)P_x - a_2 S_2 P_y \right] \quad (\text{II.2.8})$$

Finalement, les solutions analytiques du modèle géométrique inverse sont données par :

$$\begin{cases} q_3 = d_1 - P_z \\ q_2 = \pm \arccos\left(\frac{P_x^2 + P_y^2 - a_1^2 - a_2^2}{2 a_1 a_2}\right) \\ q_1 = A \tan 2 \left[a_2 S_2 P_x + (a_1 + a_2 C_2)P_y, (a_1 + a_2 C_2)P_x - a_2 S_2 P_y \right] \end{cases} \quad (\text{II.2.9})$$

II.3 MODELES CINEMATIQUES [6], [10]

II.3.1 MODELE CINEMATIQUE DIRECT

Dans ce qui suit, nous abordons le problème du calcul du modèle cinématique direct, ce modèle décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires [6].

Calcul de la matrice Jacobienne (THE TOOL CONFIGURATION JACOBIAN MATRIX) notée $V(q)$

Le calcul de cette matrice Jacobienne s'effectue en dérivant le vecteur des coordonnées opérationnelles w dont l'expression est donnée par la relation (II.2.4), par rapport aux variables articulaires q_1 , q_2 puis q_3 . Les trois dérivées obtenues constituent les trois colonnes de cette matrice, ce qui nous permet d'écrire :

$$V(q) = \begin{pmatrix} -a_1 S_1 - a_2 S_{1-2} & a_2 S_{1-2} & 0 \\ a_1 C_1 + a_{21} C_{1-2} & -a_2 C_{1-2} & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (\text{II.3.1})$$

On pourra désormais écrire : $\dot{w} = V \dot{q}$ et on aura ainsi résolu le problème du modèle cinématique direct.

II.3.2 MODELE CINEMATIQUE INVERSE

L'objectif du modèle cinématique inverse est de calculer, à partir d'une configuration q donnée, les vitesses articulaires \dot{q} qui assurent au repère terminal une vitesse opérationnelle \dot{w} imposée [6].

Pour le calcul du modèle cinématique inverse, nous utilisons la matrice V trouvée dans le calcul du modèle cinématique direct donnée par la relation (II.3.1).

Pour faciliter les calculs, nous utilisons le calcul symbolique en réécrivant la matrice V comme indiqué dans la relation (II.3.2) :

$$V = \begin{pmatrix} \alpha & \beta & 0 \\ \gamma & \delta & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (\text{II.3.2})$$

Avec :

$$\begin{cases} \alpha = -a_1 S_1 - a_2 S_{1-2} \\ \beta = a_2 S_{1-2} \\ \gamma = a_1 C_1 + a_{2_1} C_{1-2} \\ \delta = -a_2 C_{1-2} \end{cases} \quad (\text{II.3.3})$$

Et donc :

$$V^T = \begin{pmatrix} \alpha & \gamma & 0 & 0 & 0 & 0 \\ \beta & \delta & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix} \quad (\text{II.3.4})$$

Ce qui nous donne le produit matriciel :

$$V^T V = \begin{pmatrix} \alpha^2 + \gamma^2 & \alpha\beta + \gamma\delta & 0 \\ \alpha\beta + \gamma\delta & \beta^2 + \delta^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{II.3.5})$$

Et son inverse :

$$[V^T V]^{-1} = \frac{1}{\Delta} \begin{pmatrix} \beta^2 + \delta^2 & -(\alpha\beta + \gamma\delta) & 0 \\ -(\alpha\beta + \gamma\delta) & \alpha^2 + \gamma^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{II.3.6})$$

Avec :

$$\Delta = (\alpha^2 + \gamma^2)(\beta^2 + \delta^2) - (\alpha\beta + \gamma\delta)^2 \quad (\text{II.3.7})$$

Cette matrice inverse $[V^T V]^{-1}$ n'est définie que si $\Delta \neq 0$.

On aura finalement la pseudo inverse :

$$V^+ = [V^T V]^{-1} V^T = \frac{1}{\Delta} \begin{pmatrix} \delta(\alpha\delta - \beta\gamma) & \beta(\beta\gamma - \alpha\delta) & 0 & 0 & 0 & 0 \\ \gamma(\beta\gamma - \alpha\delta) & \alpha(\alpha\delta - \beta\gamma) & 0 & 0 & 0 & 0 \\ 0 & 0 & -\Delta & 0 & 0 & 0 \end{pmatrix} \quad (\text{II.3.8})$$

On peut désormais écrire :

$$\dot{q} = V^+ \dot{x} \quad (\text{II.3.9})$$

Enfin, en développant la relation (II.3.9), on obtient les équations du Modèle Cinématique Inverse :

$$\begin{cases} \dot{q}_1 = \frac{1}{\Delta} (\dot{P}_x \delta (\alpha\delta - \beta\gamma) + \dot{P}_y \beta (\beta\gamma - \alpha\delta)) \\ \dot{q}_2 = \frac{1}{\Delta} (\dot{P}_x \gamma (\beta\gamma - \alpha\delta) + \dot{P}_y \alpha (\alpha\delta - \beta\gamma)) \\ \dot{q}_3 = -\dot{P}_z \end{cases} \quad (\text{II.3.10})$$

II.4 MODELE DYNAMIQUE [6], [10]

Le modèle dynamique est la relation entre les couples (et/ou forces) appliqués aux actionneurs et les positions, vitesses et accélérations articulaires [6].

L'approche utilisée pour le calcul du modèle dynamique est celle développée par Euler Lagrange. Elle est basée sur les concepts de coordonnées généralisées (articulaires), d'énergie et de force généralisée. Son avantage est que les termes obtenus dans les équations finales du modèle dynamique ont une signification physique (inerties, frottements, gravité, forces de Coriolis et forces centrifuges).

II.4.1 EQUATIONS GENERALES DU MOUVEMENT

Une approche simple et élégante pour modéliser les dynamiques d'un manipulateur peut être obtenue en utilisant la formulation de Lagrange. Dans cette formulation, les équations générales du mouvement s'écrivent :

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_i} L(q, \dot{q}) - \frac{\partial}{\partial q_i} L(q, \dot{q}) = F_i \quad 1 \leq i \leq n \quad (\text{II.4.1})$$

Où :

$L(q, \dot{q}) = T(q, \dot{q}) - U(q)$: est le Lagrangien. Il est défini comme étant la différence entre les énergies cinétiques $T(q, \dot{q})$ et potentielles $U(q)$ du manipulateur.

F_i : est la force généralisée agissant sur la $i^{\text{ème}}$ articulation. C'est la force résiduelle obtenue quand les effets des forces gravitationnelles et inertielles sont annulés.

Sous cette formulation, les dynamiques du manipulateur sont représentées par n équations du 2^{nd} ordre en q .

Nous devons donc formuler les expressions détaillées des énergies cinétiques et potentielles T et U , ainsi que la force généralisée F et ce pour obtenir le modèle dynamique analytique de notre robot.

II.4.2 ENERGIE CINETIQUE TOTALE $T(q, \dot{q})$

L'énergie cinétique est le terme le plus difficile à déterminer pour un robot manipulateur.

Intéressons nous d'abord au mouvement du corps k du manipulateur donné par la figure 5 :

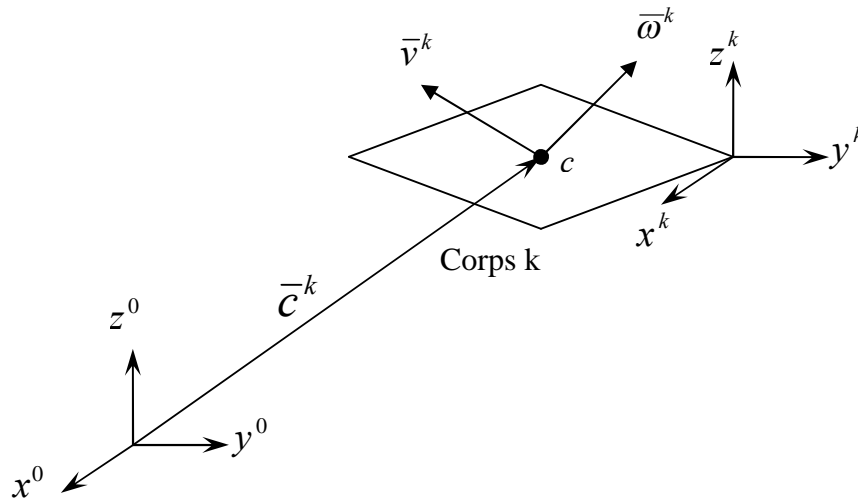


Figure II.4 Mouvement du corps k

c : est le centre de masse du corps k .

Dans l'espace tridimensionnel, le corps k se déplace avec une vitesse de translation et une vitesse de rotation. L'expression de l'énergie cinétique totale s'écrit :

$$T(q, \dot{q}) = \sum_{k=1}^n \frac{(\bar{v}^k)^T \cdot m_k \cdot \bar{v}^k + (\bar{\omega}^k)^T \cdot D_k \cdot \bar{\omega}^k}{2} \quad (\text{II.4.2})$$

Avec :

$\bar{v}^k \in \mathbb{R}^3$: Vitesse de translation de c par rapport à R0 le repère de base.

$\bar{\omega}^k \in \mathbb{R}^3$: Vitesse de rotation autour de c par rapport à R0 le repère de base.

m_k : Masse du corps k.

D_k : La matrice d'inertie de dimension (3x3) du corps k autour de son centre de masse exprimée dans le repère absolu R_0 .

Nous désirons formuler une expression de l'énergie cinétique totale en fonction de q et de \dot{q} . Pour cela, nous devons d'abord exprimer D_k, \bar{v}^k et $\bar{\omega}^k$ en fonction de q et de \dot{q} .

II.4.2.1 Tenseur d'inertie D_k

C'est une matrice de dimension 3x3 qui caractérise la distribution de la masse d'un corps rigide.

Avant de définir cette matrice par rapport au repère absolu, nous allons d'abord la définir par rapport à un repère $R_c = \{x^c, y^c, z^c\}$ dont l'origine est le centre de masse du corps en question.

Dans ce cas, le tenseur d'inertie est donnée par :

$$D_c = \begin{bmatrix} \int_V (y^2 + z^2) \rho dV & -\int_V x y \rho dV & -\int_V x z \rho dV \\ -\int_V x y \rho dV & \int_V (x^2 + z^2) \rho dV & -\int_V y z \rho dV \\ -\int_V x z \rho dV & -\int_V y z \rho dV & \int_V (x^2 + y^2) \rho dV \end{bmatrix} \quad (\text{II.4.3})$$

Avec :

ρ : La densité ou masse volumique du corps.

Dans cette matrice, on distingue :

- Les termes diagonaux qui sont positifs, ils sont appelés *moments d'inertie*.
- Les termes non diagonaux qui sont de signe quelconque, ils sont appelés *produits d'inertie*.

Pour revenir à la relation (II.4.2), nous cherchons D_k qui est le tenseur d'inertie du corps k par rapport à son centre de masse exprimé dans le repère absolu. Il est donné par :

$$D_k(q) = R_0^k(q) \bar{D}_k [R_0^k(q)]^T \quad (\text{II.4.4})$$

Avec :

\bar{D}_k : Le tenseur d'inertie du corps k par rapport à son centre de masse exprimé dans R^k . Il est obtenu en appliquant le théorème de Huygens généralisé donné en annexe B. 2.

II.4.2.2 Vitesses de translation et de rotation : $\bar{v}^k, \bar{\omega}^k$

Dans cette partie, nous introduisons la nouvelle matrice jacobienne $J(q)$ qui est différente de la matrice jacobienne $V(q)$ développée précédemment (II.3.1). Cette matrice $J(q)$ et dont le calcul est détaillé en annexe B. 3, lie les vitesses instantanées de translation \dot{p} et de rotation $\dot{\phi}$ aux vitesses articulaires. Elle est définie par :

$$\begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} = J(q) \cdot \dot{q} \quad (\text{II.4.5})$$

Afin d'exprimer \bar{v}^k et $\bar{\omega}^k$ en fonction de q et \dot{q} , il est aussi nécessaire d'introduire une matrice jacobienne analogue $J^k(q)$. Cette matrice $J^k(q)$ est calculée de la même manière que $J(q)$ en considérant le centre de masse du corps k comme étant l'extrémité de l'outil. Elle est définie par la relation :

$$\begin{bmatrix} \bar{v}^k \\ \bar{\omega}^k \end{bmatrix} = J^k(q) \cdot \dot{q} \quad 1 \leq k \leq n. \quad (\text{II.4.6})$$

L'expression des coordonnées du centre de masse du corps k exprimé dans le repère de base est donnée par :

$$\bar{c}^k(q) = H_1 T_0^k(q) \Delta c^k \quad 1 \leq k \leq n \quad (\text{II.4.7})$$

Avec :

Δc^k : Les coordonnées du centre de masse du corps k exprimé dans le repère R^k .

$H_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$: Cette matrice est introduite afin de passer d'un

vecteur de dimension 4 (coordonnées homogènes) à un vecteur de dimension 3 (coordonnées physiques).

Finalement ;

$$J^k(q) = \begin{bmatrix} \frac{\partial \bar{c}^k}{\partial q_1} & \dots & \frac{\partial \bar{c}^k}{\partial q_k} & 0 \\ \xi_1 a_0^0 & \dots & \xi_k a_0^{k-1} & 0 \end{bmatrix} = \begin{bmatrix} A^k(q) \\ B^k(q) \end{bmatrix} \quad (\text{II.4.8})$$

D'où :

$$\begin{aligned} \bar{v}^k(q, \dot{q}) &= A^k(q) \dot{q} \\ \text{et} \qquad \qquad \qquad & 1 \leq k \leq n \\ \bar{\omega}^k(q, \dot{q}) &= B^k(q) \dot{q} \end{aligned} \quad (\text{II.4.9})$$

En remplaçant ces deux expressions dans (II.4.2), après développement :

$$T(q, \dot{q}) = \frac{\dot{q}^T \cdot D(q) \cdot \dot{q}}{2} \quad (\text{II.4.10})$$

Avec :

$$D(q) \triangleq \sum_{k=1}^n \left\{ \left[A^k(q) \right]^T m_k A^k(q) + \left[B^k(q) \right]^T D_k B^k(q) \right\} \quad (\text{II.4.11})$$

II.4.3 ENERGIE POTENTIELLE $U(q)$

L'énergie potentielle emmagasinée dans le $k^{\text{ème}}$ corps du manipulateur est la quantité de travail nécessaire pour déplacer le centre de masse du corps k d'un plan horizontal de référence en présence de la force de gravité.

Dans notre cas, le plan de référence des énergies potentielles est évidemment le plan $\{x_0, y_0\}$ du repère de base R^0 .

L'énergie potentielle totale est donc donnée par :

$$U(q) = - \sum_{k=1}^n m_k \cdot g^T \cdot \bar{c}^k(q) \quad (\text{II.4.12})$$

Avec :

$$g = \begin{pmatrix} 0 \\ 0 \\ -g_0 \end{pmatrix} \in R^3 \text{ telle que : } g_0 = 9.8062 \text{ m/sec}^2$$

Afin de simplifier cette expression, nous posons :

$$\bar{c}(q) \triangleq -\sum_{k=1}^n m_k \cdot \bar{c}^k(q) \quad (\text{II.4.13})$$

C'est la somme pondérée des centres de masse des n corps du manipulateur.

Finalement, l'expression (II.4.12) de l'énergie potentielle devient :

$$U(q) = -g^T \cdot \bar{c}(q) \quad (\text{II.4.14})$$

II.4.4 FORCE GENERALISEE F

C'est l'ensemble des forces résiduelles agissant sur la structure une fois que les forces dues aux énergies cinétique et potentielle sont annulées.

Dans notre cas, la force généralisée est un vecteur de dimension $n \times 1$, défini implicitement par la relation :

$$\delta W = F^T \delta q \quad (\text{II.4.15})$$

Où :

δW représente le travail associé à un déplacement δq .

A noter que les composantes de F sont soit des forces soit des couples suivant le type d'articulation.

Dans notre cas, nous étudions deux types de forces généralisées, celles dues aux actionneurs (couples, moteurs), et celles dues aux frottements. D'où :

$$\delta W = [\tau - b(\dot{q})]^T \cdot \delta q \quad (\text{II.4.16})$$

Et :

$$F = [\tau - b(\dot{q})]^T \quad (\text{II.4.17})$$

Avec:

τ : Couples/ forces générés par les actionneurs.

$b(\dot{q})$: Fonction non linéaire qui modélise les forces de frottement.

II.4.5 APPLICATION DE L'ALGORITHME D'EULER LAGRANGE AU ROBOT SCARA

Dans ce qui précède, nous avons pu déterminer tous les termes entrant dans le calcul du modèle dynamique donné par l'équation (II.4.1). Dans la pratique, ces différentes expressions sont mises sous la forme de sommes et le calcul du modèle dynamique s'effectue suivant un algorithme général (appelé algorithme d'Euler Lagrange) facile à implémenter.

Ainsi, les équations (II.4.10) et (II.4.14) deviennent :

$$T(q, \dot{q}) = \frac{\sum_{k=1}^n \sum_{j=1}^n D_{kj}(q) \dot{q}_k \dot{q}_j}{2} \quad (\text{II.4.18})$$

$$U(q) = -\sum_{k=1}^3 \sum_{j=1}^n g_k m_j \bar{c}_k^j(q) \quad (\text{II.4.19})$$

Nous ne nous étalerons pas sur le développement de cet algorithme donné en annexe B. 4, qui n'est pas l'objet de notre étude. De même que pour les différents calculs intermédiaires résultants de son application qui encombreraient ce document.

C'est ainsi et après application de l'algorithme d'Euler Lagrange, que nous obtenons le modèle dynamique du robot SCARA, et qui est donné par :

$$\left\{ \begin{array}{l} \tau_1 = \left[\left(\frac{m_1}{3} + m_2 + m_3 \right) a_1^3 + (m_2 + 2m_3) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_1 \\ \quad - \left[\left(\frac{m_2}{2} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_2 + b_1(\dot{q}_1) \\ \quad - a_1 a_2 S_2 \left[(m_2 + 2m_3) \dot{q}_1 \dot{q}_2 - \left(\frac{m_2}{2} + m_3 \right) \dot{q}_2^2 \right] \\ \tau_2 = - \left[\left(\frac{m_2}{2} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_1 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \ddot{q}_2 \\ \quad + \left(\frac{m_2}{2} + m_3 \right) a_1 a_2 S_2 \dot{q}_1^2 + b_2(\dot{q}_2) \\ \tau_3 = m_3 \ddot{q}_3 - g_0 m_3 + b_3(\dot{q}_3) \end{array} \right. \quad (\text{II.4.20})$$

Notons que ces résultats sont valables en prenant certaines hypothèses simplificatrices. Les différents corps du robot sont assimilés à des tiges – ce qui est d'ailleurs le cas pour le dernier corps- dont le centre de masse se trouve au milieu. De même pour les différents

éléments d'assemblage (vis, bague, rondelle...etc.) dont les masses sont négligeables et que nous n'avons pas pris en compte lors de nos calculs. Ces hypothèses restent acceptables pour les deux premiers corps –qui sont des barres- bien que les masses des actionneurs excentrent le centre de masse, ce qui est considéré dans notre cas comme étant une erreur de modélisation.

Les équations du modèle dynamique peuvent se mettre sous la forme suivante :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + g(q) = \tau \quad (\text{II.4.21})$$

Avec :

- $M(q)\ddot{q}$ Les forces d'inertie.
- $C(q, \dot{q})\dot{q}$ Les forces centrifuges et forces de coriolis.
- $B\dot{q}$ Les forces de frottement.
- $g(q)$ Les forces gravitationnelles.
- τ Le vecteur des forces et/ou couples des actionneurs.

A noter que les valeurs réelles des paramètres du robot entrant dans ses différents modèles sont :

$$\begin{aligned} a_1 &= a_2 = 0.2 \text{ m} \\ m_1 &= 1.315 \text{ Kg} \\ m_2 &= 1.390 \text{ Kg} \\ m_3 &= 0.340 \text{ Kg} \end{aligned} \quad (\text{II.4.22})$$

Remarque : Conversion force/couple

Dans notre cas, le moteur 3 fournit un couple. Mais dans les équations du MD, la troisième articulation étant prismatique, τ_3 correspond donc à une force. Une transformation force/couple est nécessaire.

Nous avons :

$$q_3 = \frac{\text{pas}}{2\pi} \theta \quad (\text{II.4.23})$$

Avec :

$\text{pas} = 0.02 \text{ m}$. C'est le pas de filetage.

θ : Angle du moteur après réduction (en rad).

Dans notre cas, en appliquant le principe fondamental de la mécanique, nous avons :

$$\tau_3 = m_3 \ddot{q}_3 \quad (\text{II.4.24})$$

Car à l'équilibre, le poids du troisième corps est compensé.

Finalement le couple moteur est donnée par :

$$C_{moteur} = J_m \frac{2\pi}{pas \cdot m_3} \tau_3 \quad (II.4.25)$$

II.5 MODELE DU MOTEUR A COURANT CONTINU [11]

Dans cette partie, nous abordons le problème de la modélisation du moteur à courant continu. Ce modèle est nécessaire pour la commande du robot dont il sera question lors du prochain chapitre.

Les modèles que nous développons impliquent l'utilisation de valeurs relatives pour les positions angulaires des moteurs. Ces valeurs relatives sont introduites dans un but de simplification des modèles et donc des commandes, car ceux-ci ne seront que d'avantage fastidieux avec l'addition des conditions initiales à chaque étape du calcul.

Le fonctionnement d'un moteur à courant continu est la conjonction de deux phénomènes physiques :

- Le premier est qu'un courant électrique i parcourant un conducteur placé dans un champ magnétique ϕ engendre une force $F = \phi \times i$. Dans un moteur à courant continu, le courant du rotor (ou courant d'induit) i_a parcourt un champ magnétique constant ϕ engendré par le stator créant ainsi un couple (ou une force à un facteur près) qui fait tourner l'arbre du moteur. Ce couple est donné par :

$$\tau_m = K_1 \phi i_a \quad (II.5.1)$$

- Le second est qu'un conducteur en mouvement à la vitesse ω_m placé à l'intérieur d'un champ magnétique ϕ crée une force contre électromotrice –car elle tend à s'opposer au sens du courant dans le conducteur- donnée par la relation :

$$V_b = K_2 \phi \omega_m \quad (II.5.2)$$

Le fonctionnement du moteur à courant continu peut être représenté par la figure (II.5) :

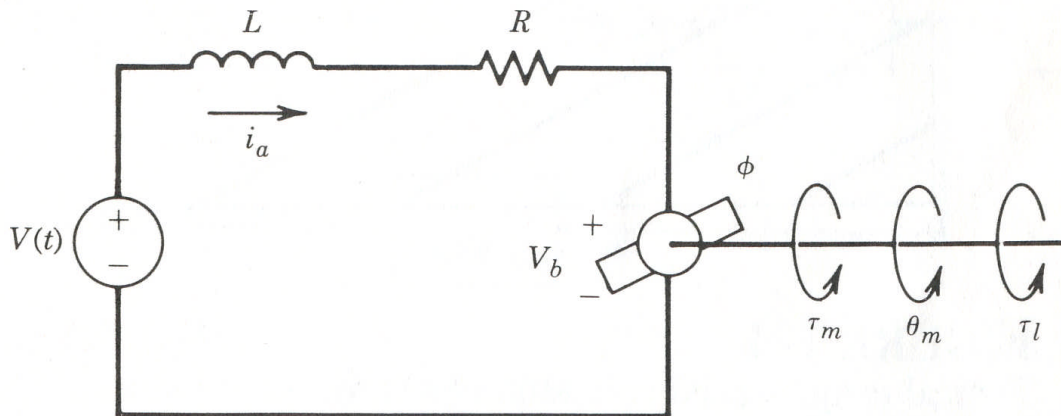


Figure II.5 Circuit de fonctionnement du moteur

On désigne par :

$V(t)$: La tension d'alimentation (en Volt).

R : La résistance de l'induit (en Ω).

L : L'inductance de l'induit (en Henri).

V_b : La force contre électromotrice induite (en Volt).

i_a : Le courant de l'induit (en Ampères).

$\theta_m(t)$: La position angulaire relative du rotor dans le domaine temporel (en radians)

$\Theta_m(s)$: La position angulaire relative du rotor dans l'espace de transfert (domaine de Laplace)

τ_m : Le couple généré par le moteur (en N.m).

τ_l : Le couple de charge (en N.m).

ϕ : Le flux magnétique généré par l'inducteur (en Weber).

L'application de la loi de Kirchoff relative aux mailles nous permet d'avoir l'équation différentielle (II.5.3) pour le courant du rotor:

$$L \frac{di_a(t)}{dt} + R i_a = V(t) - V_b \quad (\text{II.5.3})$$

Examinons maintenant la partie mécanique du système.

Puisque le flux ϕ est constant, le couple développé par le moteur est proportionnel au courant de l'induit comme indiqué par la relation (II.5.4) :

$$\tau_m = K_m i_a \quad (\text{II.5.4})$$

Avec :

K_m : La constante du couple (en N.m/A).

D'autre part, on a une relation de proportionnalité liant la force contre électromotrice à la vitesse angulaire du rotor comme le montre la relation (II.5.5) :

$$V_b = K_2 \phi \omega_m = K_b \omega_m = K_b \frac{d\theta_m}{dt} \quad (\text{II.5.5})$$

Telle que :

K_b : est la constante de la force contre électromotrice induite (en V · s / rad).

Appliquons la seconde loi de Newton. Nous obtenons alors l'équation mécanique (II.5.6) :

$$J_m \frac{d^2\theta_m}{dt^2} + B_m \frac{d\theta_m}{dt} = \tau_m - r\tau_l \quad (\text{II.5.6})$$

Avec :

$J_m = J_a + J_g$: La somme des moments d'inertie de l'actionneur et du réducteur de vitesse.

$B_m \frac{d\theta_m}{dt}$: Une approximation linéaire des forces de frottements visqueux.

r : l'inverse du rapport de réduction des vitesses (1 :r), dans notre cas, c'est le même pour les trois moteurs utilisés puisque ceux-ci sont identiques.

En combinant les relations (II.5.3), (II.5.4), (II.5.5) et (II.5.6) et en négligeant la constante de temps électrique $\frac{L}{R}$ devant la constante de temps mécanique $\frac{J_m}{B_m}$, on obtient l'équation (II.5.7) :

$$J_m \ddot{\theta}_m(t) + \left(B_m + \frac{K_b K_m}{R} \right) \dot{\theta}_m(t) = \frac{K_m}{R} V(t) - r\tau_l(t) \quad (\text{II.5.7})$$

C'est une équation différentielle d'ordre 2 en θ_m avec second membre qui représente le modèle du moteur à courant continu.

Après passage au domaine de Laplace ($\tau_l = 0$), l'équation (II.5.7) devient :

$$\Theta_m(s) = \frac{\frac{K_m}{R}}{s(J_m s + B_m + \frac{K_b K_m}{R})} V(s) \quad (\text{II.5.8})$$

Notons que la $k^{\text{ème}}$ variable articulaire q_k et la position angulaire relative θ_{mk} du $k^{\text{ème}}$ moteur sont liées par la relation :

$$q_k = r\theta_{mk} + q_k^i \quad k = 1, \dots, n \quad (\text{II.5.9})$$

Avec :

q_k^i : La valeur initiale de la $k^{\text{ème}}$ variable articulaire.

Nous pouvons aussi représenter le modèle du moteur en utilisant la représentation schéma bloc - qui est plus commode pour la simulation du système en boucle fermée – donnée par la figure II.6 :

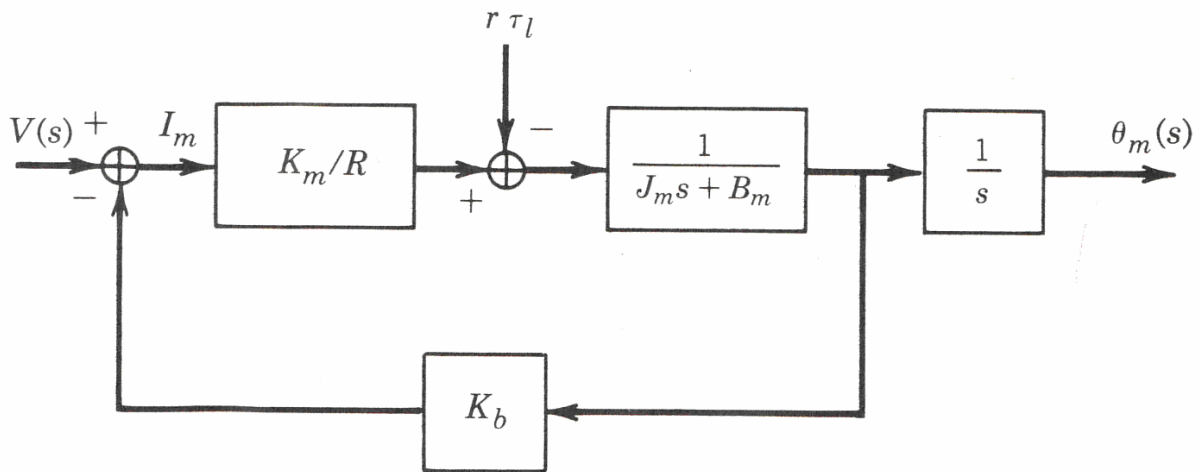


Figure II.6 Schéma bloc du système réduit

Ce qui résout le problème du calcul du modèle du moteur à courant continu dans l'espace de transfert.

II.6 CONCLUSION

L'objectif de ce chapitre était d'aboutir à la formulation des différents modèles mathématiques de notre robot. Ces modèles mettent en évidence la difficulté de commander ce genre de systèmes, étant données les fortes non linéarités qui régissent le fonctionnement dynamique de ceux-ci.

Des hypothèses simplificatrices qui concordent avec la réalité - et de ce point de vue donc restent acceptables - ont été émises tout au long de la phase de modélisation afin de simplifier ces modèles. Le modèle des connaissances développé pour le modèle dynamique a comme avantage de comporter des termes ayant une signification physique. Sa détermination par identification serait intéressante et peut faire l'objet de travaux futurs.

Les modèles obtenus seront utilisés pour la commande du robot dont fera l'objet le prochain chapitre.

Voici donc les différents modèles auxquels nous avons abouti au cours de ce chapitre :

Modèle géométrique direct MGD

$$w(q) = \begin{bmatrix} a_1 C_1 + a_2 C_{1-2} \\ a_1 S_1 + a_2 S_{1-2} \\ d_1 - q_3 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

Modèle géométrique inverse MGI

$$\begin{cases} q_3 = d_1 - P_z \\ q_2 = \pm \arccos\left(\frac{P_x^2 + P_y^2 - a_1^2 - a_2^2}{2 a_1 a_2}\right) \\ q_1 = A \tan 2 \left[a_2 S_2 P_x + (a_1 + a_2 C_2) P_y, (a_1 + a_2 C_2) P_x - a_2 S_2 P_y \right] \end{cases}$$

Modèle cinématique direct MCD

$$V(q) = \begin{pmatrix} -a_1 S_1 - a_2 S_{1-2} & a_2 S_{1-2} & 0 \\ a_1 C_1 + a_2 C_{1-2} & -a_2 C_{1-2} & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Modèle cinématique inverse MCI

$$\begin{cases} \dot{q}_1 = \frac{1}{\Delta} (\dot{P}_x \delta (\alpha \delta - \beta \gamma) + \dot{P}_y \beta (\beta \gamma - \alpha \delta)) \\ \dot{q}_2 = \frac{1}{\Delta} (\dot{P}_x \gamma (\beta \gamma - \alpha \delta) + \dot{P}_y \alpha (\alpha \delta - \beta \gamma)) \\ \dot{q}_3 = -\dot{P}_z \end{cases}$$

Avec :

$$\Delta = (\alpha^2 + \gamma^2)(\beta^2 + \delta^2) - (\alpha\beta + \gamma\delta)^2$$

Et

$$\begin{cases} \alpha = -a_1 S_1 - a_2 S_{1-2} \\ \beta = a_2 S_{1-2} \\ \gamma = a_1 C_1 + a_2 C_{1-2} \\ \delta = -a_2 C_{1-2} \end{cases}$$

Modèle dynamique MD

$$\left\{ \begin{array}{l} \tau_1 = \left[\left(\frac{m_1}{3} + m_2 + m_3 \right) a_1^3 + (m_2 + 2m_3) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_1 \\ \quad - \left[\left(\frac{m_2}{2} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_2 + b_1(\dot{q}_1) \\ \quad - a_1 a_2 S_2 \left[(m_2 + 2m_3) \dot{q}_1 \dot{q}_2 - \left(\frac{m_2}{2} + m_3 \right) \dot{q}_2^2 \right] \\ \tau_2 = - \left[\left(\frac{m_2}{2} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{q}_1 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \ddot{q}_2 \\ \quad + \left(\frac{m_2}{2} + m_3 \right) a_1 a_2 S_2 \dot{q}_1^2 + b_2(\dot{q}_2) \\ \tau_3 = m_3 \ddot{q}_3 - g_0 m_3 + b_3(\dot{q}_3) \end{array} \right.$$

Modèle du Moteur à Courant Continu

$$\Theta_m(s) = \frac{\frac{K_m}{R}}{s(J_m s + B_m + \frac{K_d K_m}{R})} V(s)$$

Telles que :

$$q_k = r\theta_{mk} + q_k^i \quad k = 1, \dots, n$$

CHAPITRE III
COMMANDE ET
SIMULATION

CHAPITRE III

COMMANDE ET SIMULATION

III.1 INTRODUCTION

La commande des robots manipulateurs a toujours constitué un sujet de recherche important pour les automaticiens. Ceci est dû en fait aux spécificités de ce genre de systèmes. En effet, en plus de présenter des couplages entre les entrées/sorties et des dynamiques fortement non linéaires, leurs modèles ont des paramètres qui peuvent varier dans le temps du fait du changement de charge, de configuration et de vitesse du mouvement par exemple.

Il apparaît donc clairement qu'il n'existe pas de stratégie de commande systématique des robots manipulateurs. Chaque manipulateur nécessite une stratégie de commande qui est spécifique à ses caractéristiques propres. Ces dernières sont la morphologie du manipulateur, son nombre de degrés de liberté ou encore le type d'actionneurs utilisés (électriques, pneumatiques ou hydrauliques, actionneurs ayant des rapports de réduction élevés ou pas ...etc.).

Dans ce chapitre, nous présentons trois techniques de commande différentes ; une commande linéarisante par géométrie différentielle, la méthode du couple calculé et en dernier lieu la commande décentralisée du robot.

Pour les deux premières méthodes, le système à commander est basé sur les équations du modèle dynamique. Le robot est vu comme un système non linéaire de type *MIMO* présentant des couplages importants. La différence majeure entre les deux méthodes est que la première commande le robot directement dans l'espace opérationnel alors que la seconde effectue une commande dans l'espace articulaire.

Une étude comparative entre les deux méthodes est donnée et présente des résultats très intéressants.

La troisième méthode (commande décentralisée) est beaucoup plus simple. Il s'agira en fait de commander indépendamment chaque articulation en se basant sur le modèle du moteur.

Le robot est donc perçu comme un ensemble de trois systèmes linéaires de type *SISO* où l'effet des couplages est perçu comme des perturbations. Celles-ci ont donc été injectées au système lors de la simulation afin d'approcher autant que possible le comportement du système réel.

Un correcteur PID dont la structure est similaire à celle implémentée dans le contrôleur du robot a été simulé.

III.2 GENERATION DE TRAJECTOIRE [4], [6]

D'une manière générale, et lors de la commande d'un manipulateur, on définit la tâche à effectuer dans l'espace opérationnel, celle-ci étant facilement et naturellement visible dans cet espace. S'ensuit généralement le passage à l'espace articulaire dans lequel sera déterminé le mouvement du robot qui constitue la consigne des systèmes d'asservissement des actionneurs. Reste alors la synthèse de la commande proprement dite.

Pour notre part, nous nous limitons dans ce chapitre à la génération de trajectoire dans l'espace articulaire, et ce pour obtenir les consignes à donner lors des simulations. Le problème de la génération de trajectoire dans l'espace opérationnel sera traité lors de la planification de tâche dans le prochain chapitre.

Nous générons la trajectoire articulaire en effectuant l'interpolation de chaque variable articulaire avec une fonction d'interpolation $r(t)$ telle que :

$$\begin{cases} q_r(t) = q^i + r(t)(q^f - q^i) \\ \dot{q}_r(t) = \dot{r}(t)(q^f - q^i) \\ \ddot{q}_r(t) = \ddot{r}(t)(q^f - q^i) \end{cases} \quad (\text{III.2.1})$$

Avec :

q^i et q^f les valeurs initiales et finales des variables articulaires respectivement.

La fonction $r(t)$ doit être calculée de façon à satisfaire les conditions :

- Positions désirées au départ et à l'arrivée.
- Vitesses nulles au départ et à l'arrivée.
- Accélérations nulles au départ et à l'arrivée.

Elle doit donc vérifier les propriétés suivantes :

$$\begin{cases} r(0) = 0 \\ r(t_f) = 1 \end{cases} \quad (\text{III.2.2})$$

Nous choisissons comme expression pour $r(t)$, un polynôme d'ordre 5, celui-ci ne présentant des discontinuités ni pour le profil de vitesse, ni pour celui de l'accélération.

Nous obtenons finalement les expressions :

$$\begin{cases} r(t) = 10\left(\frac{t}{t_f}\right)^3 - 15\left(\frac{t}{t_f}\right)^4 + 6\left(\frac{t}{t_f}\right)^5 \\ \dot{r}(t) = \frac{30}{t_f^3}t^2 - \frac{60}{t_f^4}t^3 + \frac{30}{t_f^5}t^4 \\ \ddot{r}(t) = \frac{60}{t_f^3}t - \frac{180}{t_f^4}t^2 + \frac{120}{t_f^5}t^3 \end{cases} \quad (\text{III.2.3})$$

Ces dernières seront par la suite injectées dans les relations (III.2.1), et on aura ainsi généré nos trajectoires de référence.

Pour la simulation de ces trajectoires, les vecteurs positions initiales et finales choisis sont :

$$q^i = \left(\frac{\pi}{6} \text{Rad} \quad \frac{\pi}{6} \text{Rad} \quad 0.1\text{m} \right)^T \text{ et } q^f = \left(\frac{\pi}{2} \text{Rad} \quad \frac{\pi}{2} \text{Rad} \quad 0.3\text{m} \right)^T$$

Les profils des trajectoires générées représentant les positions articulaires, les vitesses articulaires et les sorties générées (calculées avec le modèle géométrique direct) sont donnés dans les figures III.1, III.2 et III.3 respectivement :

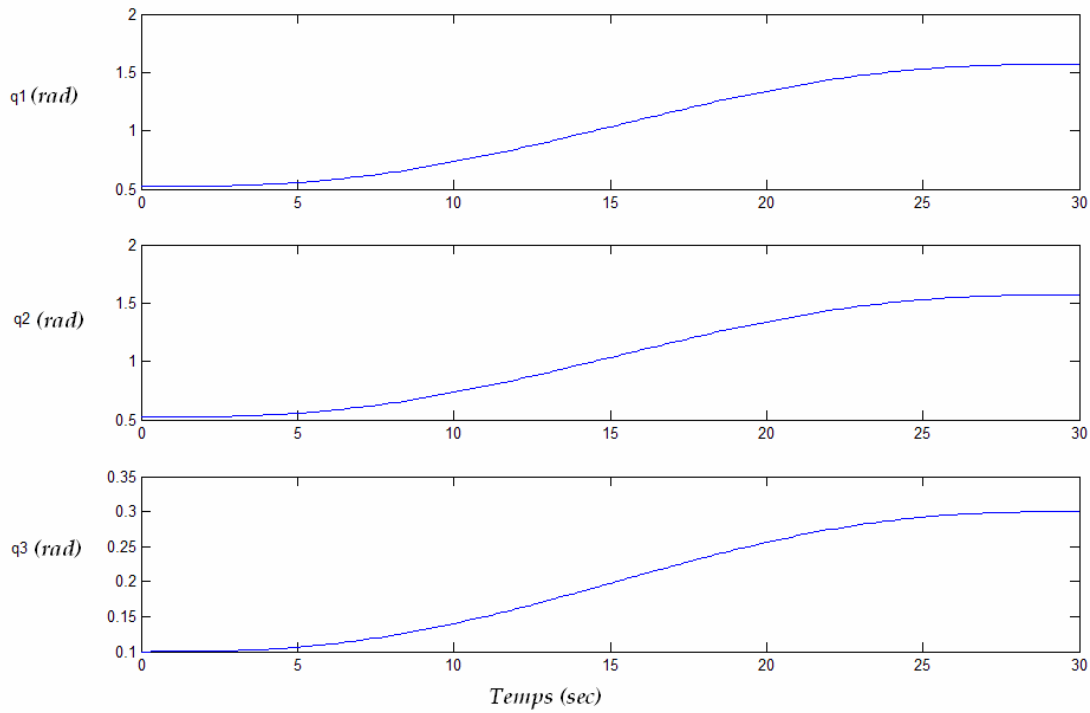


Figure III.1 Profil des variables articulaires générées

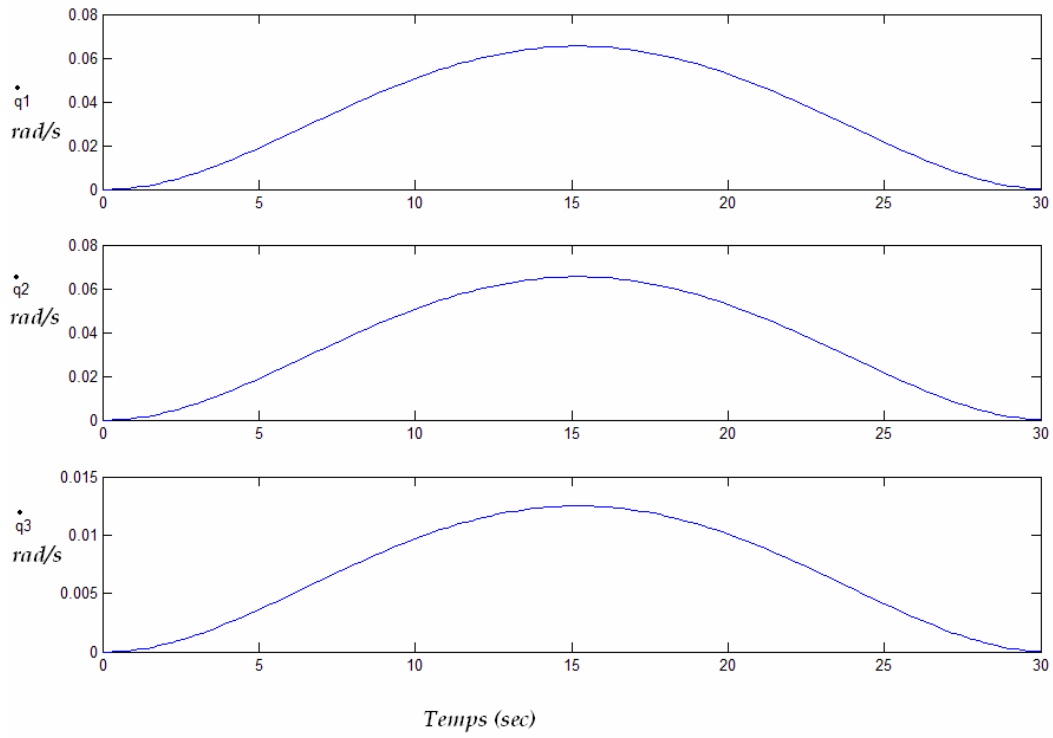


Figure III.2 Profil des vitesses articulaires générées

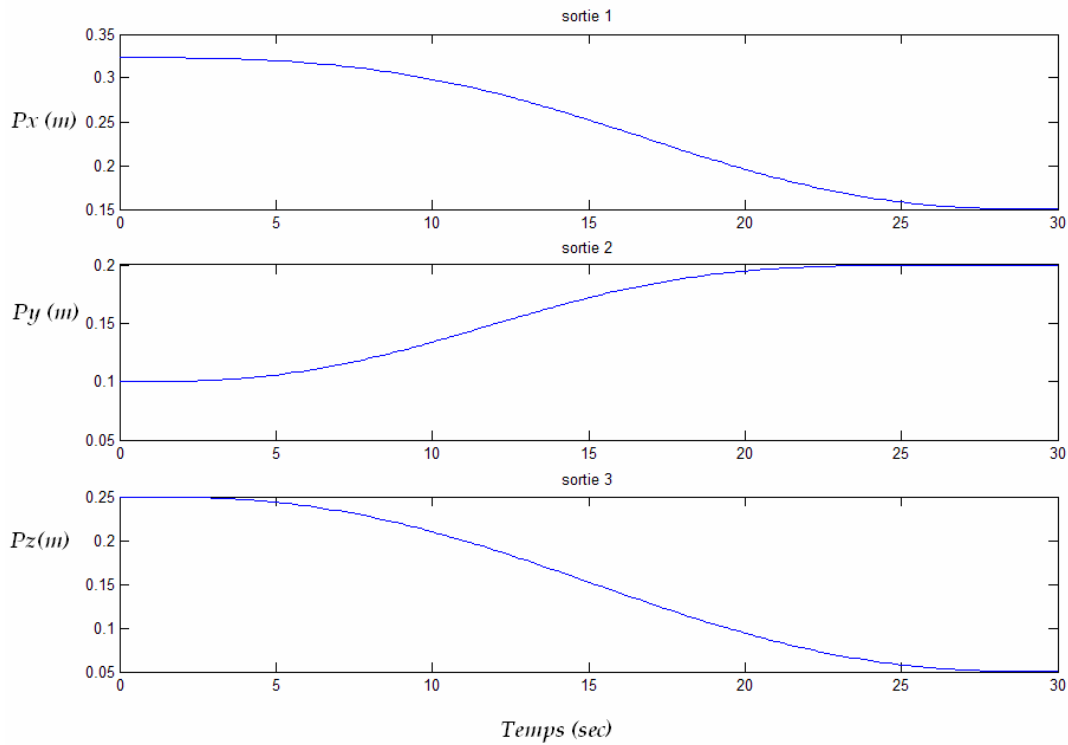


Figure III.3 Consignes opérationnelles correspondantes

Ces sorties constituent donc les consignes à donner au robot lors de la commande.

III.3 COMMANDE LINEARISANTE PAR GEOMETRIE

DIFFERENTIELLE [12], [13]

III.3.1 EQUATIONS D'ETAT ET LINEARISATION DU SYSTEME

A partir du modèle dynamique donné par la relation (II.4.20), nous pouvons tirer les équations d'état du système et ce, en prenant pour vecteur d'état : $x^T = [q^T \dot{q}^T]$ et comme commande le vecteur couple $u^T = [\tau_1 \ \tau_2 \ \tau_3]$. La sortie du système est donnée quant à elle par le vecteur position opérationnelle : $P^T = [P_x \ P_y \ P_z]$.

Le vecteur d'état est donc de dimension (6 x 1), où les 3 premières composantes sont les variables articulaires et les 3 dernières représentent les vitesses articulaires.

Ainsi donc, nous obtenons les équations d'état du système, qui sont données par :

$$\left\{ \begin{array}{l} \dot{x}_1 = x_4 \\ \dot{x}_2 = x_5 \\ \dot{x}_3 = x_6 \\ \dot{x}_4 = \frac{1}{\Delta} \left(D_{22} \left\{ u_1 + a_1 a_2 S_2 \left[(m_2 + 2m_3)x_4 x_5 - \left(\frac{m_2}{2} + m_3\right)x_5^2 \right] \right\} - D_{12} \left[u_2 - \left(\frac{m_2}{2} + m_3\right)a_1 a_2 S_2 x_4^2 \right] \right) \\ \dot{x}_5 = \frac{1}{\Delta} \left(-D_{21} \left\{ u_1 + a_1 a_2 S_2 \left[(m_2 + 2m_3)x_4 x_5 - \left(\frac{m_2}{2} + m_3\right)x_5^2 \right] \right\} + D_{11} \left[u_2 - \left(\frac{m_2}{2} + m_3\right)a_1 a_2 S_2 x_4^2 \right] \right) \\ \dot{x}_6 = \frac{u_3 + g_0 m_3}{m_3} \\ y_1 = a_1 C_1 + a_2 C_{12} \\ y_2 = a_1 S_1 + a_2 S_{12} \\ y_3 = d_1 - x_3 \end{array} \right. \quad (III.3.1)$$

Avec : $\Delta = D_{11} D_{22} - D_{12} D_{21}$

$$\begin{aligned}
 C_1 &= \text{Cos } q_1 \\
 C_2 &= \text{Cos } q_2 \\
 S_1 &= \text{Sin } q_1 \\
 S_2 &= \text{Sin } q_2 \\
 C_{12} &= \text{Cos}(q_1 - q_2) \\
 S_{12} &= \text{Cos}(q_1 - q_2)
 \end{aligned} \tag{III.3.2}$$

Et :

$$\begin{aligned}
 D_{11} &= \left(\frac{m_1}{3} + m_2 + m_3\right)a_1^3 + (m_2 + 2m_3)a_1a_2C_2 + \left(\frac{m_2}{3} + m_3\right)a_2^2 \\
 D_{12} = D_{21} &= -\left[\left(\frac{m_2}{2} + m_3\right)a_1a_2C_2 + \left(\frac{m_2}{3} + m_3\right)a_2^2\right] \\
 D_{22} &= \left(\frac{m_2}{3} + m_3\right)a_2^2
 \end{aligned} \tag{III.3.3}$$

Remarque :

Pour plus de commodité, nous préférons écrire la 4^{ème} et 5^{ème} équations d'état sous la forme suivante :

$$\begin{aligned}
 \dot{x}_4 &= f_4(x) + \frac{D_{22}}{\Delta}u_1 - \frac{D_{12}}{\Delta}u_2 \\
 \dot{x}_5 &= f_5(x) - \frac{D_{21}}{\Delta}u_1 + \frac{D_{11}}{\Delta}u_2
 \end{aligned} \tag{III.3.4}$$

Telles que :

$$\begin{aligned}
 f_4(x) &= \frac{1}{\Delta} \left(a_1a_2S_2D_{22} \left[(m_2 + 2m_3)x_4x_5 - \left(\frac{m_2}{2} + m_3\right)x_5^2 \right] + D_{12} \left(\frac{m_2}{2} + m_3\right)a_1a_2S_2x_4^2 \right) \\
 f_5(x) &= \frac{1}{\Delta} \left(-a_1a_2S_2D_{21} \left[(m_2 + 2m_3)x_4x_5 - \left(\frac{m_2}{2} + m_3\right)x_5^2 \right] - D_{11} \left(\frac{m_2}{2} + m_3\right)a_1a_2S_2x_4^2 \right)
 \end{aligned} \tag{III.3.5}$$

Ceci pour faciliter les calculs dans la recherche de la commande linéarisante.

III.3.1.1 Calcul du degré relatif

Commençons tout d'abord par déterminer le degré relatif du système en dérivant les sorties du système jusqu'à ce que l'une des commandes apparaisse.

Nous trouvons :

$$\begin{cases} y_1 = a_1 C_1 + a_2 C_{12} \\ y_1^{(1)} = -a_1 S_1 x_4 - a_2 S_{12} (x_4 - x_5) \\ y_1^{(2)} = -a_1 C_1 x_4^2 - a_2 C_{12} (x_4 - x_5)^2 - [a_1 S_1 + a_2 S_{12}] \dot{x}_4 + a_2 S_{12} \dot{x}_5 \end{cases} \quad (\text{III.3.6})$$

D'où :

$$y_1^{(2)} = C_1(x) - \left[\frac{D_{22}}{\Delta} (a_1 S_1 + a_2 S_{12}) + a_2 S_{12} \frac{D_{21}}{\Delta} \right] u_1 + \left[\frac{D_{12}}{\Delta} (a_1 S_1 + a_2 S_{12}) + a_2 S_{12} \frac{D_{11}}{\Delta} \right] u_2 \quad (\text{III.3.7})$$

Avec :

$$C_1(x) = L_f^2 h_1(x) = -a_1 C_1 x_4^2 - a_2 C_{12} (x_4 - x_5)^2 - [a_1 S_1 + a_2 S_{12}] f_4 + a_2 S_{12} f_5 \quad (\text{III.3.8})$$

Nous en déduisons que :

$$r_1 = 2 \quad (\text{III.3.9})$$

De même pour la deuxième sortie :

$$\begin{cases} y_2 = a_1 S_1 + a_2 S_{12} \\ y_2^{(1)} = a_1 C_1 x_4 + a_2 C_{12} (x_4 - x_5) \\ y_2^{(2)} = -a_1 S_1 x_4^2 - a_2 S_{12} (x_4 - x_5)^2 + [a_1 C_1 + a_2 C_{12}] \dot{x}_4 - a_2 C_{12} \dot{x}_5 \end{cases} \quad (\text{III.3.10})$$

D'où :

$$y_2^{(2)} = C_2(x) + \left[\frac{D_{22}}{\Delta} (a_1 C_1 + a_2 C_{12}) - a_2 C_{12} \frac{D_{21}}{\Delta} \right] u_1 - \left[\frac{D_{12}}{\Delta} (a_1 C_1 + a_2 C_{12}) + a_2 C_{12} \frac{D_{11}}{\Delta} \right] u_2 \quad (\text{III.3.11})$$

Avec :

$$C_2(x) = L_f^2 h_2(x) = -a_1 S_1 x_4^2 - a_2 S_{12} (x_4 - x_5)^2 + [a_1 C_1 + a_2 C_{12}] f_4 - a_2 C_{12} f_5 \quad (\text{III.3.12})$$

Nous en déduisons que :

$$r_2 = 2 \quad (\text{III.3.13})$$

Et enfin pour la dernière sortie nous trouvons :

$$\begin{cases} y_3 = d_1 - x_3 \\ y_3^{(1)} = -\dot{x}_3 \\ y_3^{(2)} = -g_0 - \frac{1}{m_3} u_3 \end{cases} \quad (\text{III.3.14})$$

Avec :

$$L_f^2 h_3(x) = -g_0 \quad (\text{III.3.15})$$

D'où :

$$r_3 = 2 \quad (\text{III.3.16})$$

Nous pouvons écrire les équations précédentes sous la forme suivante :

$$\begin{bmatrix} y_1^{(2)} \\ y_2^{(2)} \\ y_3^{(2)} \end{bmatrix} = \begin{bmatrix} L_f^2 h_1(x) \\ L_f^2 h_2(x) \\ L_f^2 h_3(x) \end{bmatrix} + A(x) \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (\text{III.3.17})$$

Avec :

$$A(x) = \begin{pmatrix} -\left[\frac{D_{22}}{\Delta} (a_1 S_1 + a_2 S_{12}) + a_2 S_{12} \frac{D_{21}}{\Delta} \right] & \left[\frac{D_{12}}{\Delta} (a_1 S_1 + a_2 S_{12}) + a_2 S_{12} \frac{D_{11}}{\Delta} \right] & 0 \\ \left[\frac{D_{22}}{\Delta} (a_1 C_1 + a_2 C_{12}) - a_2 C_{12} \frac{D_{21}}{\Delta} \right] & -\left[\frac{D_{12}}{\Delta} (a_1 C_1 + a_2 C_{12}) + a_2 C_{12} \frac{D_{11}}{\Delta} \right] & 0 \\ 0 & 0 & -\frac{1}{m_3} \end{pmatrix} \quad (\text{III.3.18})$$

Son déterminant est égal à :

$$\det[A(x)] = \frac{a_1 a_2 S_2}{m_3 \cdot \Delta} \quad (\text{III.3.19})$$

Il est non nul pour :

$$q_2 \neq k\pi \quad (\text{III.3.20})$$

Nous pouvons donc conclure que le degré relatif de notre système est :

$$r = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} \text{ Et ce pour } q_2 \neq k\pi. \quad (\text{III.3.21})$$

III.3.1.2 Calcul de la commande linéarisante

La commande réalisant la linéarisation entrée/sortie est donnée par :

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = A^{-1}(x) \left[-\begin{bmatrix} L_f^2 h_1(x) \\ L_f^2 h_2(x) \\ L_f^2 h_3(x) \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \right] \quad (\text{III.3.22})$$

D'où :

$$\begin{bmatrix} y_1^{(2)} \\ y_2^{(2)} \\ y_3^{(2)} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (\text{III.3.23})$$

Ce qui devient dans le domaine de Laplace :

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} = \begin{pmatrix} \frac{1}{s^2} & 0 & 0 \\ 0 & \frac{1}{s^2} & 0 \\ 0 & 0 & \frac{1}{s^2} \end{pmatrix} \begin{bmatrix} v_1(s) \\ v_2(s) \\ v_3(s) \end{bmatrix} \quad (\text{III.3.24})$$

C'est-à-dire :

$$\frac{y_1(s)}{v_1(s)} = \frac{1}{s^2}, \frac{y_2(s)}{v_2(s)} = \frac{1}{s^2} \text{ et } \frac{y_3(s)}{v_3(s)} = \frac{1}{s^2} \quad (\text{III.3.25})$$

Nous voyons bien que chaque commande $v_i(s)$ agit uniquement sur la sortie correspondante $y_i(s)$.

Ceci veut dire que cette commande, en plus d'assurer la linéarisation du système—comme son nom l'indique- elle réalise le découplage entre les entrées/sorties.

III.3.1.3 Forme normale

A partir du moment où $\sum_{i=1}^3 r_i = 6 = n$ (La dimension du vecteur d'état). Nous pouvons conclure que tout le vecteur d'état peut être linéarisé, et ce, en opérant le changement de variable suivant :

$$\begin{cases} Z_1 = y_1 = h_1(x) \\ Z_2 = \dot{y}_1 = L_f h_1(x) \\ Z_3 = y_2 = h_2(x) \\ Z_4 = \dot{y}_2 = L_f h_2(x) \\ Z_5 = y_3 = h_3(x) \\ Z_6 = \dot{y}_3 = L_f h_3(x) \end{cases} \quad (\text{III.3.26})$$

Autrement dit :

$$\left\{ \begin{array}{l} \dot{Z}_1 = Z_2 \\ \dot{Z}_2 = v_1 \end{array} \right\} \text{ Sous système (1)} \\ \left\{ \begin{array}{l} \dot{Z}_3 = Z_4 \\ \dot{Z}_4 = v_2 \end{array} \right\} \text{ Sous système (2)} \\ \left\{ \begin{array}{l} \dot{Z}_5 = Z_6 \\ \dot{Z}_6 = v_3 \end{array} \right\} \text{ Sous système (3)} \quad (\text{III.3.27})$$

La sortie quant à elle s'écrit :

$$y = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} Z \quad (\text{III.3.28})$$

Vérification que $\phi(x)$ est un difféomorphisme

Nous savons que $z = \phi(x)$, c'est-à-dire :

$$\left\{ \begin{array}{l} z_1 = a_1 \cos(x_1) + a_2 \cos(x_1 - x_2) \\ z_2 = -a_1 x_4 \sin(x_1) - a_2 (x_4 - x_5) \sin(x_1 - x_2) \\ z_3 = a_1 \sin(x_1) + a_2 \sin(x_1 - x_2) \\ z_4 = a_1 x_4 \cos(x_1) + a_2 (x_4 - x_5) \cos(x_1 - x_2) \\ z_5 = d_1 - x_3 \\ z_6 = -x_6 \end{array} \right. \quad (\text{III.3.29})$$

La condition suffisante pour que $\phi(x)$ soit un difféomorphisme est :

$$\left| \frac{\partial \phi}{\partial x} \right| \neq 0 \quad (\text{III.3.30})$$

Le calcul de $\frac{\partial \phi}{\partial x}$ donne :

$$\frac{\partial \phi}{\partial x} = \begin{pmatrix} -a_1 s_1 - a_2 s_{1-2} & a_2 s_{1-2} & 0 & 0 & 0 & 0 \\ -a_1 c_1 x_4 - a_2 c_{1-2} (x_4 - x_5) & a_2 c_{1-2} (x_4 - x_5) & 0 & -a_1 s_1 - a_2 s_{1-2} & a_2 s_{1-2} & 0 \\ a_1 c_1 + a_2 c_{1-2} & -a_2 c_{1-2} & 0 & 0 & 0 & 0 \\ -a_1 s_1 x_4 - a_2 s_{1-2} (x_4 - x_5) & a_2 s_{1-2} (x_4 - x_5) & 0 & a_1 c_1 + a_2 c_{1-2} & -a_2 c_{1-2} & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

$$(III.3.31)$$

Et celui du déterminant :

$$\left| \frac{\partial \phi}{\partial x} \right| = a_1^2 a_2^2 \sin^2(x_2) \quad (III.3.32)$$

Il est non nul pour :

$$q_2 \neq k\pi \quad (III.3.33)$$

On peut donc conclure que $\phi(x)$ est un difféomorphisme à condition que la deuxième variable articulaire q_2 soit différente de $k\pi$.

III.3.3 COMMANDE DU SYSTEME LINEARISE

Puisque le système est constitué en fait de trois sous systèmes découplés nous pouvons donc commander chaque sortie indépendamment.

Les trois sous systèmes étant identiques, la commande obtenue pour l'un peut servir pour les deux autres.

Le sous-système s'écrit :

$$\begin{cases} \dot{Z}_1 = Z_2 \\ \dot{Z}_2 = v_1 \\ y_1 = Z_1 \end{cases} \quad (III.3.34)$$

Ou bien :

$$\begin{cases} \begin{pmatrix} \dot{Z}_1 \\ \dot{Z}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v_1 \\ y_1 = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \end{cases} \quad (III.3.35)$$

Commandabilité :

Avant de synthétiser une commande pour ce système, il est impératif d'en étudier la commandabilité.

La matrice de commandabilité est donnée par :

$$M_c = (B \quad AB) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (\text{III.3.36})$$

Puisque :

$$\text{Rang}[M_c] = 2 \quad (\text{III.3.37})$$

Le critère de Kalman est vérifié, la paire (A, B) est donc complètement commandable. La matrice de retour d'état K existe.

III.3.3.1 Retour d'état

Nous appliquons dans un premier temps un retour d'état simple donné par :

$$v(t) = -K \cdot z(t) \quad (\text{III.3.38})$$

Avec :

$r(t)$: La référence qui n'est autre que la sortie désirée du système.

Pour un placement de pôles : $V^T = [-2 \quad -1]$, nous obtenons les réponses suivantes :

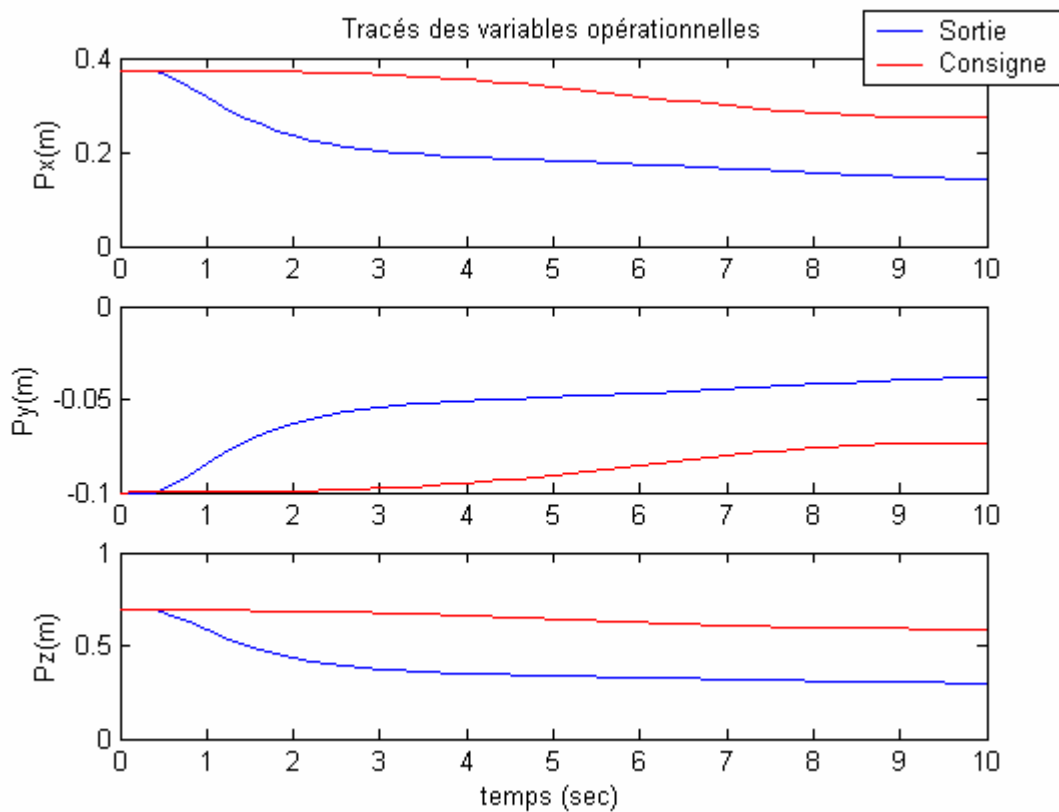


Figure III.4 Résultats de la poursuite après application d'un retour d'état simple

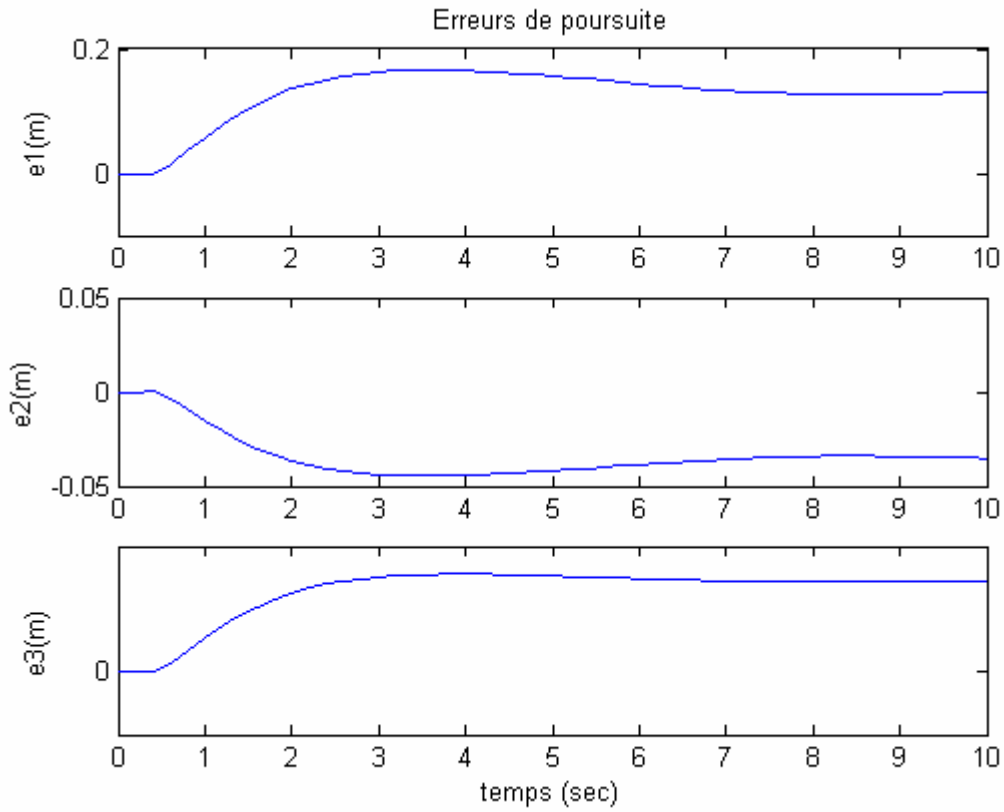


Figure III.5 Erreurs de poursuite pour le retour d'état simple

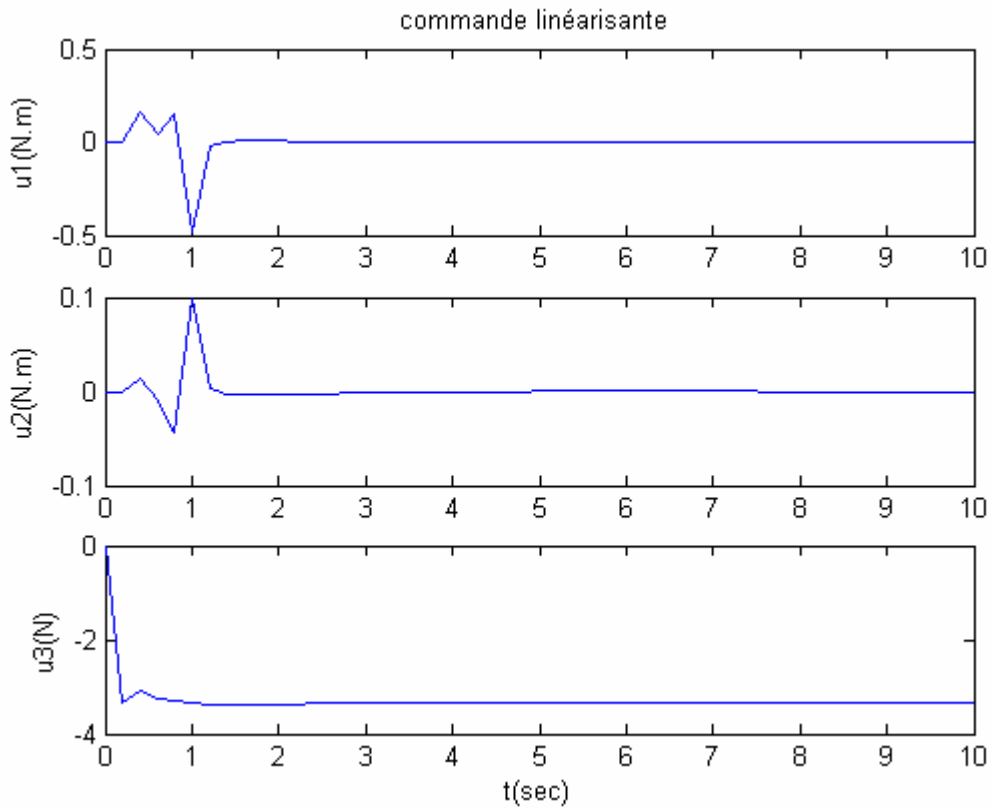


Figure III.6 Signaux de commande réels correspondant au retour d'état simple

Les résultats de poursuite obtenus ne sont pas assez satisfaisants. Afin de les améliorer, nous introduisons dans un premier temps un filtre précompensateur puis une action intégrale.

Ajout d'un filtre précompensateur :

La commande appliquée est de la forme :

$$v(t) = -K \cdot z(t) + N \cdot r(t) \tag{III.3.39}$$

Avec :

N : Le filtre précompensateur qui est donné par :

$$N = [C \cdot [-A + BK]^{-1} \cdot B]^{-1} \tag{III.3.40}$$

Et :

$r(t)$: La référence.

Pour les simulations, nous avons gardé le même placement de pôles que précédemment (même dynamique désirée). Les résultats obtenus sont les suivants :

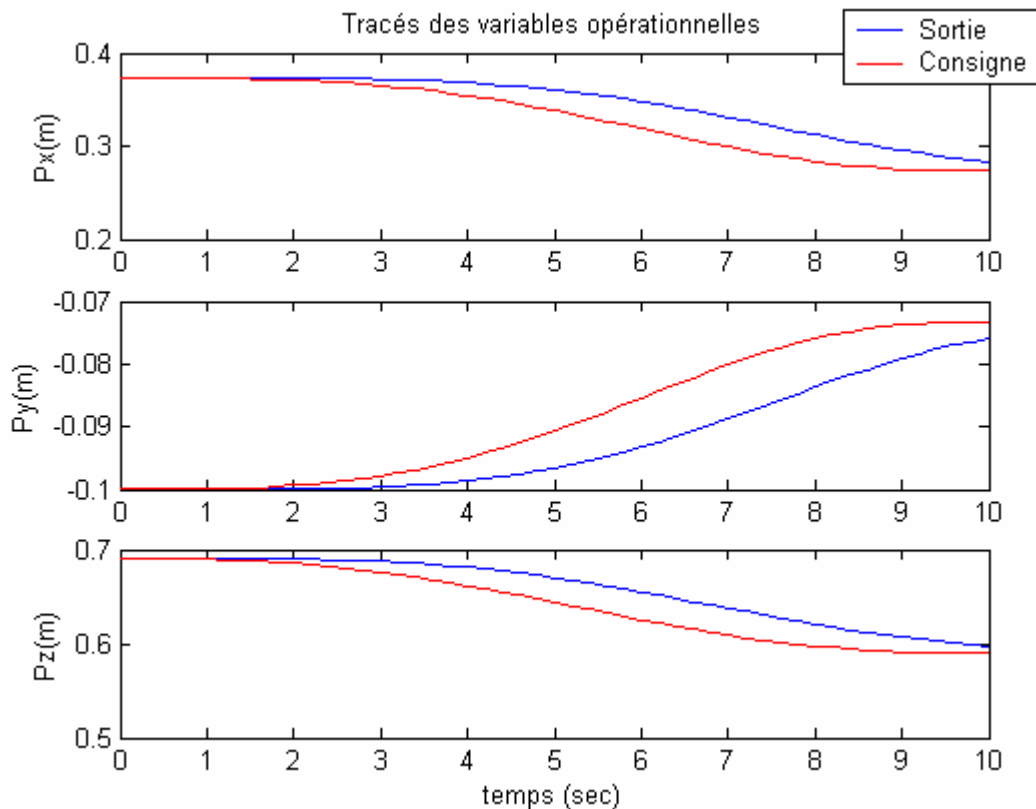


Figure III.7 Résultats de la poursuite après application d'un retour d'état avec ajout du filtre précompensateur

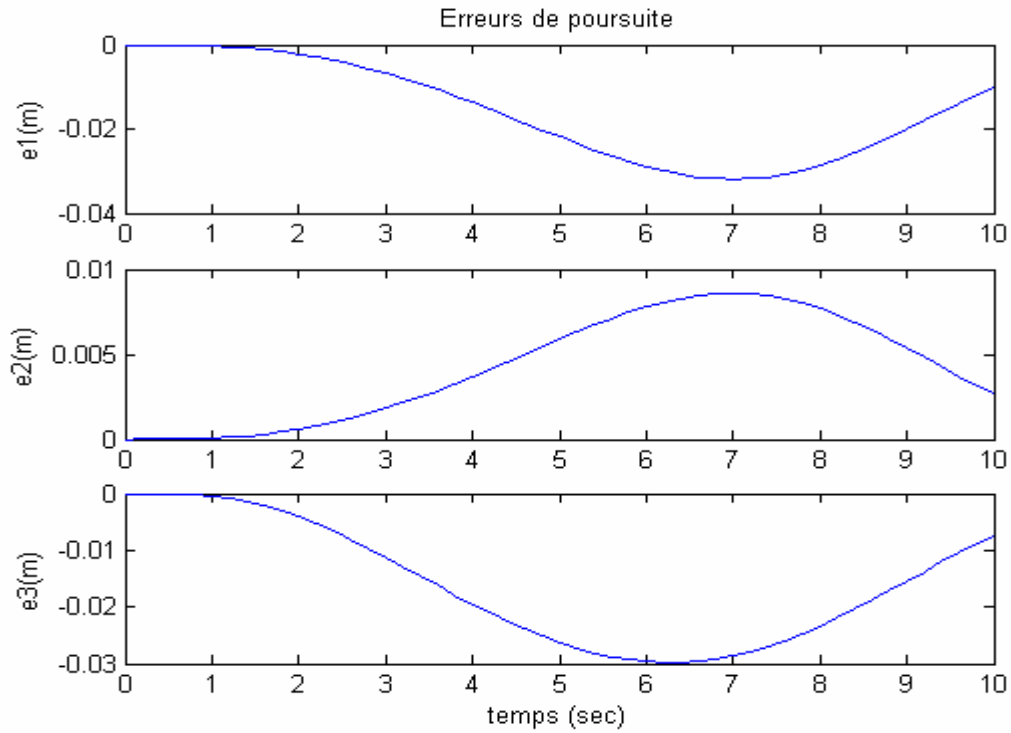


Figure III.8 Erreurs de poursuite pour le retour d'état avec ajout du filtre précompensateur

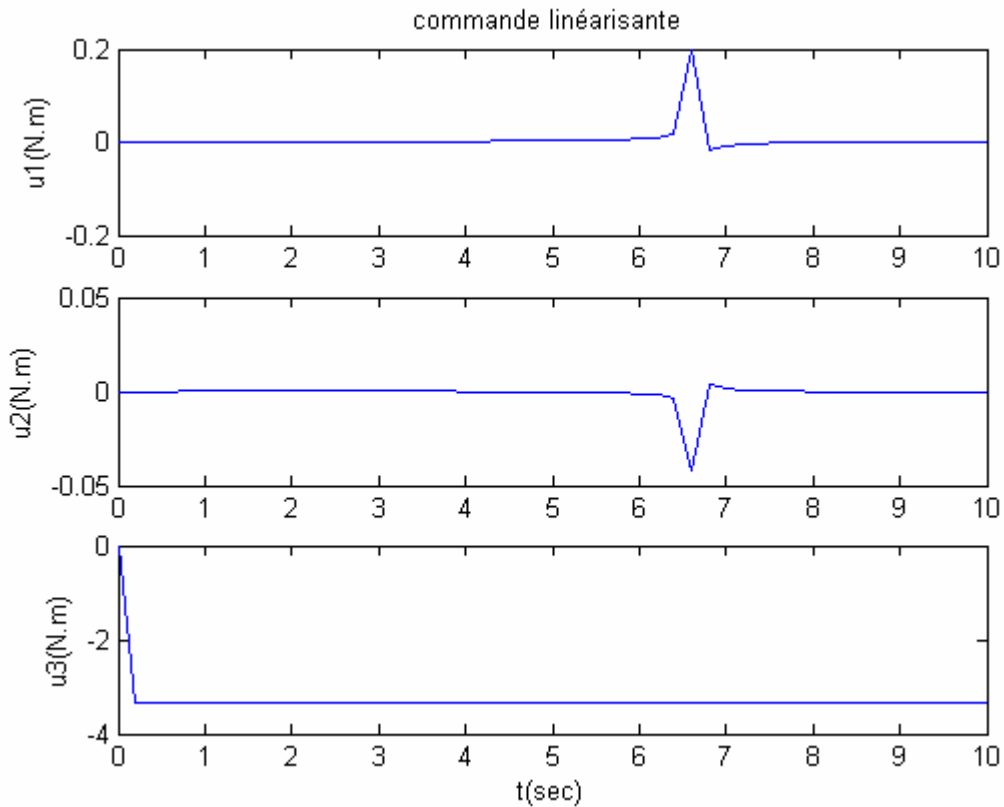


Figure III.9 Signaux de commande réels correspondant au retour d'état avec ajout du filtre précompensateur

Nous constatons une amélioration sensible de la réponse du système. Cependant, les signaux d'erreurs sont quand même significatifs pour certaines tâches nécessitant plus de précision.

Intégration de l'erreur de poursuite :

Dans cette partie, on ajoute au sous-système une troisième dynamique qui est celle de l'erreur sur la sortie correspondante comme l'indiquent les relations suivantes :

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = v_1 \\ \dot{f}_1 = z_1 - y_{1r} = e_1 \end{cases} \quad (\text{III.3.41})$$

Le système ainsi défini est augmenté et peut être mis sous la forme matricielle suivante :

$$\begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{f}_1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ f_1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} v_1 + \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} y_{1r} \quad (\text{III.3.42})$$

Il est clair qu'avant de commander le système augmenté, nous devons en étudier la commandabilité au préalable.

Etude de la commandabilité du système augmenté :

$$M_c = (B \quad AB) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{III.3.43})$$

Puisque :

$$\text{Rang}[M_c] = 3 \quad (\text{III.3.44})$$

Le critère de Kalman est vérifié, la paire (A, B) est donc complètement commandable.

Pour un placement de pôles $V^T = [-2 \quad -1 \quad -1.5]$, les résultats de simulation sont :

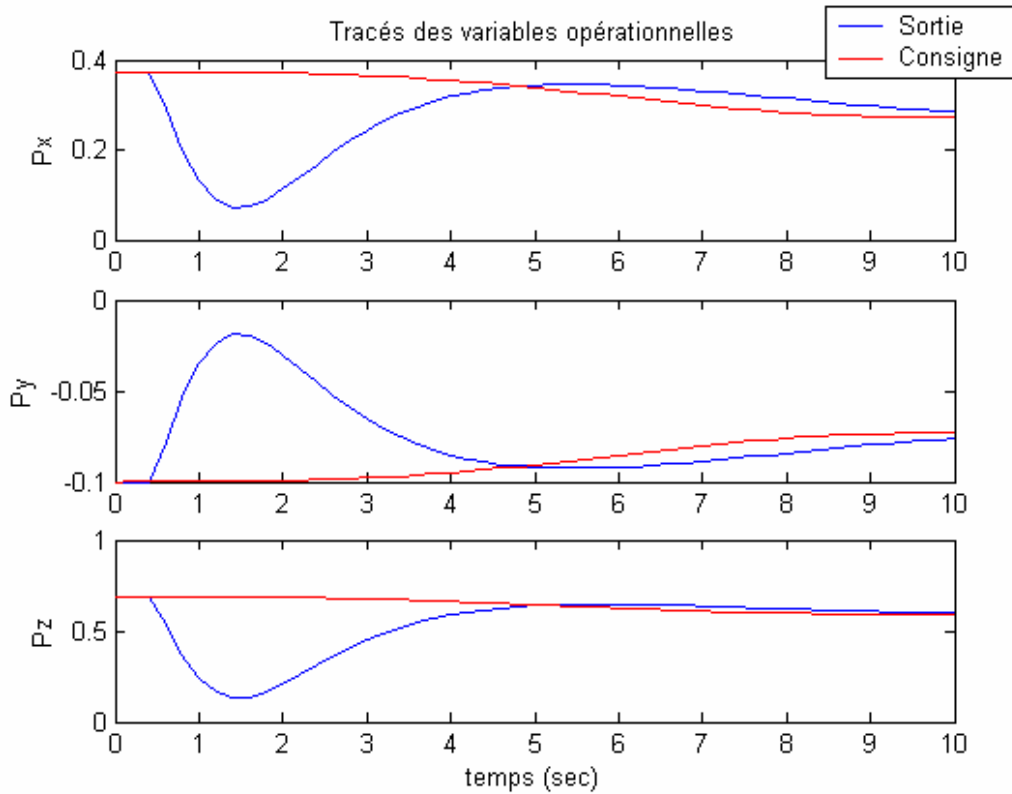


Figure III.10 Résultats de la poursuite après application d'un retour d'état avec intégration de l'erreur de poursuite

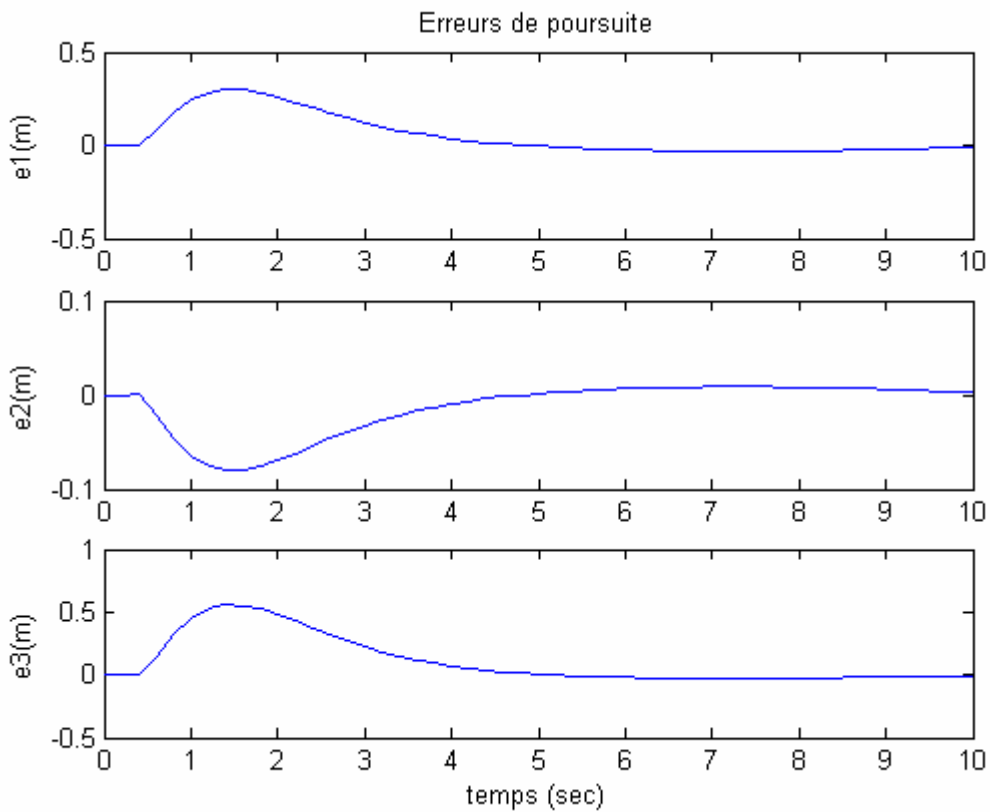


Figure III.11 Erreurs de poursuite pour le retour d'état avec intégration de l'erreur de poursuite

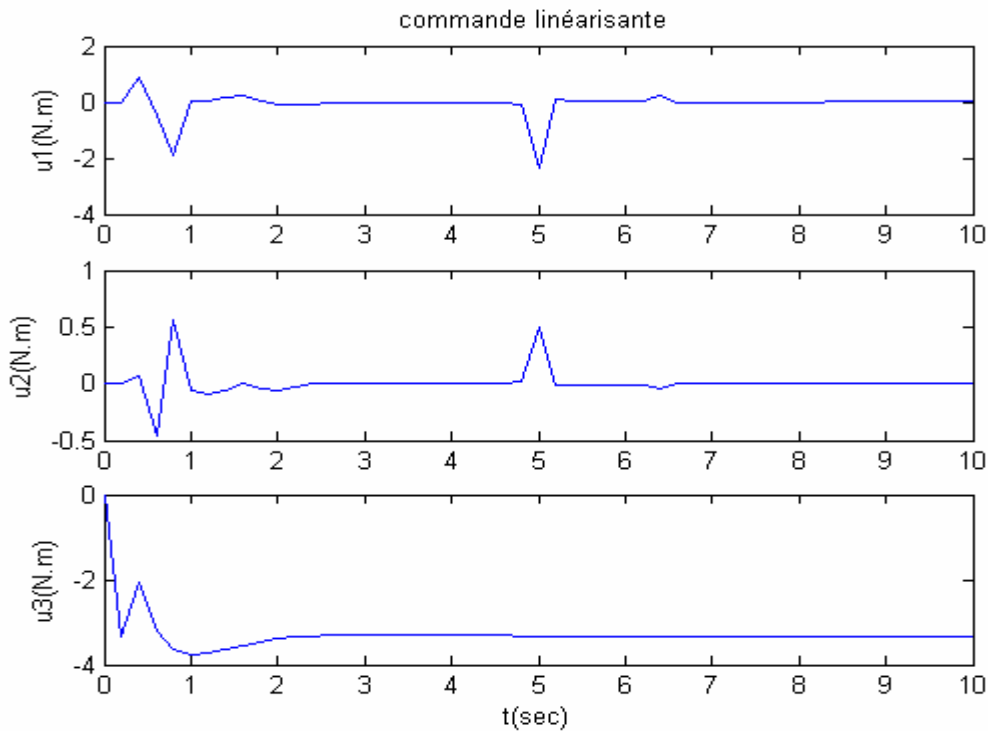


Figure III.12 Signaux de commande réels correspondant au retour d'état avec intégration de l'erreur de poursuite

Les figures précédentes illustrent bien la mauvaise poursuite de trajectoire obtenue par cette méthode. La sortie du système augmenté passe par une phase où elle ne suit pas du tout la consigne. Nous constatons la présence d'un zéro à l'origine pour le système augmenté qui expliquerait certainement un tel comportement.

Modification de la référence :

Même si le retour d'état donne satisfaction pour des consignes fixes. Les résultats précédents montre bien que ceci n'est pas vrai pour la poursuite de référence. Nous expliquons ceci par l'absence d'informations sur les vitesses et accélérations désirées dans le terme de référence.

La démarche adoptée consiste donc à modifier le terme de référence dans le but d'annuler le signal d'erreur suivant la dynamique désirée donnée par :

$$\ddot{e}_1 + k_1 \dot{e}_1 + k_0 e_1 = 0 \tag{III.3.45}$$

En remplaçant le signal d'erreur par son expression :

$$e_1 = y_1^d - y_1 \tag{III.3.46}$$

Nous obtenons :

$$(\ddot{y}_1^d - \ddot{y}_1) + k_1(\dot{y}_1^d - \dot{y}_1) + k_0(y_1^d - y_1) = 0 \Leftrightarrow (\ddot{y}_1^d - v_1) + k_1(\dot{y}_1^d - \dot{y}_1) + k_0(y_1^d - y_1) = 0 \quad (\text{III.3.47})$$

Car $v_1 = \ddot{y}_1$ (relation III.3.34).

Nous tirons l'expression de la commande permettant d'obtenir la convergence asymptotique du signal d'erreur :

$$v_1 = \ddot{y}_1^r + k_1\dot{e}_1 + k_0e_1 \quad (\text{III.3.48})$$

Nous savons aussi que :

$$v_1 = -k_0y_1 - k_1\dot{y}_1 + r \quad (\text{III.3.49})$$

D'où, par identification, nous obtenons la forme de la référence :

$$r = \ddot{y}_1^r + k_0y_1 + k_1\dot{y}_1 \quad (\text{III.3.50})$$

Les gains sont choisis de sorte qu'ils vérifient :

$$\begin{aligned} k_0 &= \omega^2 \\ k_1 &= 2\omega \end{aligned} \quad (\text{III.3.51})$$

Les résultats de simulation (pour $\omega = 2 \text{ rad/s}$) sont les suivants :

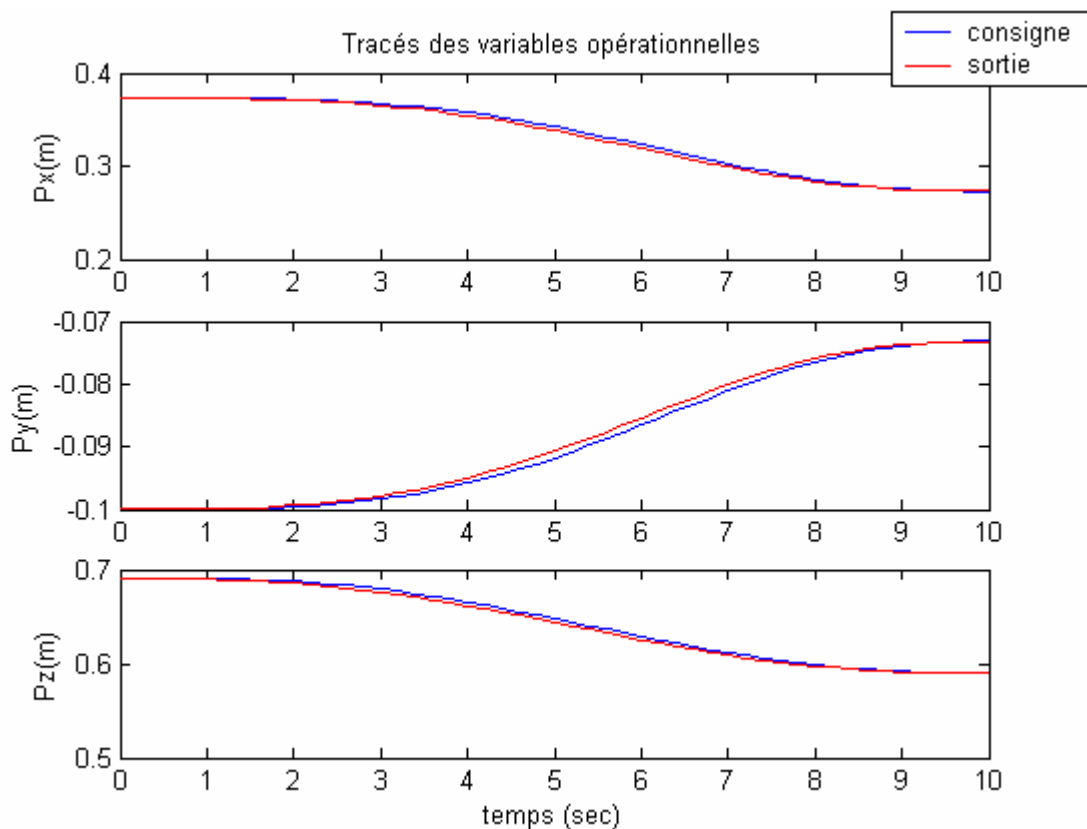


Figure III.13 Résultats de la poursuite après application d'un retour d'état après modification du signal de référence

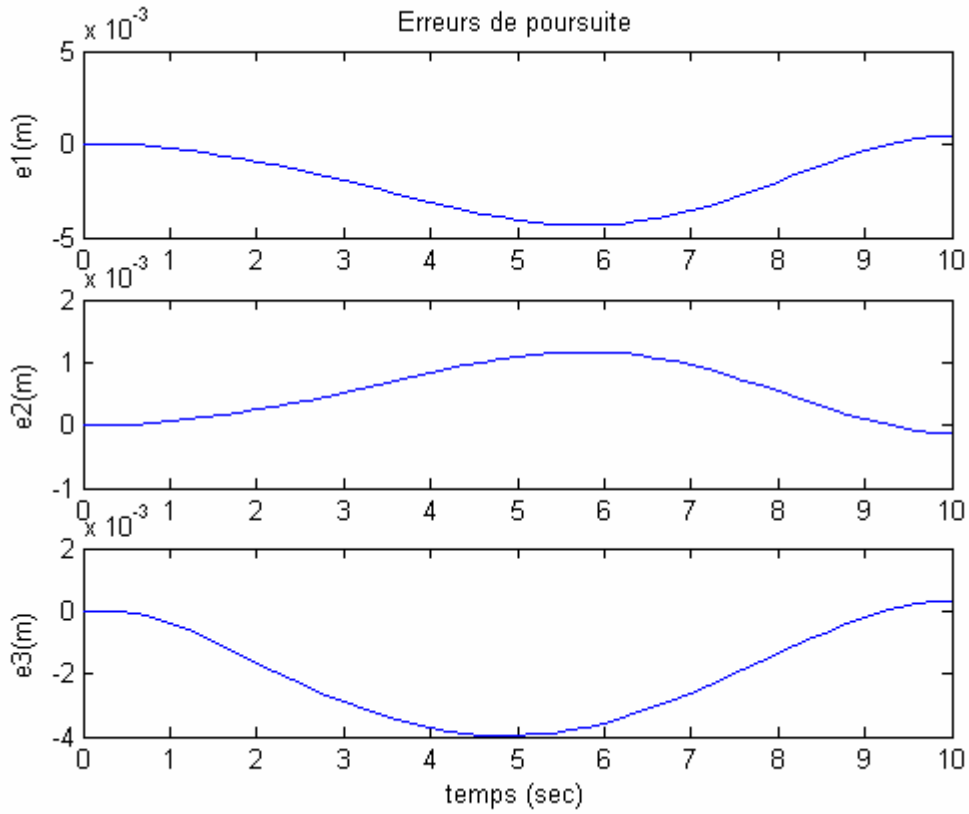


Figure III.14 Erreurs de poursuite pour le retour d'état avec signal de référence modifiée

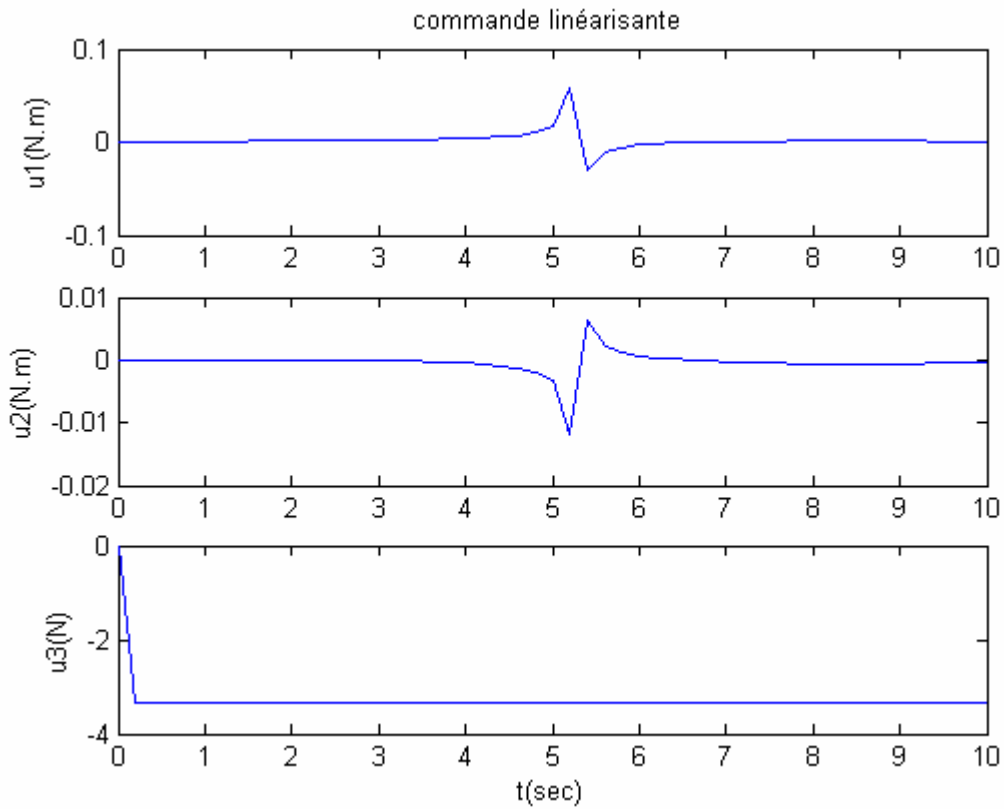


Figure III.15 Signaux de commande réels correspondant au retour d'état avec référence modifiée

Nous voyons bien que la sortie suit la référence, ce qui confirme bien que le choix de la référence est très important. En effet, l'ajout des termes d'accélération et de vitesses désirées a conduit à de bons résultats. Ceci est logique car pour pouvoir poursuivre une sortie donnée l'information sur la position uniquement est insuffisante, il faut aussi connaître la vitesse et l'accélération de celle-ci.

III.3.3.2 Backstepping

Reprenons les équations du système :

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = v - \ddot{y}_r \\ y = z_1 \end{cases} \quad (\text{III.3.52})$$

Avec :

$$\begin{cases} e_1 = z_1 - y_r \\ e_2 = z_2 - \dot{y}_r \end{cases} \quad (\text{III.3.53})$$

Le principe de cette méthode est le suivant :

Etape 1 :

Considérer dans un premier temps e_2 comme étant une commande pour e_1 . Il ne reste plus qu'à synthétiser la nouvelle commande pour obtenir la convergence de l'erreur vers zéro. On définit alors la fonction candidate de Lyapunov V_1 définie et positive :

$$V_1 = \frac{1}{2} e_1^2 \quad (\text{III.3.54})$$

Sa dérivée est donnée par :

$$\dot{V}_1 = e_1 \dot{e}_1 = e_1 e_2 \quad (\text{III.3.55})$$

Elle est définie négative pour :

$$e_2 = -K e_1. \quad (\text{III.3.56})$$

Et ce pour $K > 0$.

Etape 2 :

Calculer la commande v permettant de faire tendre e_2 vers $-K e_1$ autrement dit $e_2 + K e_1 \rightarrow 0$.

Pour cela, on définit la fonction de Lyapunov suivante :

$$V_2 = \frac{1}{2} e_1^2 + \frac{1}{2} (e_2 + K e_1)^2 \quad (\text{III.3.57})$$

Sa dérivée est donnée par :

$$\begin{aligned} \dot{V}_2 &= e_1 \dot{e}_1 + (e_2 + K e_1)(\dot{e}_2 + K \dot{e}_1) \\ &= e_1 e_2 + (e_2 + K e_1)(v - \ddot{y}_r + K e_2) \\ &= e_1 e_2 + K e_1^2 - K e_1^2 + (e_2 + K e_1)(v - \ddot{y}_r + K e_2) \\ &= -K e_1^2 + e_1 (e_2 + K e_1) + (e_2 + K e_1)(v - \ddot{y}_r + K e_2) \end{aligned} \quad (\text{III.3.58})$$

Pour obtenir la convergence de l'erreur, il faut que la dérivée de la fonction candidate soit définie négative. C'est-à-dire :

$$\dot{V}_2 = -K e_1^2 - \beta (e_2 + K e_1)^2 < 0 \quad (\text{III.3.59})$$

Avec $\beta > 0$

Il suffit pour cela de prendre la commande suivante :

$$v = \ddot{y}_r - K e_2 - e_1 - \beta (e_2 + K e_1) \quad (\text{III.3.60})$$

En posant : $K_p = 1 + K\beta$ et $K_d = K + \beta$

Nous obtenons finalement :

$$v = \ddot{y}_r - K_p e_1 - K_d \dot{e}_1 \quad (\text{III.3.61})$$

Nous retrouvons la même forme de commande que celle développée précédemment pour le retour d'état (III.3.48).

Les résultats de simulations sont les suivants :

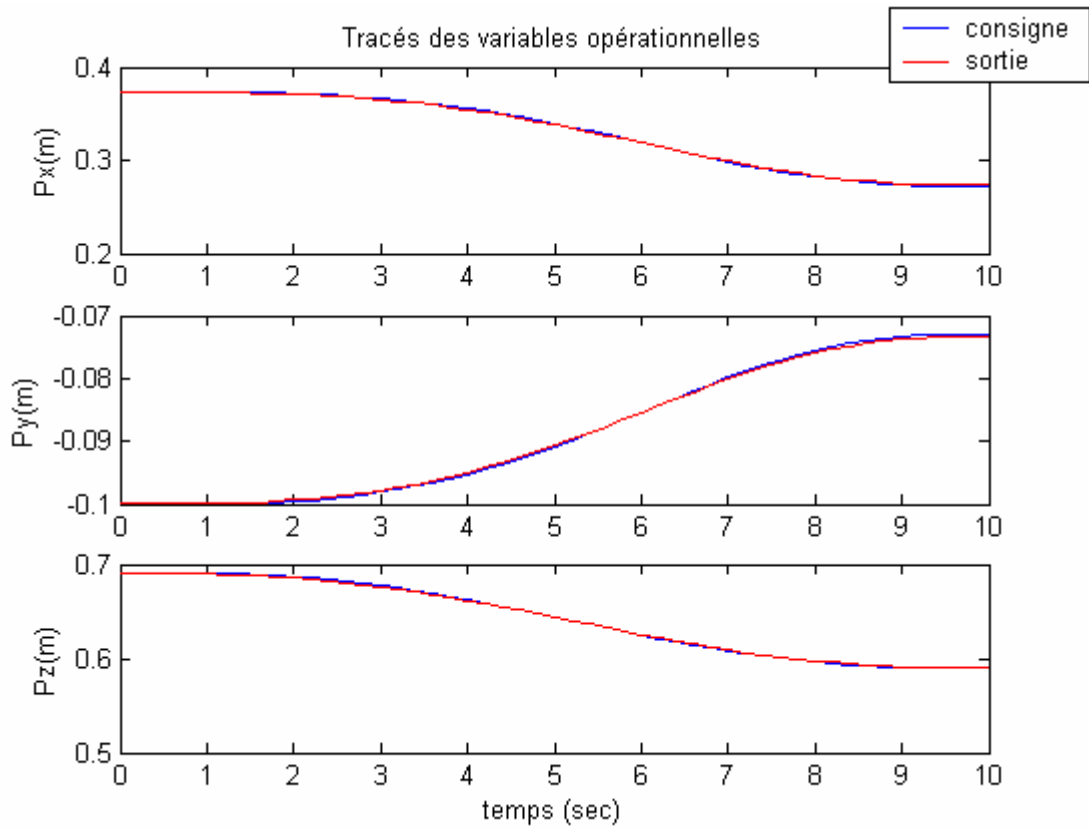


Figure III.16 Résultats de la poursuite après application de la commande par Backstepping

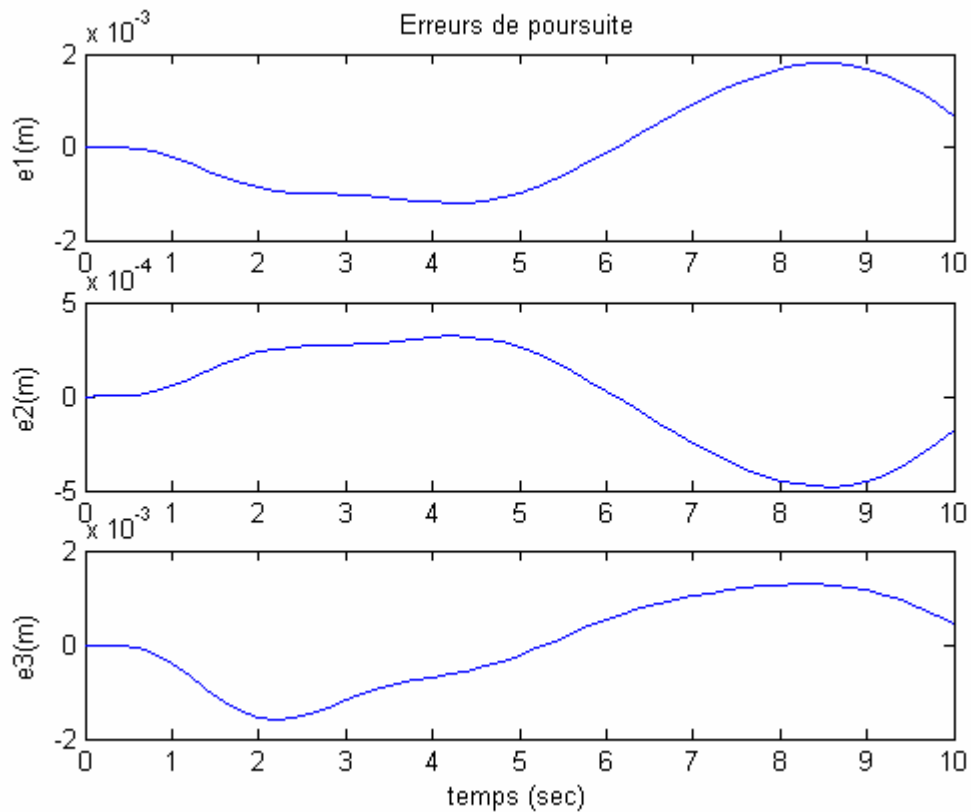


Figure III.17 Erreurs de poursuite pour la commande par Backstepping

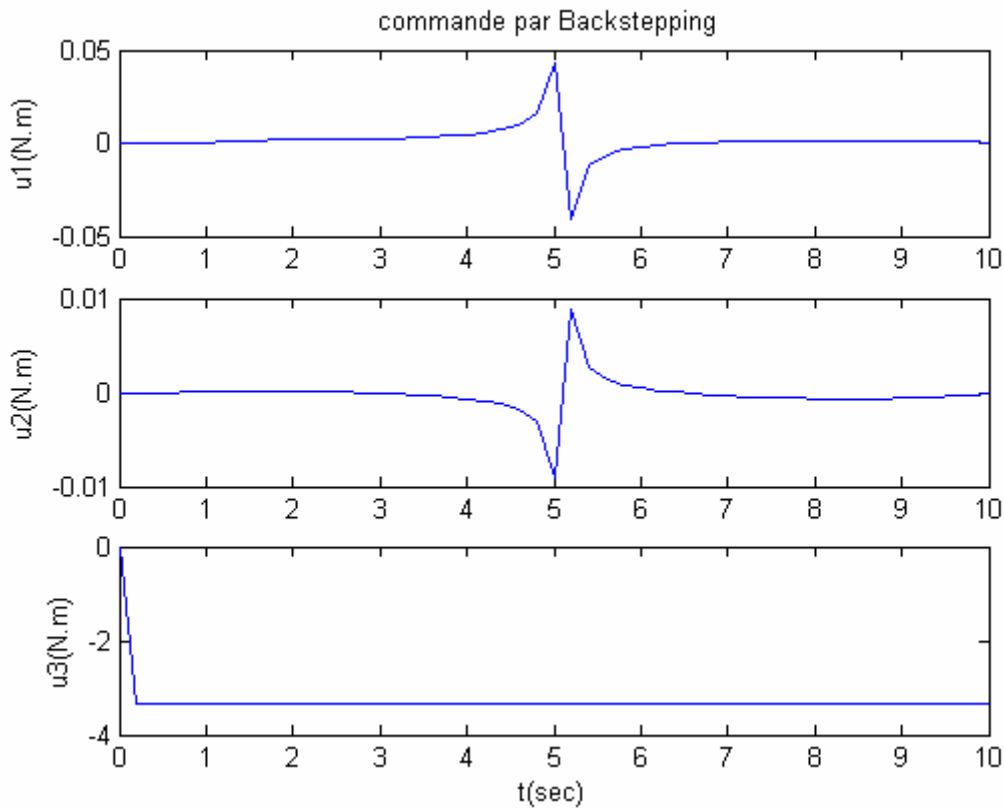


Figure III.18 Signaux de la commande par Backstepping

III.3.3.3 Modes de glissement

Dans cette partie, nous reprenons les équations du système données par (III.3.52) :

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = v - \ddot{y}_r \end{cases}$$

Ces deux équations peuvent se mettre sous la forme compacte suivante :

$$\dot{e} = f + gv \quad (\text{III.3.62})$$

Avec :

$$f^T = (e_2 \quad -\ddot{y}_r) \text{ et } g^T = (0 \quad 1) \quad (\text{III.3.63})$$

La surface de glissement choisie est la suivante :

$$S = \lambda e_1 + \dot{e}_1 \quad (\text{III.3.64})$$

Avec :

$$\lambda > 0 \quad (\text{III.3.65})$$

Et ce afin de satisfaire l'objectif de poursuite sur cette surface, à savoir :

$$\begin{aligned} S = 0 &\Rightarrow \dot{e}_1 = -\lambda e_1 \\ &\Rightarrow e_1 = e_1(0) e^{-\lambda t} \end{aligned} \quad (\text{III.3.66})$$

La convergence de l'erreur de sortie vers zéro -et donc la convergence de la sortie vers la consigne correspondante- est assurée.

Nous appliquons la commande satisfaisant l'attractivité ainsi que l'invariance :

$$\begin{aligned} v &= \left[\frac{\partial S}{\partial e} g \right]^{-1} \left\{ -\frac{\partial S}{\partial e} f - K \operatorname{sign}(S) \right\} \\ &= -K \operatorname{sign}(S) - \lambda (z_2 - \dot{y}_r) + \ddot{y}_r \end{aligned} \quad (\text{III.3.67})$$

Reste à voir si la dynamique réduite sur la surface de glissement est stable ou pas, celle-ci étant de dimension $n-m=1$. Cette dynamique est effectivement stable car sur la surface de glissement, on a :

$$\dot{e}_1 = e_2 = -\lambda e_1 \quad (\text{III.3.68})$$

L'application de la commande précédente nous donne :

$$\dot{e}_2 = [-K \operatorname{sign}(\lambda e_1 + e_2) - \lambda e_2 + \ddot{y}_r] - \ddot{y}_r \Rightarrow \dot{e}_2 = -\lambda e_2 \quad (\text{III.3.69})$$

Et comme :

$$e_2 \dot{e}_2 = -\lambda e_2^2 < 0 \quad (\text{III.3.70})$$

Alors l'erreur e_2 converge vers zéro. La dynamique réduite sur la surface est donc stable.

Simulation 1 :

La simulation de la commande sur les trois sorties avec : $\lambda = 2$ et $K = 0.001$ donne les résultats suivants :

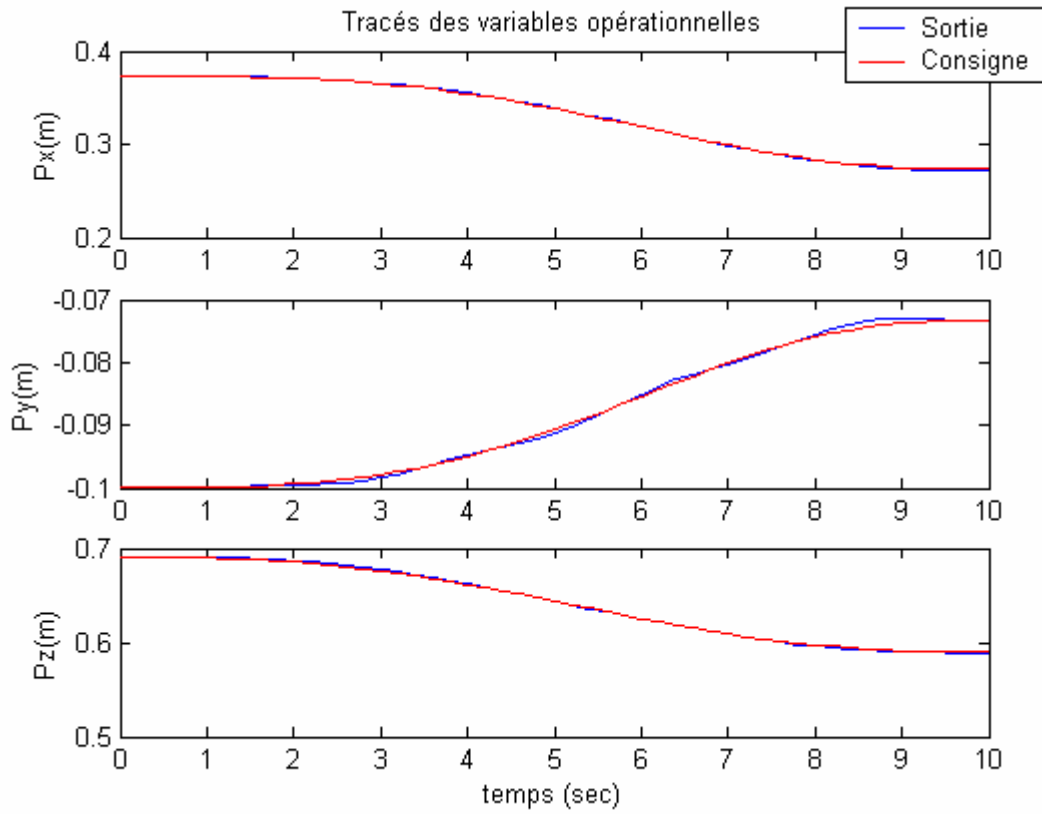


Figure III.19 Résultats de la poursuite après application de la commande par modes de glissement

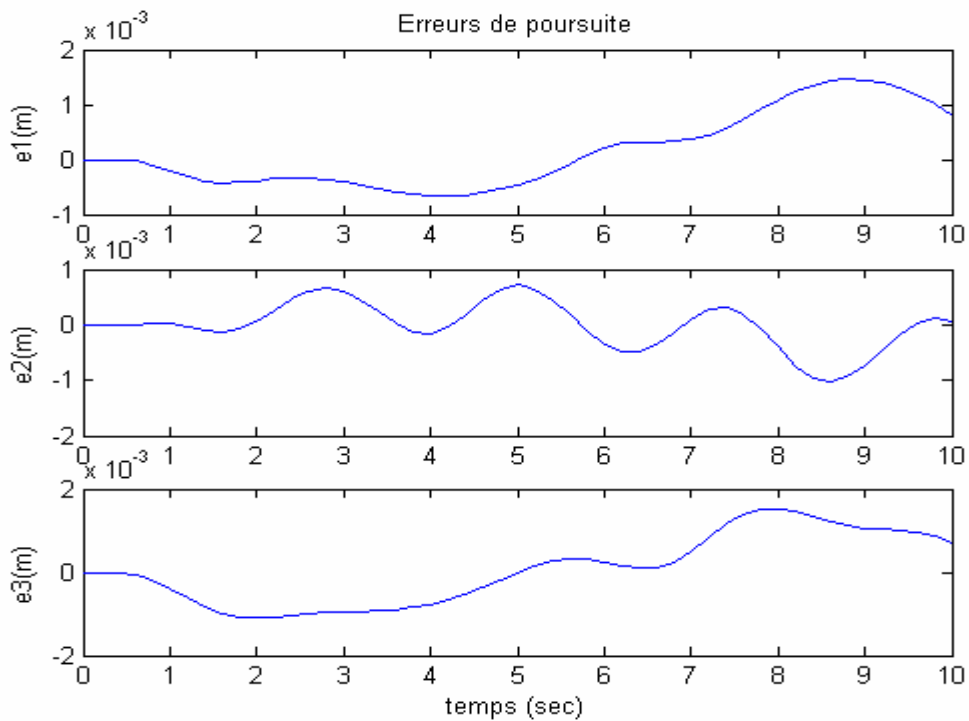


Figure III.20 Erreurs de poursuite correspondant à l'application de la commande par modes de glissement

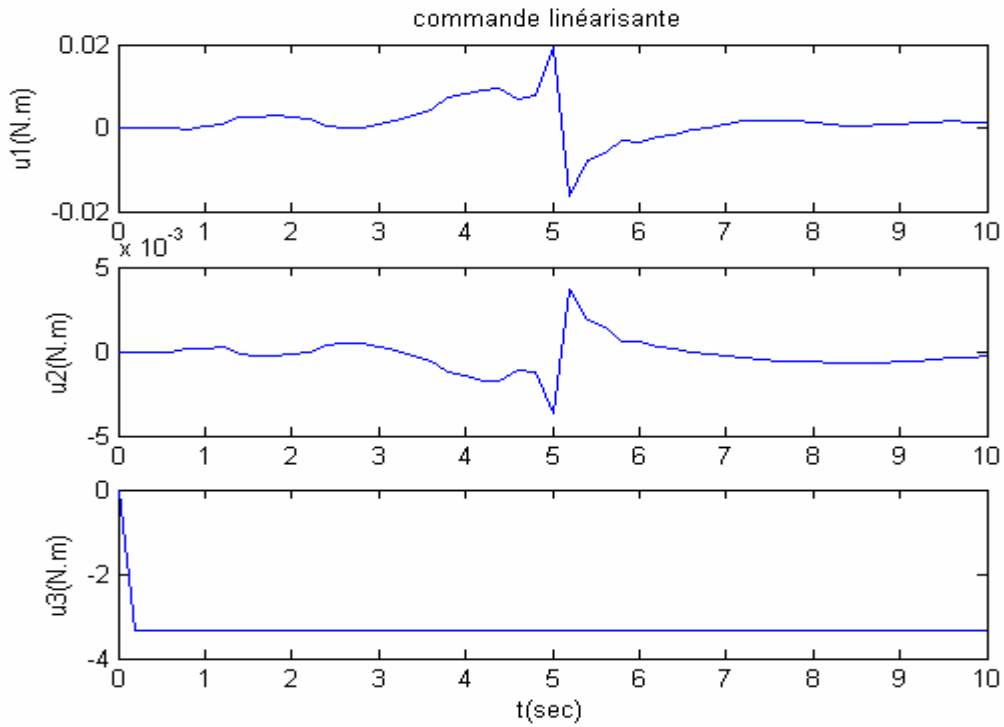


Figure III.21 Signaux de commande réels correspondant à l'application de la commande par modes de glissement

Remarque : On peut améliorer ou bien détériorer les résultats de la poursuite en agissant sur les deux paramètres K et λ et ce en fonctions des objectifs à atteindre.

On remarque que pour des valeurs non appropriées et mal dimensionnées de λ et K , les sorties peuvent carrément diverger, comme nous l'indique les deux exemples de simulation suivants :

- 1- Pour $\lambda = 5$ et $K = 0.001$:

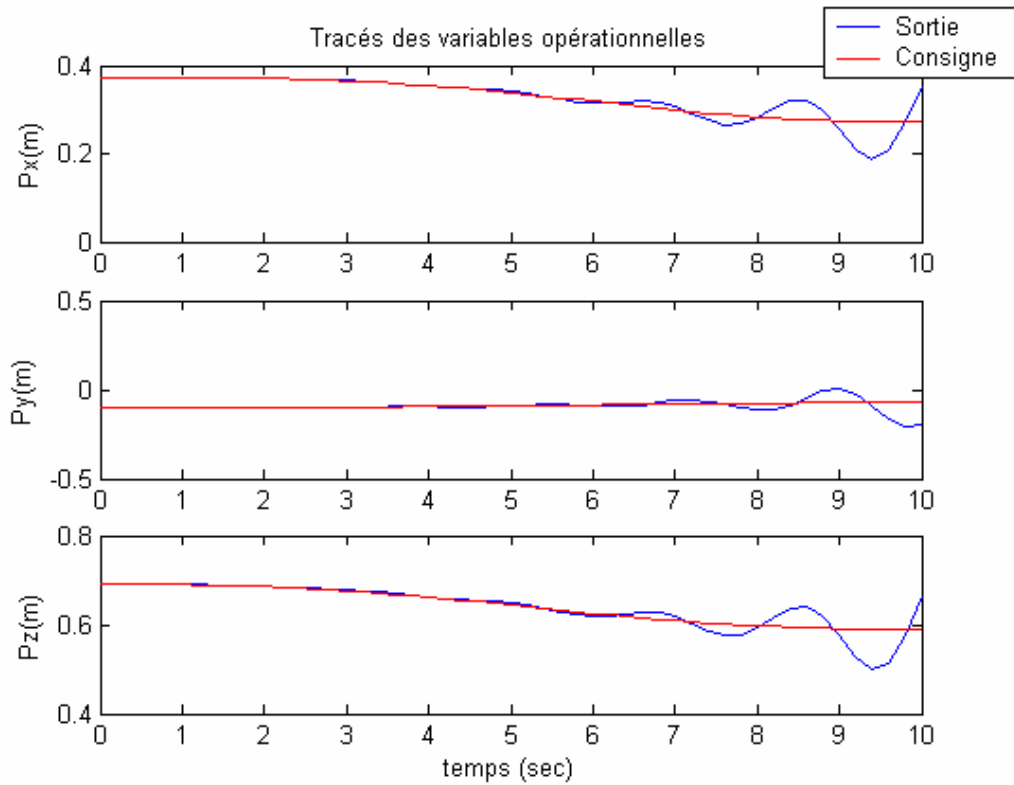


Figure III.22 Divergence des sorties après changement des paramètres de la commande par modes de glissement ($\lambda = 5$ et $K = 0.001$)

2- Pour $\lambda = 2$ et $K = 1$:

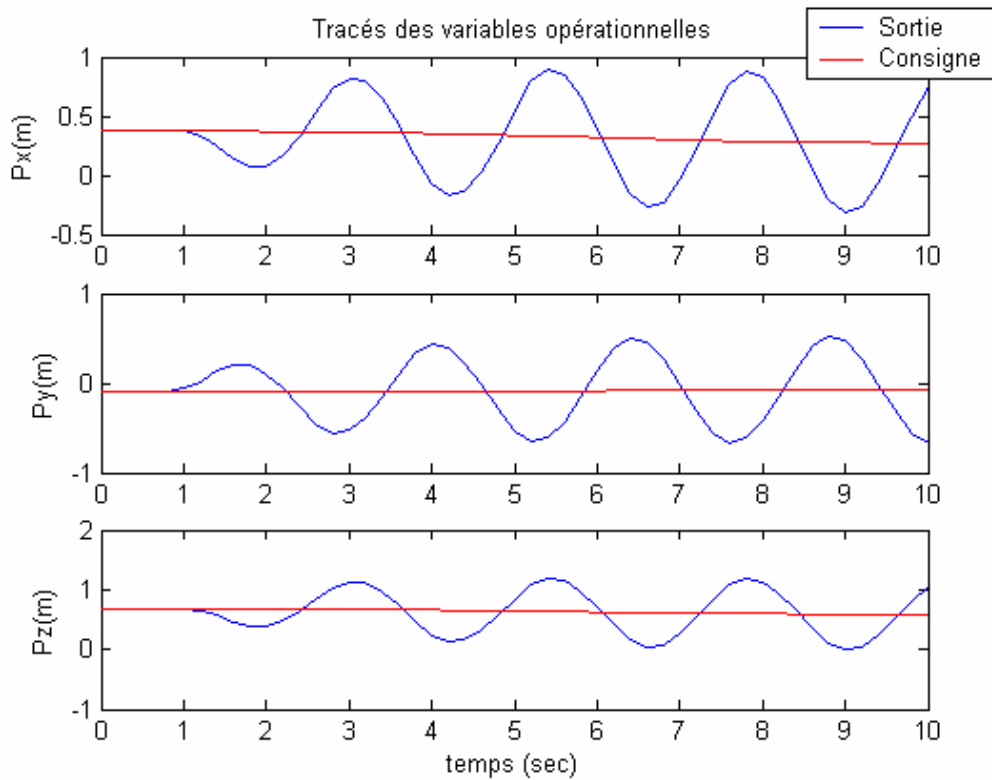


Figure III.23 Divergence des sorties après changement des paramètres de la commande par modes de glissement ($\lambda = 2$ et $K = 1$)

Analyses des techniques utilisées :

Nous superposons dans un même graphe les résultats des méthodes précédentes qui ont donné de bons résultats (RE avec référence modifiée, Backstepping et modes glissants MG). Nous obtenons :

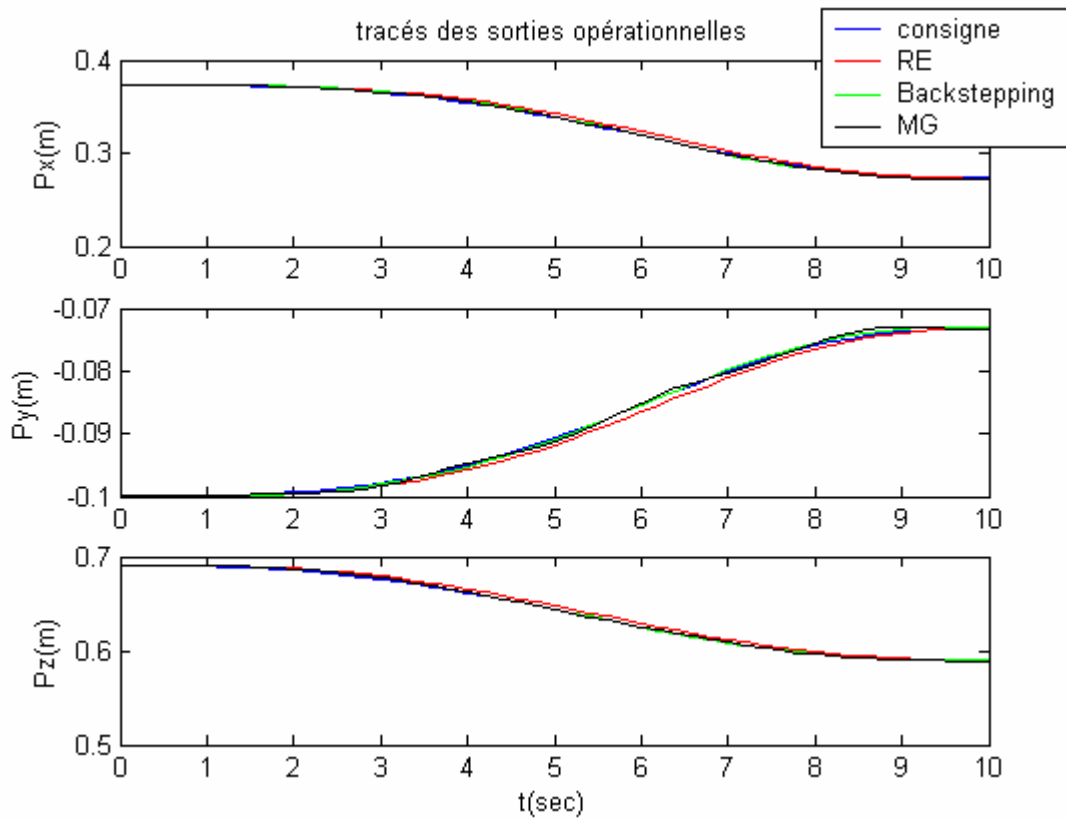


Figure III.24 Comparaison des résultats de poursuite entre les commandes de la géométrie différentielle

Nous voyons que les trois méthodes donnent des résultats comparables avec une bonne poursuite de trajectoire. Analysons les signaux d'erreurs :

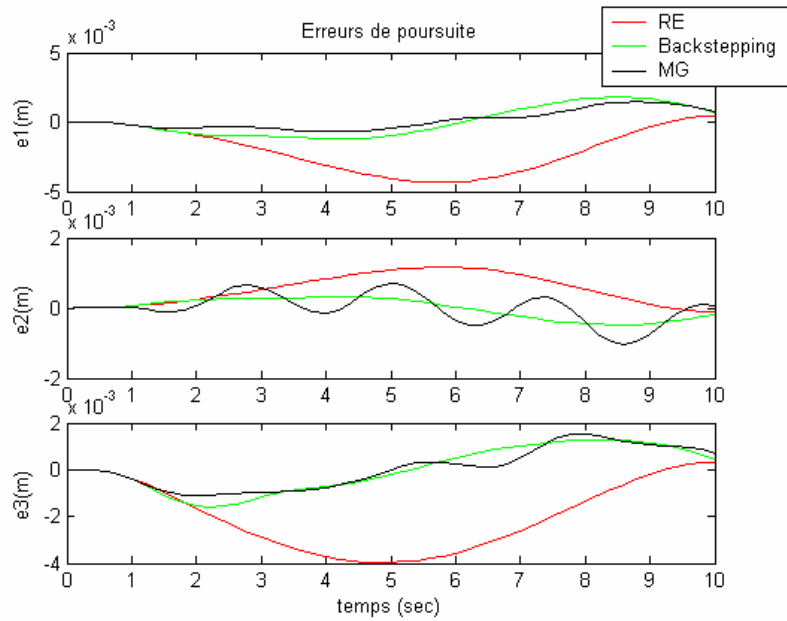


Figure III.25 Comparaison des erreurs de poursuite entre les commandes de la géométrie différentielle

Les erreurs sont de l'ordre du millimètre ce qui est un bon résultat.

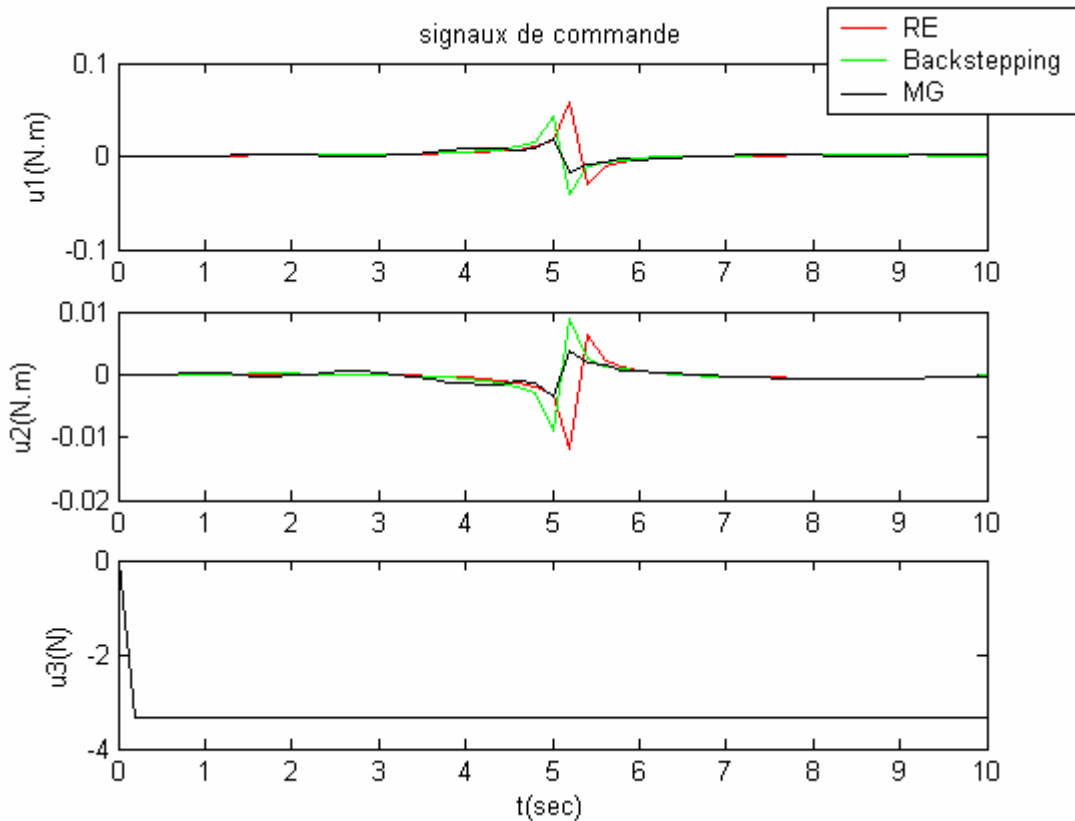


Figure III.26 Comparaison entre les commandes de la géométrie différentielle

Les signaux de commande ont des amplitudes assez faibles à part pour la troisième articulation car la tige supporte en permanence le poids du troisième corps.

III.4 COMMANDE PAR COUPLE CALCULE

Reprenons les équations du modèle dynamique données par l'équation (II.4.21) :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + g(q) = \tau \quad (\text{III.4.1})$$

Pour simplifier, nous la reprenons sous la forme suivante :

$$M(q)\ddot{q} + H(q, \dot{q}) = u \quad (\text{III.4.2})$$

Avec :

$$H(q, \dot{q}) = C(q, \dot{q})\dot{q} + B\dot{q} + g(q) \quad (\text{III.4.3})$$

Et :

$u = \tau$: le signal de commande du système.

Nous choisissons la commande non linéaire suivante :

$$u = M(q)v + H(q, \dot{q}) \quad (\text{III.4.4})$$

Celle-ci nous permet de réaliser à la fois le découplage et la linéarisation du système. En effet, en la remplaçant dans l'équation (III.4.2), et pour une matrice M inversible, nous obtenons :

$$\ddot{q} = v \quad (\text{III.4.5})$$

Où : v est la nouvelle commande du système.

Nous retrouvons le système du double intégrateur. Ce sont les mêmes équations que celles qui ont été retrouvées lors de la linéarisation par géométrie différentielle, à la seule différence que la sortie est la variable articulaire. La commande est donc faite dans l'espace articulaire directement.

Résultats des simulations :

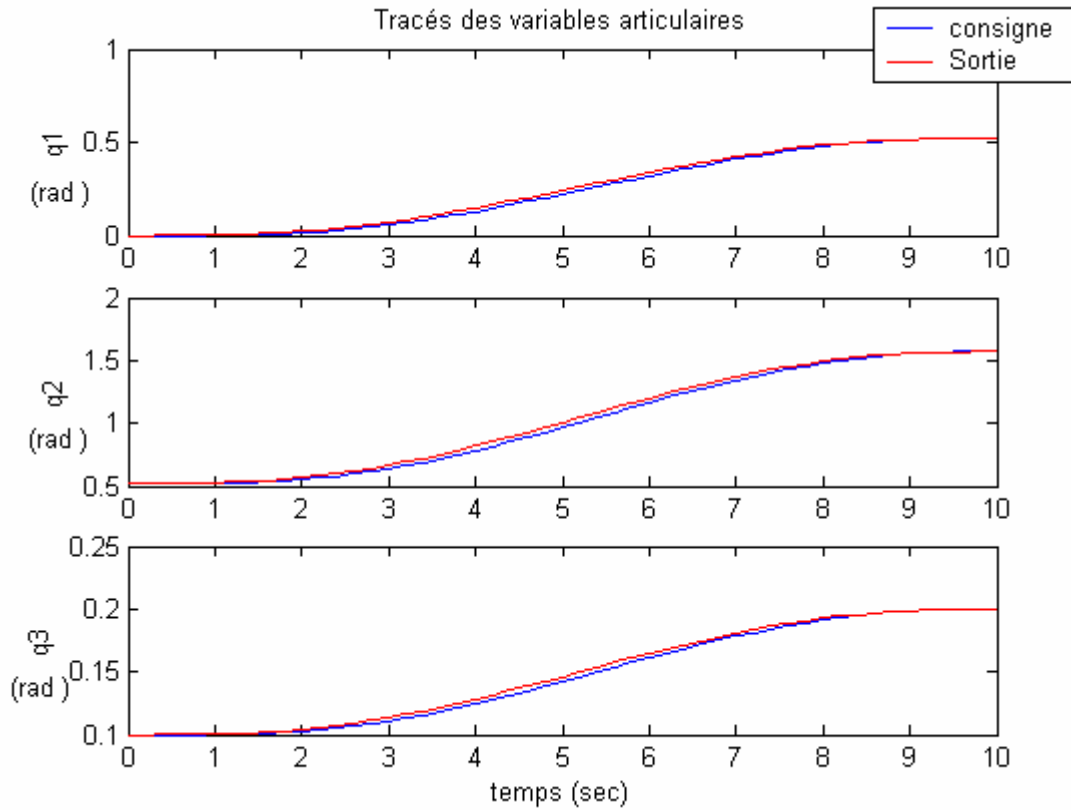


Figure III.27 Résultats de la poursuite dans l'espace articulaire après application de la commande par couple calculé

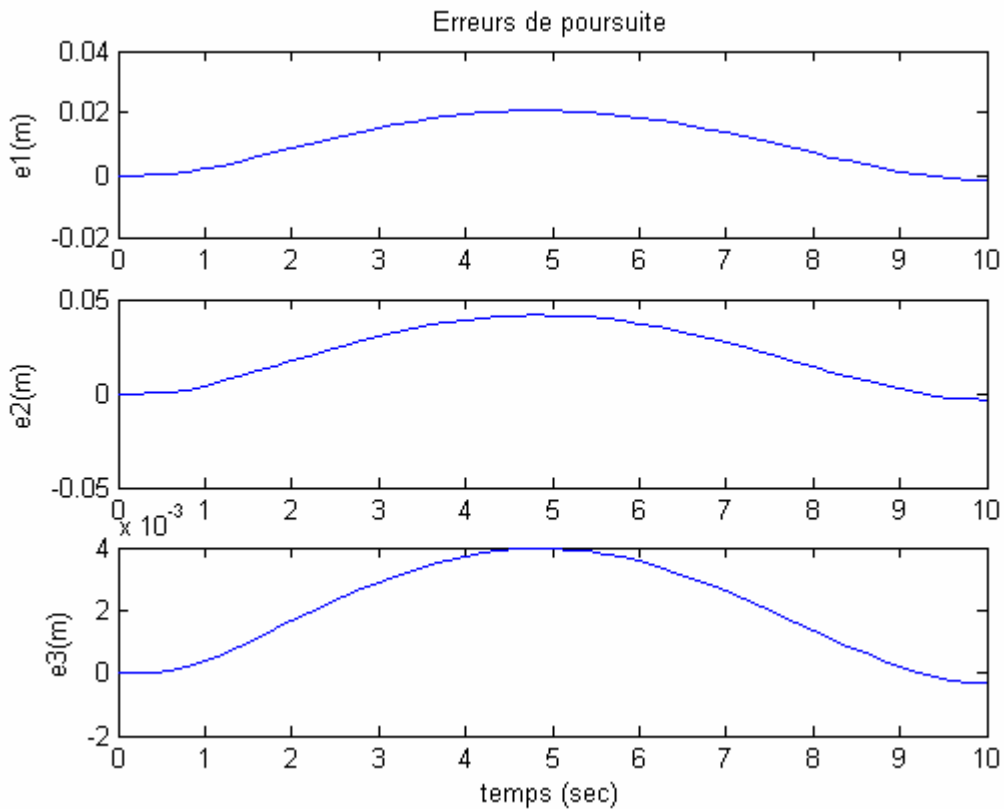


Figure III.28 Erreurs de poursuite correspondant à l'application de la commande par couple calculé

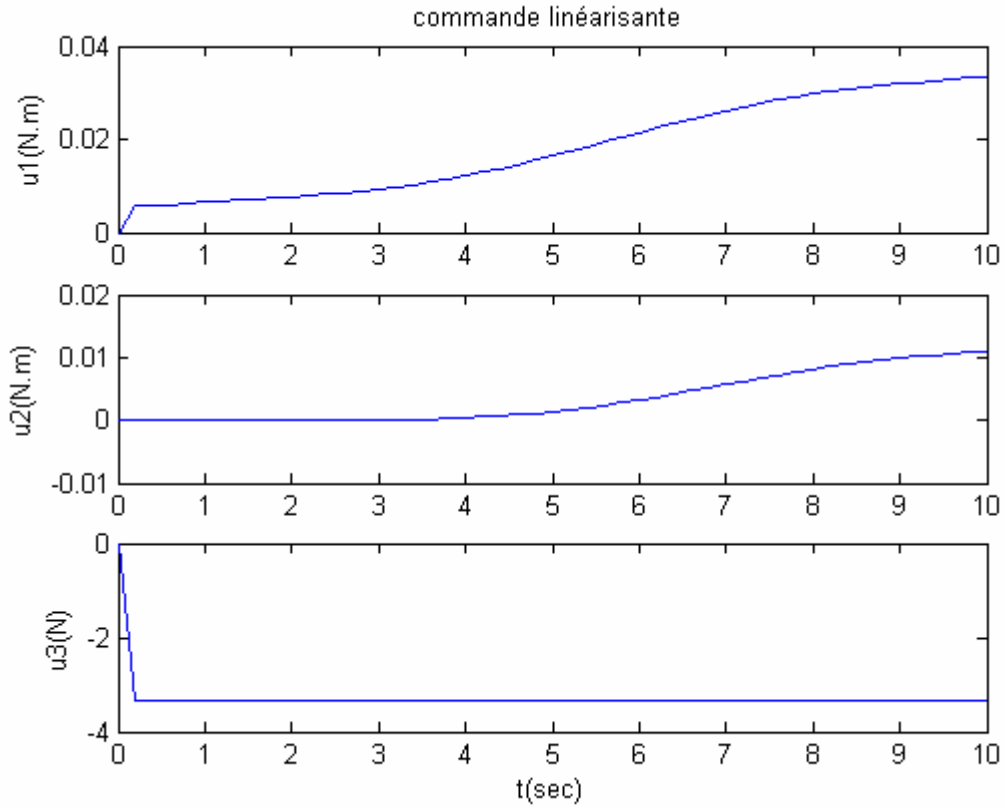


Figure III.29 Signaux de commande réels correspondant à l'application de la commande par couple calculé

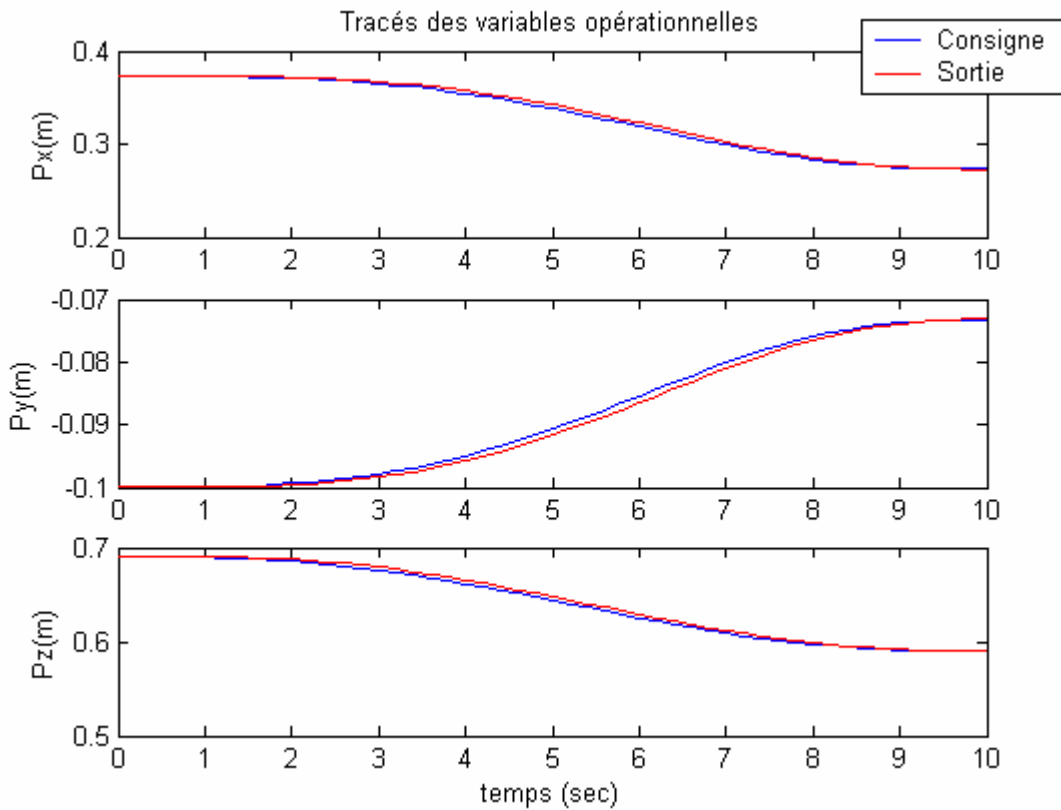


Figure III.30 Résultats de la poursuite dans l'espace opérationnel après application de la commande par couple calculé

La commande par couple calculé donne de très bons résultats qui sont comparables à ceux obtenus après linéarisation par géométrie différentielle.

III.5 COMMANDE DECENTRALISEE

Le principe de cette méthode est de commander chaque axe du robot indépendamment des autres. Le problème n'est plus celui de la commande d'un système MIMO, mais plutôt celui de la commande de trois systèmes SISO. Dans ce cas, les effets des couplages dus aux mouvements des autres articulations du robot sont soit ignorés, soit considérés comme étant des perturbations –ceci est le cas car ce sont des entrées sur lesquelles on ne peut agir.

Tout le problème de la commande consiste donc à synthétiser un régulateur réalisant le double objectif qui est *la poursuite de trajectoire* ainsi que *le rejet de perturbations*.

Rappelons que le système SISO à commander est le moteur à courant continu dont le modèle simplifié est donné par les équations (II.5.8) et (II.5.9) :

$$\left\{ \begin{array}{l} \Theta_m(s) = \frac{\frac{K_m}{R}}{s(J_m s + B_m + \frac{K_d K_m}{R})} V(s) \\ q_k = r\theta_{mk} + q_k^i \quad k = 1, \dots, n \end{array} \right.$$

Reprenons les deux équations (II.4.21) et (II.5.7) :

Afin de simplifier le problème de commande, nous posons :

$$\begin{aligned} q_k &= r\theta_{mk} & k &= 1, \dots, n \\ \tau_{l_k} &= \tau_k \end{aligned} \tag{III.4.1}$$

Autrement dit, nous travaillons avec des valeurs relatives pour les variables articulaires. Et nous considérons que le couple généré par chaque moteur correspond au couple de charge qu'il entraîne.

Les équations du mouvement (II.4.21) et (II.5.7) du robot manipulateur s'écrivent alors :

$$\sum_{j=1}^n d_{jk}(q) \ddot{q}_j + \sum_{i,j} c_{ijk}(q) \dot{q}_i \dot{q}_j + g_k(q) = \tau_k \quad (\text{III.4.2})$$

$$J_m \ddot{\theta}_{m_k} + \left(B_m + \frac{K_b K_m}{R} \right) \dot{\theta}_{m_k} = \frac{K_m}{R} V_k - r_k \tau_k, \quad k = 1, \dots, n \quad (\text{III.4.3})$$

L'équation (III.4.3) représente la dynamique des actionneurs, elle est issue de l'équation (II.5.7). L'équation (III.4.2) quand à elle représente la forme générale du modèle dynamique du robot donnée par l'équation (II.4.20).

L'approche la plus simple pour commander le robot consiste à considérer le terme non linéaire τ_k entrant dans l'équation (III.4.3) et défini par (III.4.2) comme étant une perturbation pour le moteur.

Nous synthétiserons ainsi un correcteur indépendant pour chaque articulation en utilisant le modèle donné par (III.4.3). C'est le principe de la *commande décentralisée*. L'avantage de cette approche est sa simplicité car le modèle est linéaire.

Notons que pour des moteurs à rapports de réduction élevés, l'influence des couplages non linéaires est très faible. Ceci est dû au fait que le terme les représentant à savoir τ_k est pondéré par r_k dans l'équation (III.4.3). Ce qui rend la validité de cette technique de commande d'autant plus importante, les moteurs utilisés ayant des rapports de réduction assez élevés.

Cela dit, et pour des actionneurs à transmission directe (sans réducteurs) ou encore pour un asservissement à des vitesses élevées, les couplages non linéaires auront un effet beaucoup plus important sur les performances du système. Considérer donc le terme τ_k uniquement comme une perturbation sera insuffisant et causera à coup sûr de grandes erreurs de poursuites.

La loi de commande utilisée est alors donnée par :

$$V_1(t) = K_p^1 e_1(t) + K_i^1 \int_0^t e_1(u) du + K_d^1 \dot{y}_1(t)$$

$$V_2(t) = K_p^2 e_2(t) + K_i^2 \int_0^t e_2(u) du + K_d^2 \dot{y}_2(t)$$

$$V_3(t) = K_p^3 e_3(t) + K_i^3 \int_0^t e_3(u) du + K_d^3 \dot{y}_3(t)$$

Il s'agit de contrôleurs PID de forme particulière. Cette forme est la même que celle implémentée dans le contrôleur UMAC du robot que nous présenterons au chapitre IV.

Les différents gains proportionnels, intégrales et dérivés des trois régulateurs peuvent être choisis indépendamment les uns des autres. Les résultats de simulation obtenus par tâtonnement pour la première variable articulaire (pour les valeurs de gains $K_p^1 = 8, K_i^1 = 0.001, K_d^1 = 0.05$) sont :

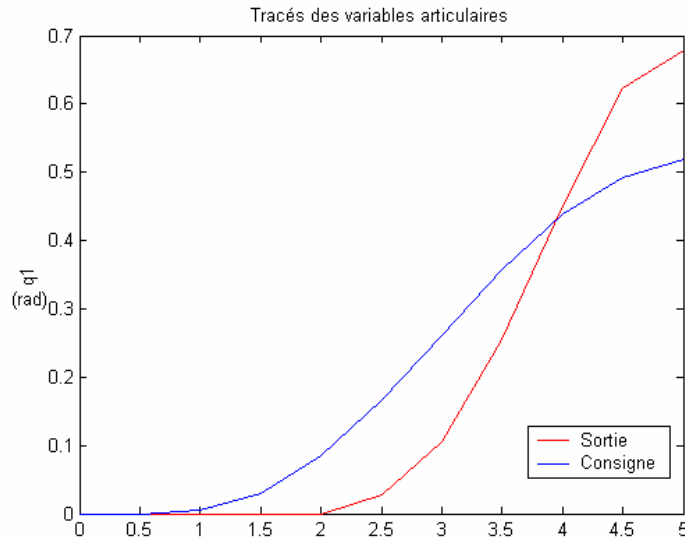


Figure III.31 Résultats de poursuite pour la commande PID

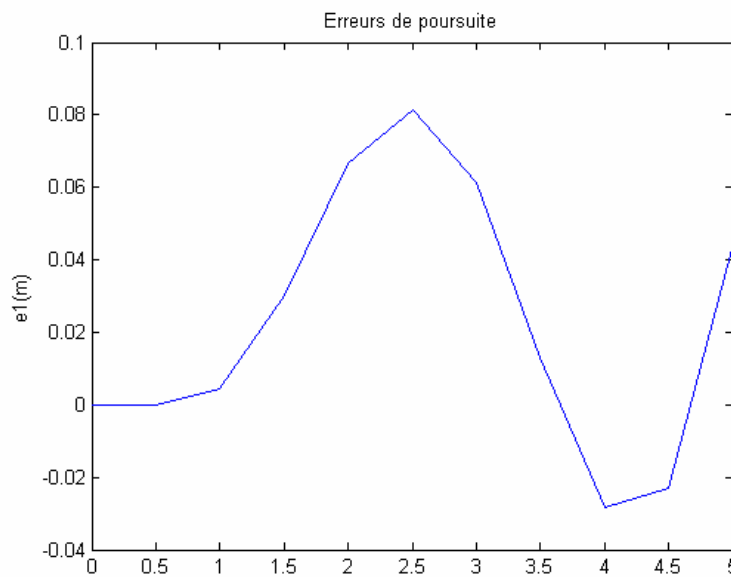


Figure III.32 Erreur de poursuite pour la commande PID

L'ajout d'un terme Feedforward est aussi possible. Son introduction donne de meilleurs résultats de poursuite car le système a des informations sur l'accélération du système.

$$V_1(t) = K_p^1 e_1(t) + K_i^1 \int_0^t e_1(u) du + K_d^1 \dot{e}_1(t) + K_{al} \ddot{y}_{1r}$$

$$V_2(t) = K_p^2 e_2(t) + K_i^2 \int_0^t e_2(u) du + K_d^2 \dot{e}_2(t) + K_{a2} \ddot{y}_{2r}$$

$$V_3(t) = K_p^3 e_3(t) + K_i^3 \int_0^t e_3(u) du + K_d^3 \dot{e}_3(t) + K_{a3} \ddot{y}_{3r}$$

Les résultats de simulation pour la première variable articulaire (pour les valeurs de gains $K_p^1 = 8, K_i^1 = 0.001, K_d^1 = 0.05$ et $K_{a1} = 2$) sont :

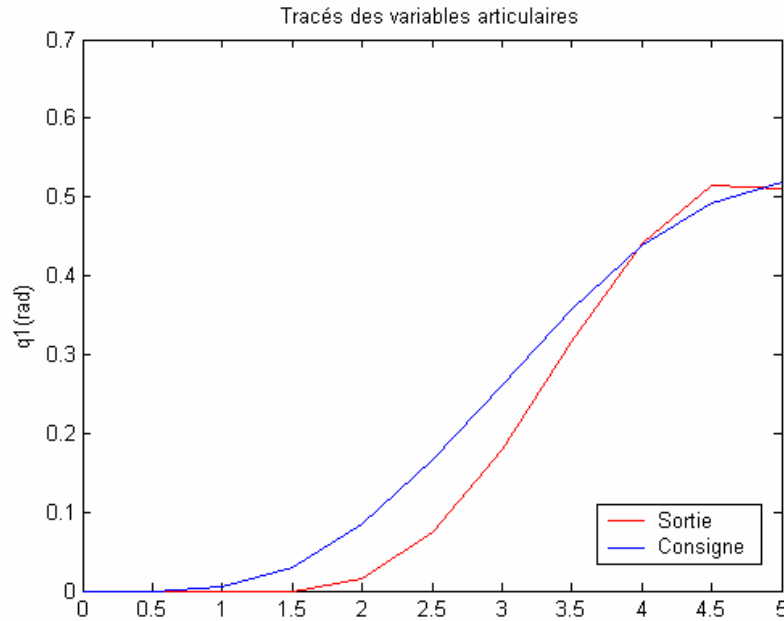


Figure III.33 Résultats de poursuite pour la commande PID après ajout du terme Feedforward

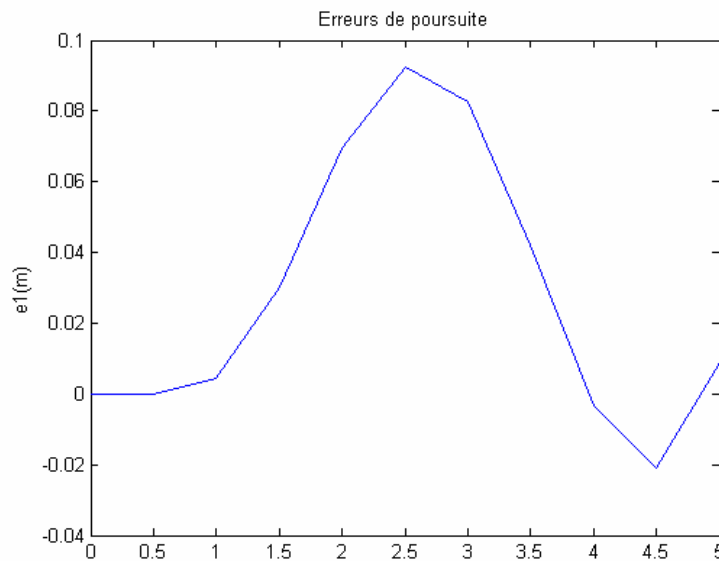


Figure III.34 Erreur de poursuite pour la commande PID après ajout de l'action Feedforward

Nous voyons bien que la réponse est meilleure après ajout du terme d'anticipation. Cependant, la réponse du système n'est pas aussi bonne que les méthodes développées précédemment. Une optimisation des coefficients du régulateur peut être faite pour améliorer les performances du système.

III.6 ANALYSE DES RESULTATS OBTENUS

D'une façon générale, les résultats de poursuite obtenus sont très satisfaisants pour les deux premières méthodes utilisées, à savoir la géométrie différentielle et le couple calculé.

La dernière technique qui est la commande décentralisée n'a pas donné les mêmes résultats de poursuite.

Afin de mieux comparer les résultats obtenus, nous superposons sur les mêmes graphes les résultats de poursuite des deux premières techniques. A noter que ce sont les méthodes commandant le système linéarisé par retour d'état qui sont présentées.

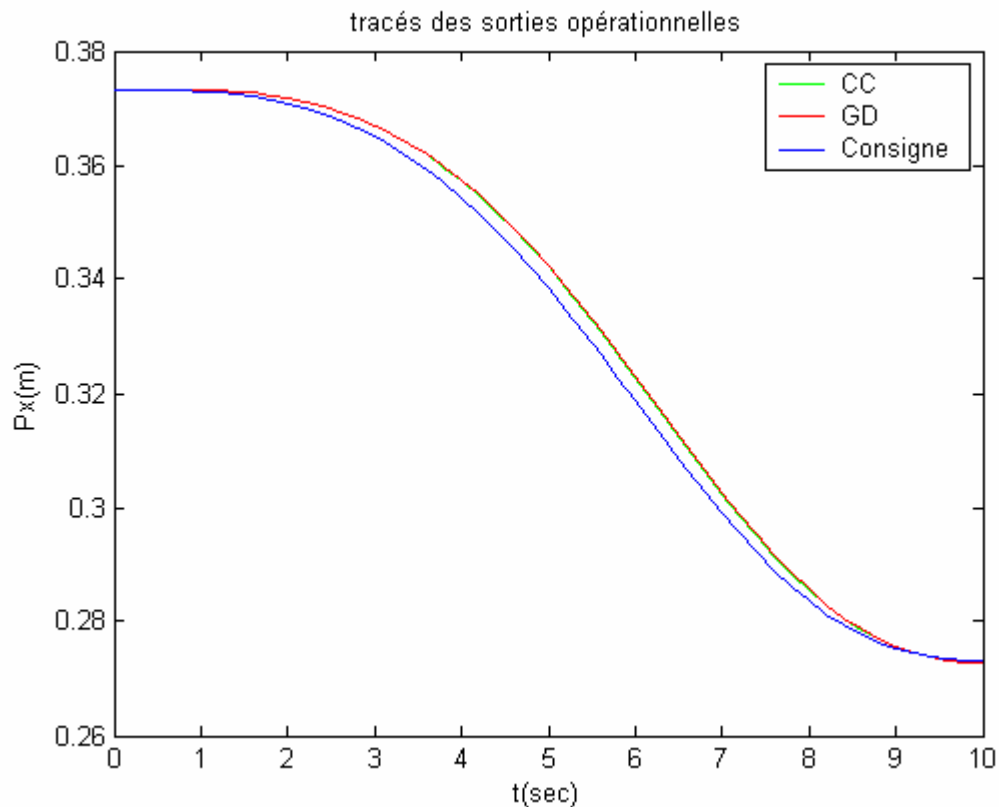


Figure III.35 Comparaison des résultats de poursuite (sortie 1) entre la commande par couple calculé et celle de la géométrie différentielle

Du point de vue poursuite de référence, nous voyons bien que les résultats des deux méthodes sont quasiment identiques. En fait, les deux techniques sont presque similaires les deux opérant une linéarisation du système puis sa commande. C'est cette similitude que nous retrouvons dans les résultats de la figure III.33.

Intéressons-nous maintenant aux signaux d'erreurs :

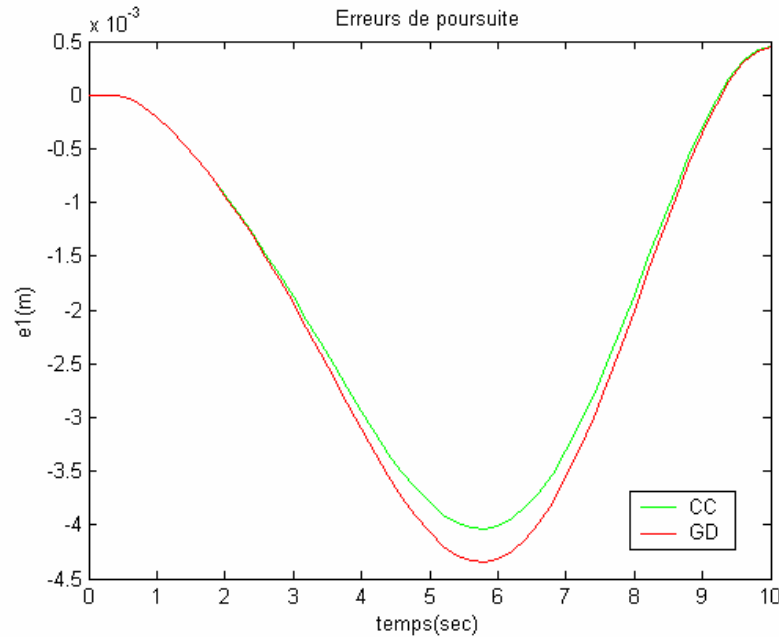


Figure III.36 Comparaison des erreurs de poursuite (sortie1) entre la commande par couple calculé et le retour d'état avec référence modifiée

Les erreurs sont de l'ordre du millimètre, ce qui est quand même un très bon résultat qui pourrait être amélioré en jouant sur les paramètres du régulateur. Le prix à payer étant la forme et la valeur de la commande auxquelles nous nous intéressons justement à la figure III.37 :

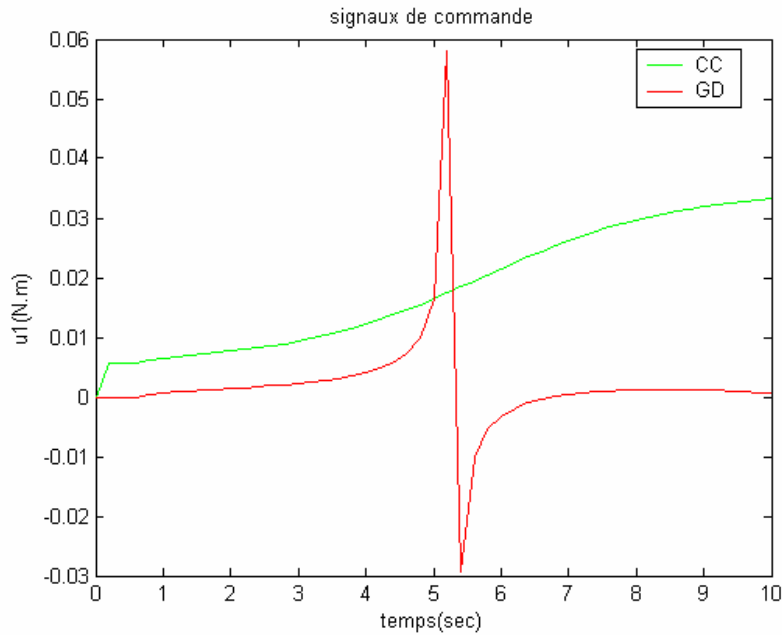


Figure III.37 Comparaison entre les commandes (sortie1) par couple calculé et par retour d'état avec référence modifiée

Cette figure est importante, car elle nous permet de voir que le signal de commande du couple calculé est nettement meilleur que celui de la géométrie différentielle dans notre cas. Ceci veut dire que pour les mêmes performances, nous avons une commande plus douce pour la méthode du couple calculé. Elle n'exige pas des efforts trop importants pour les actionneurs.

Ceci ne signifie en aucun cas que l'on préférera cette méthode systématiquement à la première. La méthode de la géométrie différentielle effectue la commande directement dans l'espace opérationnel ce qui est très avantageux pour certaines tâches.

III.5 CONCLUSION

Nous avons développé au cours de ce chapitre deux approches pour la commande du robot. Ce sont l'approche MIMO et SISO.

Pour la première, deux types de commandes ont été synthétisés. Ce sont la commande linéarisante par géométrie différentielle et la commande par couple calculé. Toutes deux linéarisent tout d'abord le système, puis le commandent dans l'espace opérationnel pour la première et articulaire pour la seconde. Les résultats de poursuite simulés sont très

satisfaisants tant du point de vue précision que celui des énergies et des profils des signaux de commandes qui sont physiquement réalisables.

La seconde approche est beaucoup plus simple. C'est une commande de type décentralisé où l'on commande trois systèmes linéaires de type SISO. L'intérêt des simulations que nous avons faites est qu'elles s'approchent du modèle réel car la structure du correcteur PID utilisé est identique à celle implémentée dans le contrôleur du robot. De même que les signaux de perturbation qui utilisent les équations du modèle dynamique.

Les résultats obtenus ne sont pas aussi bons que ceux des techniques précédentes. Cependant, cette technique est très utilisée car la loi de commande est très simple et l'optimisation des gains du régulateur conduit à des résultats satisfaisants.

CHAPITRE IV

PLANIFICATION DE TACHES ET IMPLEMENTATION

CHAPITRE IV

PLANIFICATION DE TACHES ET IMPLEMENTATION

IV.1 INTRODUCTION

Au cours des chapitres précédents, nous avons pu réaliser et modéliser le robot SCARA. Nous avons ensuite synthétisé et simulé un certain nombre de commandes à partir de ses différents modèles.

Reste maintenant à programmer le robot afin qu'il suive des trajectoires désirées. C'est le problème de planification puis d'implémentation de tâches que nous abordons au cours de ce chapitre.

IV.2 PLANIFICATION DES TACHES

IV.2.1 TACHE 1: PICK AND PLACE

L'une des tâches les plus simples à programmer est la tâche *Pick and place* – expression anglo-saxonne qui veut dire saisir et poser. C'est en fait une tâche de positionnement où le robot doit passer par un certain nombre de points dans l'espace opérationnel. Le but est de déplacer des objets d'une position à une autre. La trajectoire entre les deux points peut être quelconque en cas d'absence d'obstacles.

Dans ce type de tâche, l'outil employé est un dispositif servant à saisir des objets (pince, électroaimant...etc.). Pour la tâche que nous planifions, nous supposons qu'il n'y a pas d'obstacles et ce pour simplifier la simulation. Le schéma descriptif de la tâche est donné par la figure IV.1 :

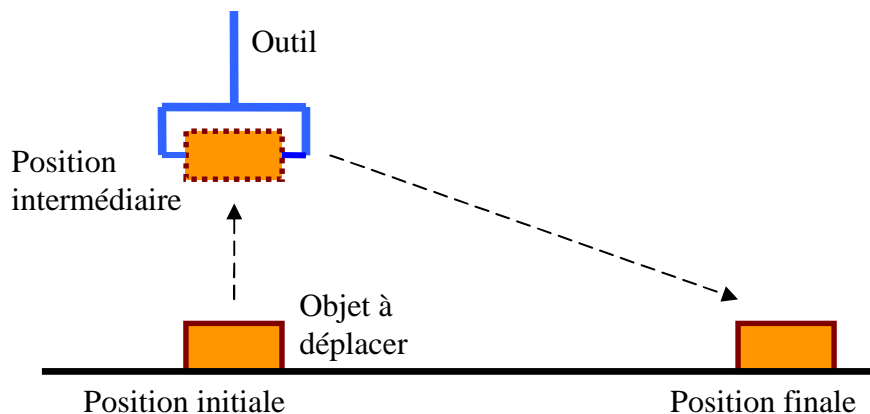


Figure IV.1 Tâche 1: *Pick and place*

Le robot doit déplacer un objet d'une position initiale placé sur un plan horizontal à une position finale située sur le même plan. Le déplacement du robot comportera deux phases :

- La première phase consiste à soulever l'objet du plan horizontal à une position intermédiaire.
- La seconde où le robot devra atteindre sa position finale pour y déposer l'objet.

Les simulations ont été faites pour les paramètres suivants :

$$[P^{initiale}]^T = [-0.237 \quad 0.298 \quad 0.162]$$

$$[P^{intermédiaire}]^T = [-0.237 \quad 0.298 \quad 0.413]$$

$$[P^{finale}]^T = [0.362 \quad -0.118 \quad 0.162]$$

Les configurations du robot correspondant à chacune de ces positions sont données par la figure IV.2 :

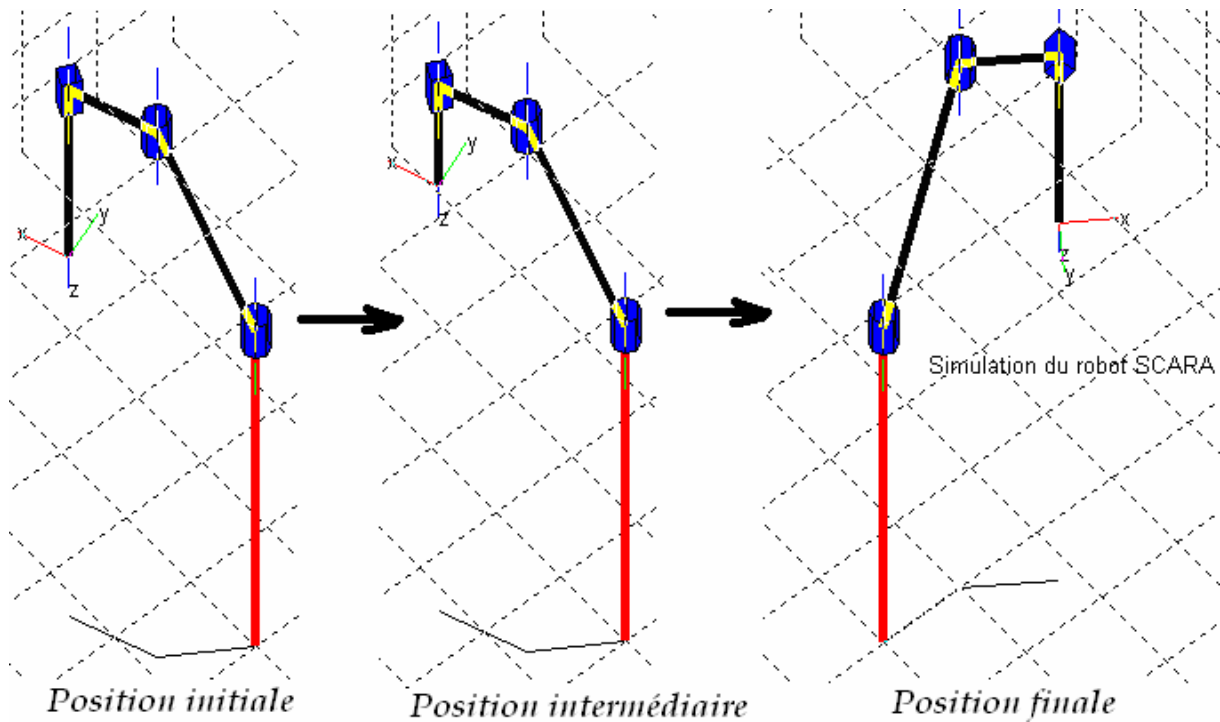


Figure IV.2 Configurations initiale, intermédiaire et finale du robot SCARA exécutant la tâche Pick and place

Puisque la génération de trajectoire est faite entre deux points avec trajectoire quelconque entre les deux, il est intéressant de la visualiser dans l'espace opérationnel :

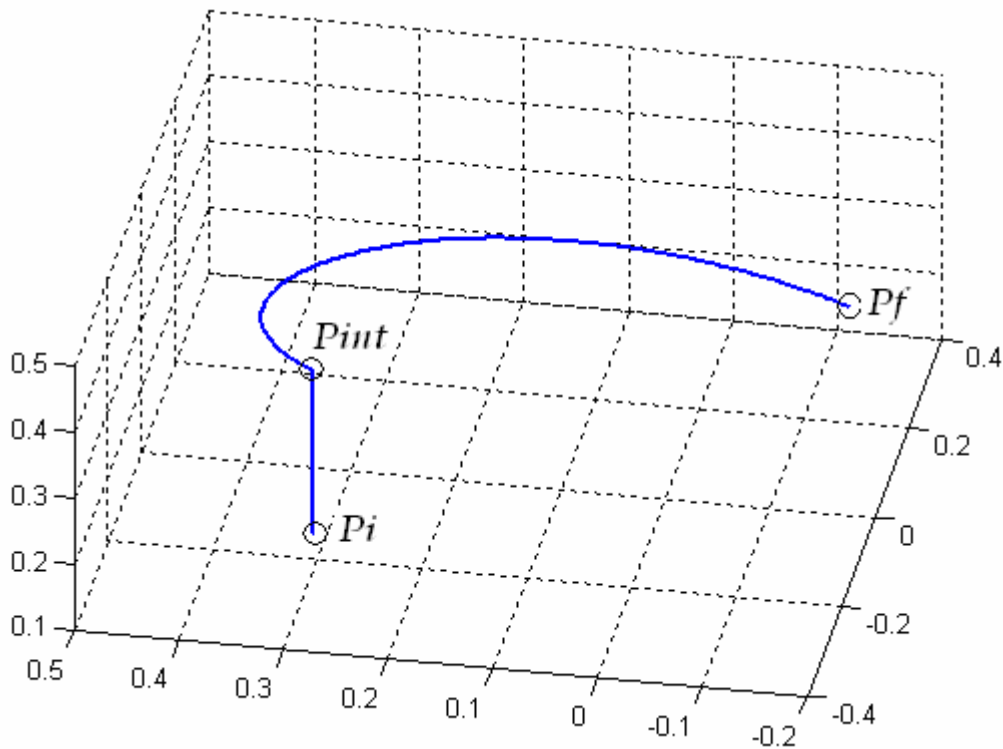


Figure IV.3 Trajectoire opérationnelle dans l'espace 3D

Durant la première phase, la trajectoire du robot est une droite verticale ce qui est tout à fait logique car le robot est en train de soulever l'objet. C'est donc uniquement l'articulation prismatique qui est à l'action.

Pour la seconde phase, la trajectoire est quelconque comme planifié lors de la génération de mouvement.

V.2.2 TACHE 2: ECRITURE D'UN MOT ENTIER

La tâche à effectuer consiste à écrire le mot ENP sur un plan horizontal (plan d'écriture). L'outil utilisé est donc un stylo, feutre, crayon ou tout autre élément servant à écrire.

Pour cela, nous devons donc traduire cette tâche en équations mathématiques.

La démarche que nous avons adoptée peut être résumée en ces points principaux :

- 1- diviser chaque lettre en parties élémentaires (segments de droites).
- 2- distinguer les tronçons se trouvant sur le plan d'écriture et ceux se trouvant hors ce plan. (pour ces derniers, effectuer une génération de trajectoire simple entre deux

points avec trajectoire opérationnelle quelconque entre ces points, tout en définissant les profils de q , \dot{q} et \ddot{q}).

Pour les tronçons se trouvant sur le plan d'écriture :

- 3- définir l'abscisse curviligne s pour chaque tronçon.
- 4- imposer un profil de variation pour s , \dot{s} et \ddot{s} .
- 5- exprimer les position, vitesse et accélération opérationnelles en fonction de s , \dot{s} et \ddot{s} .
- 6- traduire le problème du domaine opérationnel en un problème dans le domaine articulaire.
- 7- calculer la commande proprement dite.

Appliquons donc les différents points énoncés précédemment :

- 1- La division des lettres en segments de droites se fait suivant le schéma de la figure IV.4 sur papier millimétré :

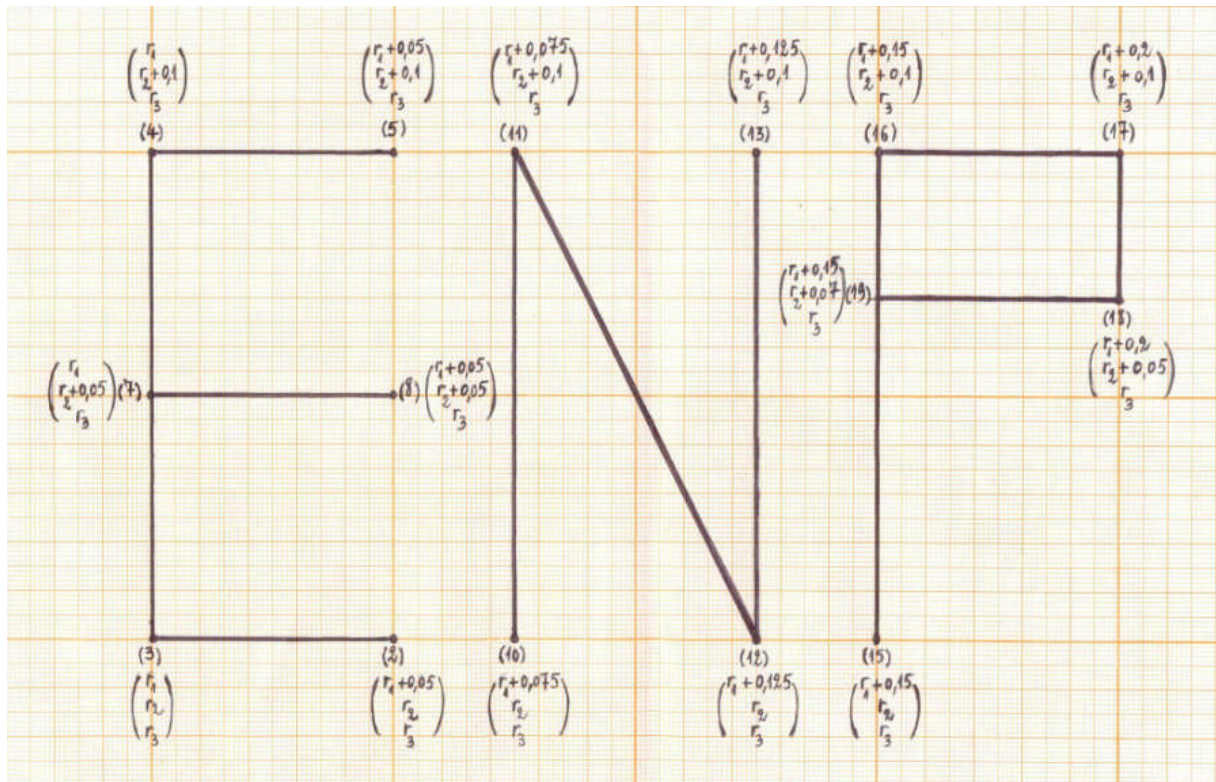


Figure IV.4 Segmentation du mot ENP

- 2- La tâche se constitue désormais de plusieurs tâches élémentaires résumées dans le tableau IV.1 qui traduit la figure IV.5, et dans lesquels nous détaillons les numérotations des points ainsi que la situation de chaque segment par rapport au plan d'écriture.

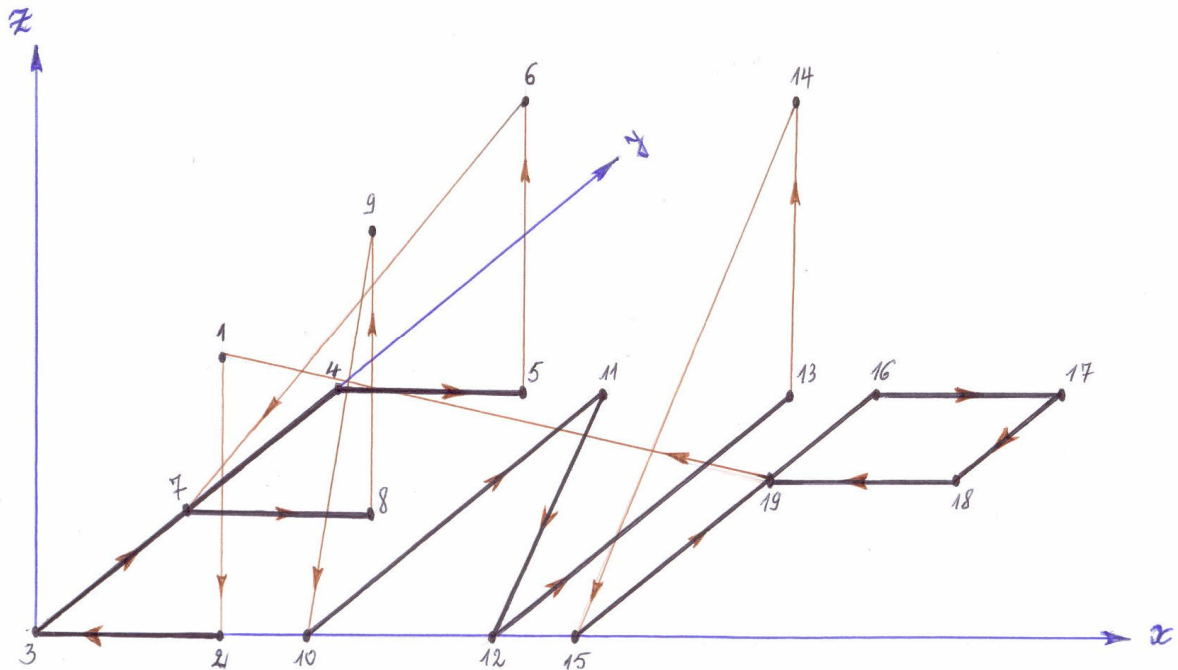


Figure IV.5 Trajectoires opérationnelles empruntées par l'outil

Tâche	Sens des segments dans le plan (xoy)	Situation par rapport au plan d'écriture	Tâche	Sens des segments dans le plan (xoy)	Situation par rapport au plan d'écriture
1 → 2	–	Hors	11 → 12	Incliné	Sur
2 → 3	$(x'ox)$	Sur	12 → 13	$(y'oy)$	Sur
3 → 4	$(y'oy)$	Sur	13 → 14	–	Hors
4 → 5	$(x'ox)$	Sur	14 → 15	–	Hors
5 → 6	–	Hors	15 → 16	$(y'oy)$	Sur
6 → 7	–	Hors	16 → 17	$(x'ox)$	Sur
7 → 8	$(x'ox)$	Sur	17 → 18	$(y'oy)$	Sur
8 → 9	–	Hors	18 → 19	$(x'ox)$	Sur
9 → 10	–	Hors	19 → 1	–	Hors
10 → 11	$(y'oy)$	Sur			

Tableau IV.1 Positionnement des différents tronçons par rapport au plan d'écriture

3- Nous définissons par la suite l'abscisse curviligne s comme moyen de décrire le parcours de chaque tronçon. Nous plaçons au début du parcours de chaque tronçon un repère de travail (o, x, y, z) . Pour le cas du tronçon incliné, nous schématiserons l'emplacement du repère associé.

Nous avons trois types de segments pour lesquels nous définissons l'abscisse s :

- Segment de droite suivant $(x' o x)$
- Segment de droite suivant $(y' o y)$
- Segment de droite inclinée d'équation $y = ax + b$

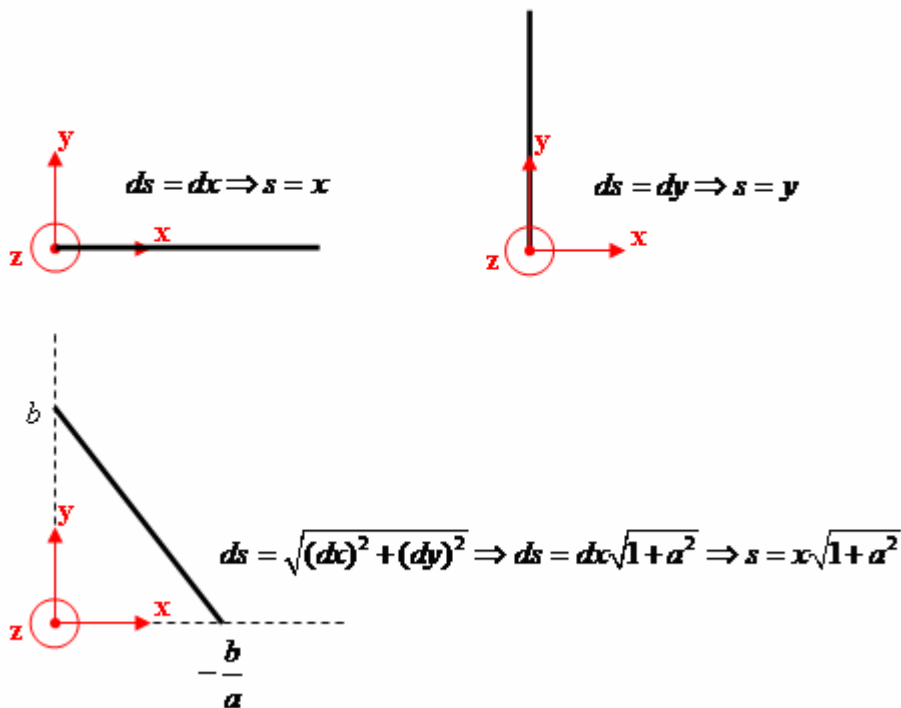


Figure IV.6 Différents segments avec leur abscisse curviligne correspondante

4- Le profil de variation de s en fonction du temps doit vérifier les conditions suivantes :

$$\begin{cases} s(0) = \dot{s}(0) = \ddot{s}(0) = \dot{s}(t_f) = \ddot{s}(t_f) = 0 \\ s(t_f) = D \end{cases}$$

Avec :

D : La distance parcourue le long du tronçon.

Nous choisissons un polynôme d'ordre 5. En plus de satisfaire ces 6 conditions, il a l'avantage de présenter des profils de vitesse et d'accélération continus. Nous arrivons après développement des calculs aux expressions :

$$\begin{cases} s(t) = \left(\frac{10D}{t_f^3}\right)t^3 - \left(\frac{15D}{t_f^4}\right)t^4 + \left(\frac{6D}{t_f^5}\right)t^5 \\ \dot{s}(t) = \left(\frac{30D}{t_f^3}\right)t^2 - \left(\frac{60D}{t_f^4}\right)t^3 + \left(\frac{30D}{t_f^5}\right)t^4 \\ \ddot{s}(t) = \left(\frac{60D}{t_f^3}\right)t^2 - \left(\frac{180D}{t_f^4}\right)t^3 + \left(\frac{120D}{t_f^5}\right)t^4 \end{cases}$$

5- Maintenant que le profil de variation de s est imposé, il est important de revenir à la position opérationnelle dont la commande est notre objectif principal.

L'expression de la position opérationnelle P en fonction de s est une relation bijective et diffère selon le type de segment de droite :

Tronçon suivant (x' o x) :

Position opérationnelle	Vitesse opérationnelle	Accélération opérationnelle
$P(s) = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} s + P_x^i \\ P_y^i \\ P_z^i \end{pmatrix}$	$\dot{P}(s) = \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{pmatrix} = \begin{pmatrix} \dot{s} \\ 0 \\ 0 \end{pmatrix}$	$\ddot{P}(s) = \begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \\ \ddot{P}_z \end{pmatrix} = \begin{pmatrix} \ddot{s} \\ 0 \\ 0 \end{pmatrix}$

Tronçon suivant (y' o y) :

Position opérationnelle	Vitesse opérationnelle	Accélération opérationnelle
$P(s) = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} P_x^i \\ s + P_y^i \\ P_z^i \end{pmatrix}$	$\dot{P}(s) = \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{pmatrix} = \begin{pmatrix} 0 \\ \dot{s} \\ 0 \end{pmatrix}$	$\ddot{P}(s) = \begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \\ \ddot{P}_z \end{pmatrix} = \begin{pmatrix} 0 \\ \ddot{s} \\ 0 \end{pmatrix}$

Tronçon incliné :

Position opérationnelle	Vitesse opérationnelle	Accélération opérationnelle
$P(s) = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{1+a^2}}s + P_x^i \\ \frac{a}{\sqrt{1+a^2}}s + b + P_y^i \\ P_z^i \end{pmatrix}$	$\dot{P}(s) = \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{1+a^2}}\dot{s} \\ \frac{a}{\sqrt{1+a^2}}\dot{s} \\ 0 \end{pmatrix}$	$\ddot{P}(s) = \begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \\ \ddot{P}_z \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{1+a^2}}\ddot{s} \\ \frac{a}{\sqrt{1+a^2}}\ddot{s} \\ 0 \end{pmatrix}$

A partir des points qui viennent d'être développés, il ne reste plus qu'à traduire la position opérationnelle $P(s)$ à l'aide du MGI en fonction des coordonnées articulaires, et de ce fait, calculer l'une des commandes déjà étudiées dans le chapitre précédent.

La commande par modes de glissement étant celle qui a donné les meilleurs résultats de poursuite précédemment, c'est elle que nous avons choisie pour cette application.

Nous obtenons pour notre tâche les résultats suivants :

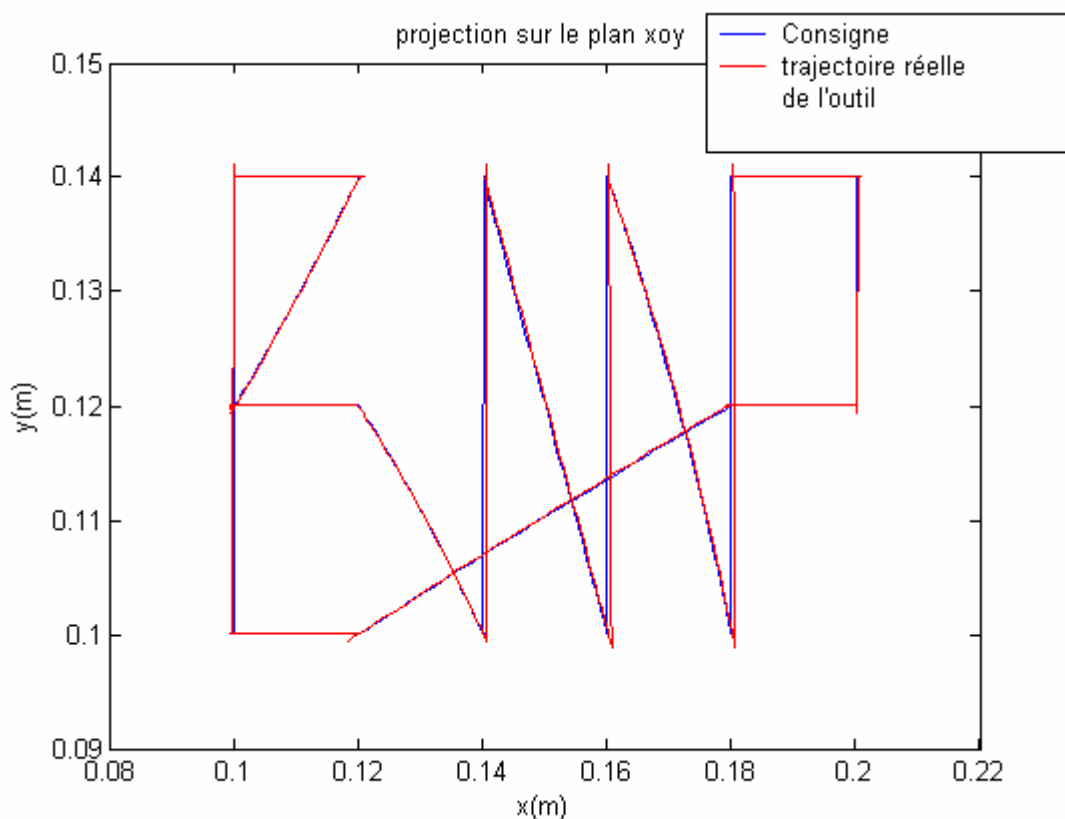


Figure IV.7 Projection de la trajectoire opérationnelle sur le plan d'écriture horizontal

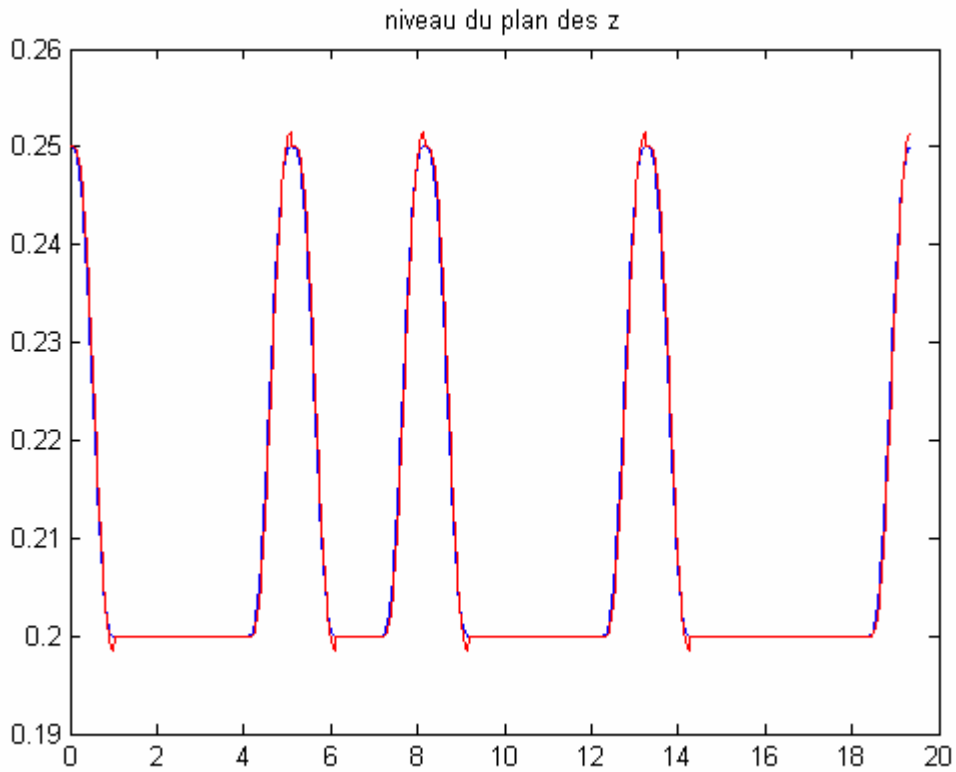


Figure IV.8 Position de l'outil par rapport au plan d'écriture

Dans l'espace tridimensionnel, nous obtenons :

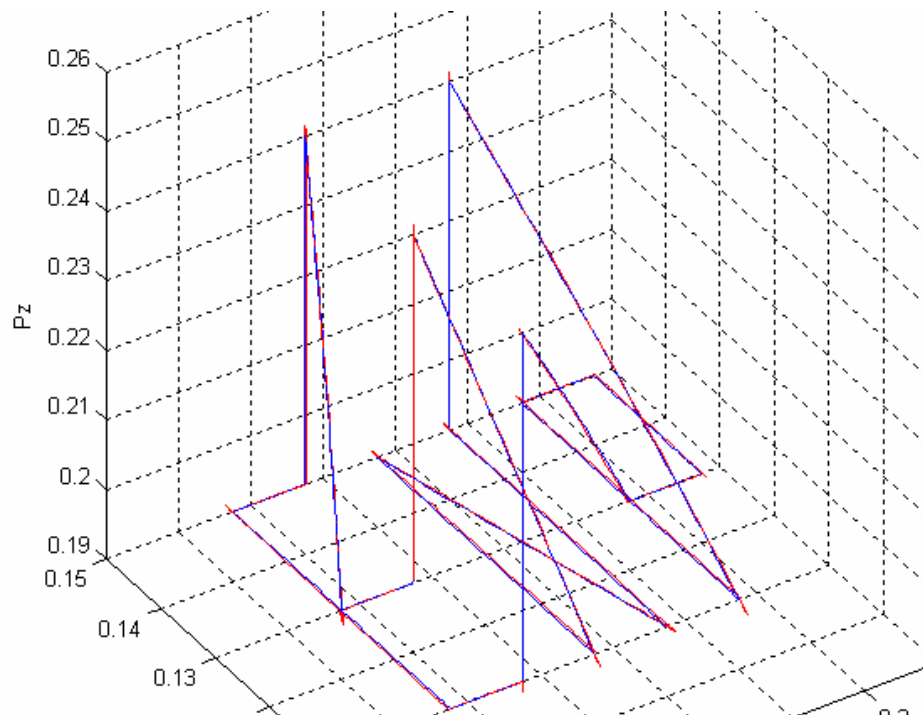


Figure IV.9 Trajectoire opérationnelle de l'outil dans l'espace 3D

Nous constatons les bons résultats de poursuite avec cependant de légers dépassements de l'ordre du millimètre (1 à 2 mm).

IV.2.3 TACHE 3 : ALPHABET ET ECRITURE DE PHRASES

Dans la tâche précédente, nous avons vu comment on pouvait écrire un mot en le définissant comme une trajectoire sur un plan dans l'espace opérationnel. Cette approche est intéressante et nous a permis d'obtenir de bons résultats.

Cependant, cette approche atteint très vite ses limites surtout si on veut que le robot écrive toutes sortes de mots ou encore des phrases entières. En effet, on devra à chaque fois calculer et introduire la trajectoire opérationnelle correspondante ce qui devient à la longue très laborieux. L'introduction d'une méthode plus facile et surtout plus flexible est donc nécessaire.

IV.2.3.1 Principe

Le principe de la méthode est très simple. Il s'agit de programmer un total de 18 droites -présentées dans la figure IV.10 - grâce auxquelles nous pourrons écrire n'importe quelle lettre de l'alphabet.

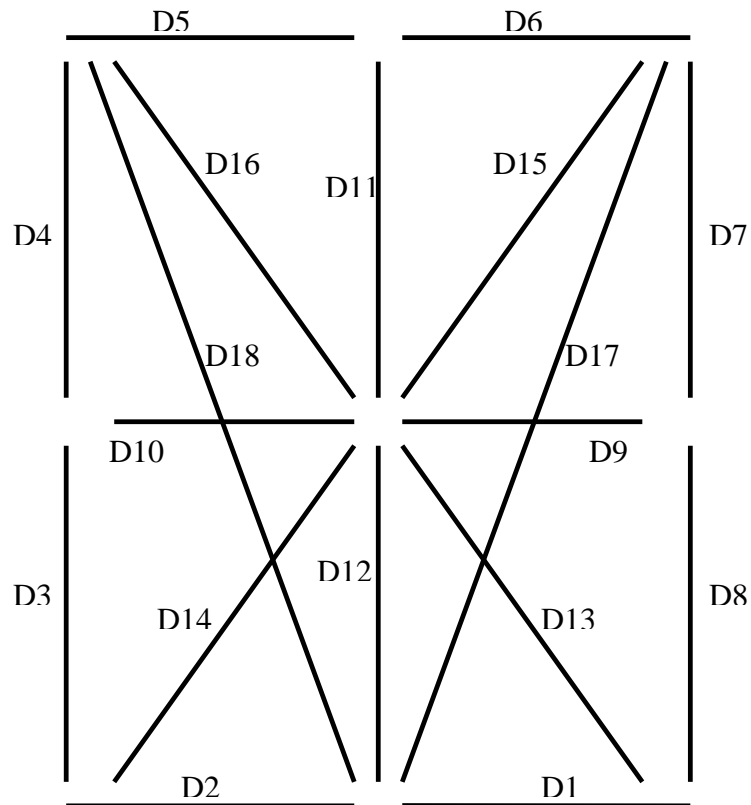


Figure IV.10 Droites à programmer afin d'exécuter toutes les lettres de l'alphabet

Par exemple, le programme d'écriture de la lettre A comportera les sous programmes de traçage des droites D3, D4, D5, D6, D7, D8, D9 et D10 on aura ainsi le résultat suivant :

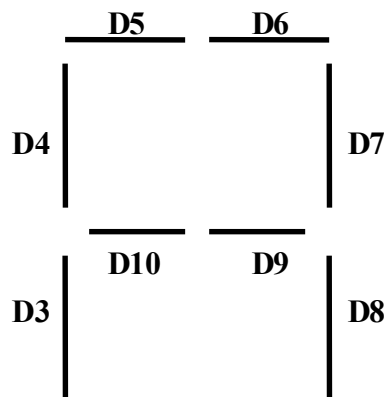
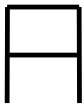
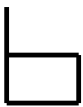

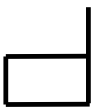

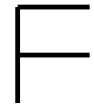
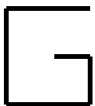
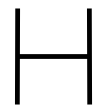






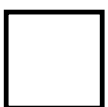
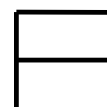
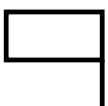



Figure IV.11 Ecriture de la lettre A

IV.2.3.2 Implémentation des lettres de l'alphabet et des chiffres

La programmation d'une lettre devient donc très facile, il suffit par exemple d'écrire un programme *LETA* pour la lettre A (*LETN* pour la lettre N par exemple) et de faire appel à

des sous programmes d'exécution de droites *DR1, DR2, DR3, ...DR18* correspondant à la lettre en question. Le tableau IV.2 spécifie le groupe de droites à implémenter associé aux lettres correspondantes.

Lettres et droites correspondantes	Lettre tracée	Lettres et droites correspondantes	Lettre tracée
A (LETA) 3-4-5-6-7-8-9-10		B (LETB) 1-2-3-4-8-9-10	
C (LETC) 1-2-3-4-5-6		D (LETD) 1-2-3-7-8-9-10	
E (LETE) 1-2-3-4-5-6-9-10		F (LETF) 3-4-5-6-9-10	
G (LETG) 1-2-3-4-5-6-8-9		H (LETH) 3-4-7-8-9-10	
I (LETI) 11-12		J (LETJ) 2-3-5-6-11-12	
K (LETK) 3-4-10-13-15		L (LETL) 1-2-3-4	
M (LETM) 3-4-7-8-15-16		N (LETN) 3-4-7-8-13-16	
O (LETO) 1-2-3-4-5-6-7-8		P (LETP) 3-4-5-6-7-9-10	
Q (LETQ) 4-5-6-7-8-9-10		R (LETR) 3-4-5-6-7-9-10-13	

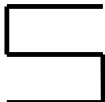
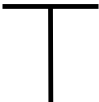
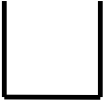




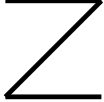
S (LETS) 1-2-4-5-6-8-9-10		T (LETT) 5-6-11-12	
U (LETU) 1-2-3-4-7-8		V (LETV) 17-18	
W (LETW) 3-4-7-8-13-14		X (LETX) 13-14-15-16	
Y (LETY) 12-15-16		Z (LETZ) 1-2-5-6-14-15	

Tableau IV.2 Programmation de l'alphabet

Pour les chiffres, leur programmation est la même que celle d'un afficheur 7 segments et est expliquée dans le tableau IV.3 :

Chiffres	Droites correspondantes	Chiffres	Droites correspondantes
1 (CH1)	7-8	2 (CH2)	1-6-7-9-12
3 (CH3)	1-6-7-8-9	4 (CH4)	7-8-9-11
5 (CH5)	1-6-8-9-11	6 (CH6)	1-6-8-9-11-12
7 (CH7)	6-7-8	8 (CH8)	1-6-7-8-9-11-12
9 (CH9)	1-6-7-8-9-11	0 (CH0)	1-6-7-8-11-12

Tableau IV.3 Programmation des chiffres

IV.2.3.3 Ecriture de phrases

Les lettres de l'alphabet étant programmées, le problème de leur écriture ne se pose plus. Reste alors à écrire des phrases entières et pour cela on doit ajouter les deux sous programmes suivants :

- Le premier servant à introduire un espace entre deux lettres du même mot. Pour séparer deux mots d'une même phrase, il ne reste plus qu'à exécuter ce sous-programme trois fois par exemple.
- Le second correspondant au retour chariot appelé « RC ».

La programmation de ces deux sous programmes a déjà été abordée car la génération de leur trajectoire est la même que celle de la tâche Pick and Place (soulever la pointe du stylo pour la déposer en un autre point du plan).

Remarque : Afin de simplifier la programmation, nous ne nous sommes pas intéressés aux caractères spéciaux (point, virgule, apostrophe...etc.).

IV.3 PRESENTATION DU CONTROLEUR UMAC [1]

IV.3.1 CONSTITUTION MATERIELLE ET FONCTIONNEMENT

UMAC veut dire *Universal Motion and Automation Controller*. C'est un contrôleur de mouvement multiaxe programmable. Il est constitué d'un système modulaire combinant puissance et flexibilité.

Les différentes cartes composant l'UMAC sont représentées dans la figure IV.12 :

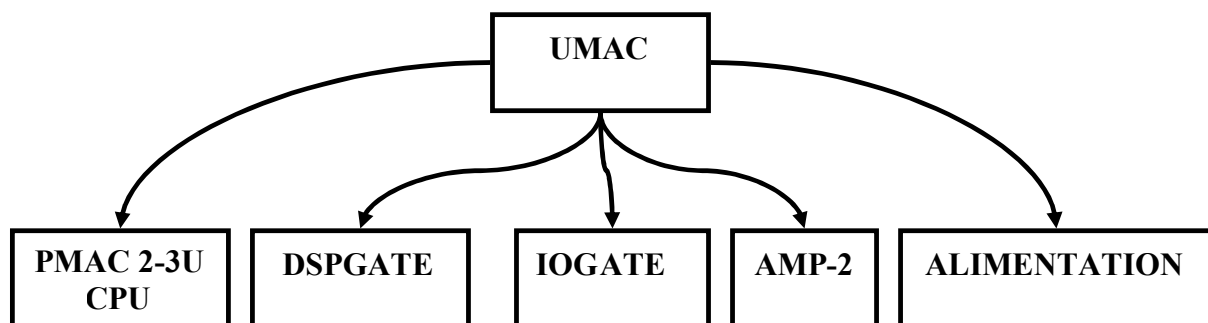


Figure IV.12 Cartes constituant l'UMAC

- **PMAC 2-3U CPU** : Elle gère le système et contrôle les moteurs. Elle est constituée d'un DSP Motorola 56303, de mémoires pour les micro programmes et d'un connecteur RS 232/422.

- **2 DSPGATE** : Ils constituent une interface entre le CPU et les signaux d'E/S pour les moteurs. Chacune contrôle 4 axes, et chaque axe envoie des signaux de sortie et de commande pour les amplificateurs ;
- **IOGATE** : Il adapte les signaux d'E/S digitaux avec isolation optique.
- **AMP-2** : L'interface dispose de 8 amplificateurs répartis sur deux cartes. Chaque amplificateur a pour rôle de rendre les signaux de commande de la DSPGATE exploitable.
- **Alimentation** : Elle fournit des sorties 5V, 15V et -15V continues pour l'alimentation des circuits numériques et analogiques.

Le fonctionnement de l'UMAC obéit aux étapes suivantes :

- Le programme de mouvement est élaboré sur un PC puis chargé dans une zone mémoire de la carte TURBO PMAC2-3U CPU grâce à un logiciel et au câble série RS 232/422.
- Les instructions du programme sont exécutées. Le CPU et la DSPGATE s'échangent les signaux de commande et de mesure de l'encodeur et des indicateurs à travers l'UBUS.
- Au même moment, la DSPGATE envoie les signaux de commande analogiques en courant (après CNA) à l'ampli et compte les impulsions venant de l'encodeur (conditionnement) tout en mettant l'état des indicateurs dans des registres bien spécifiques.
- L'amplificateur génère un courant proportionnel au courant de commande livré par la DSPGATE en appliquant une tension par MLI sur les bornes du moteur.
- Si les signaux D'E/S digitaux sont utilisés pour une application quelconque. L'IOGATE les conditionne et les sauvegarde dans un registre pour que le CPU puisse les utiliser.

IV.3.1 ASPECT LOGICIEL DE L'INTERFACE

L'écriture de programme de mouvement nécessite l'étude logicielle préalable de l'interface. Celle-ci requiert la connaissance d'un certain nombre d'éléments dont :

- **L'exécutible principal Pwin32pro** : C'est un pilote développé par DELTA TAU qui fournit un environnement permettant la configuration et la programmation de l'interface Turbo UMAC, l'accès à toutes ses applications, ainsi que la création des programmes de mouvement.

- **Les variables de travail** : Afin de pouvoir travailler dans un environnement adéquat et sûr, des variables ont été mises au point. Elles se présentent en quatre classes :

- *Variables I* : Elles permettent d'accéder au registre de contrôle et de configurer l'interface pour les applications désirées.
- *Variables P* (globales) : Elles sont utilisées pour le traitement de données.
- *Variables Q* (locales) : Elles servent d'arguments pour les fonctions.
- *Variables M* : Elles permettent de pointer des adresses mémoire.

- **La définition d'axe** : Un axe est défini en l'affectant à un moteur avec un facteur de *graduation* et un *offset*.

Du point de vue logiciel, un axe se réfère à un moteur par l'une des lettres alphabétiques : X, Y, Z, A, B, C, U, V et W. Une affectation d'une valeur à cette lettre correspond à un déplacement du moteur associé. Les caractéristiques d'un axe sont :

- *La graduation* : C'est la définition d'une nouvelle unité d'utilisation, qui sera la nouvelle résolution de déplacement du moteur.
- *L'offset* : Ce sont les quelques unités d'utilisateur correspondant à la position zéro du moteur.
- *Le type* : Trois types d'axe existent ; les axes cartésiens (X, Y, Z, U, V et W), les axes de rotation (A, B et C) et les axes fantômes qui sont ceux auxquels aucun moteur n'est attaché.

- **Les systèmes en coordination** : L'exécution de programme de mouvement nécessite la définition des systèmes en coordination (même avec un seul moteur).

Un système en coordination est un groupement de moteurs effectué afin d'avoir une même synchronisation du mouvement de ceux-ci. Il est établi d'abord en affectant des axes aux moteurs. Aussi, au moins un moteur doit être attaché à un axe.

La définition syntaxique du $n^{ième}$ système en coordination s'effectue par : $\&n$ suivi des affectations d'axes aux moteurs faisant partie du système, sur une même ligne du programme.

- **Les principaux profils de vitesse** : Il existe plusieurs types de mouvements, chacun présentant un certain profil de vitesse, on en cite :

- *Le mouvement linéaire* : C'est un mouvement ayant un profil de vitesse linéaire.

- *Le mouvement spline* : Son principe est de raccorder une série de mouvements d'une manière cubique, sans discontinuité de la vitesse ou de l'accélération.

- *Le mouvement PVT* (Position – Velocity – Time) : On utilise ce type de mouvement pour un contrôle plus précis de la forme de la trajectoire.

- *Le mouvement circulaire* : Il est effectué en segmentant des arcs. Ce mouvement peut être utilisé en interpolation, et à l'aide des axes fantômes, pour générer des trajectoires sinusoïdales.

IV.4 IMPLEMENTATION DES PROGRAMMES DE MOUVEMENT

Un programme de mouvement est un ensemble d'informations et de lignes de commande que le processeur de l'UMAC utilise afin de planifier et d'enclencher un mouvement.

IV.4.1 ELABORATION D'UN PROGRAMME DE MOUVEMENT

Les commandes usuelles pour l'élaboration d'un programme de mouvement sont :

OPEN PROG {n° du programme} : ouvre un programme dans un buffer sur l'UMAC.

CLEAR : Efface le contenu du programme ouvert.

LINEAR, RAPID, CIRCLE, PVT, et SPLINE : Servent à définir les trajectoires (profil de vitesse).

INC et ***ABS*** : Définissent le mode de mouvement (incrémentale ou absolu).

CLOSE : ferme le buffer ouvert afin d'exclure toutes commandes en ligne (écrites dans le Terminale Windows).

Lettre d'axe {α} : la commande de mouvement (***X60*** par exemple) s'effectue en spécifiant un axe suivi d'un déplacement.

De plus, nous introduisons les notions suivantes dont la connaissance est indispensable pour l'élaboration des programmes de mouvements :

- **Les instructions usuelles** : telles que les boucles *WHILE*, *IF...ELSE*.
- **Les paramètres temporels** : qui sont des paramètres dont la déclaration dans le programme est recommandée (autrement, des valeurs par défaut leur seront attribuées). Ces paramètres sont *TA* (le temps d'accélération), *TS* (le temps de courbure pour le mode S-Curve), *TM* (le temps de mouvement) et la vitesse *F* (Feedrate).
- **Les sous-programmes** : Ils sont utilisés afin de procéder à une programmation bien structurée. Pour ce faire, des commandes telles que : *GOSUBx* et *CALLx* ont été prévues. Ces dernières permettent de faire des sauts à l'étiquette ou au programme *x* respectivement. Le retour à la ligne suivante du programme principal se fait à la rencontre de la commande *RETURN*.

IV.4.2 EXECUTION D'UN PROGRAMME DE MOUVEMENT

Après avoir choisi le système en coordination pour l'exécution d'un programme, le programme à exécuter est sélectionné avec la commande *B* suivie de son numéro. Enfin, la commande *R* sert à lancer son exécution.

Avant l'exécution du programme, tous les systèmes en coordination doivent être mis en boucle fermée, en tapant < *CTRL+A* >.

On peut également, en cas de doute, contrôler chaque moteur séparément à titre d'essais, à l'aide de la commande *JOG*.

En cas d'anomalies de fonctionnement, des commandes sont mises à la disposition de l'utilisateur, et qui sont :

A < *RC* > : Arrête le programme de mouvement.

Q < *RC* > : Stoppe le programme à la fin de l'exécution du mouvement en cours.

K < *RC* > : Neutralise les amplificateurs qui alimentent les moteurs. Ces derniers poursuivront des mouvements dus à leurs inerties respectives.

IV.4.3 EXEMPLE DE PROGRAMME

Voici un exemple commenté de programme de mouvement pour une tâche de type Pick and place :

CLOSE ; ferme tous les buffers ouvert afin d'exclure toutes commandes en ligne

DELETE GATHER ; efface toutes les données intermédiaires

INDEFINE ALL ; efface les définitions liées à tous les systèmes en coordination

&7 #1->363.889X #2->363.889Y #3->363.889Z ; lie le moteur 1 à l'axe X, 2 à Y et 3 à Z et définit la graduation d'axe à 1 degré

OPEN PROG 25001 CLEAR ; ouvre le buffer pour écrire le programme 25001

LINEAR ; le profil de vitesse est linéaire (interpolation linéaire)

INC ; mode incrémental

TA 1000 ; le temps d'accélération est de 1000msec

TM 2000 ; le temps de mouvement est de 2000msec

TS 300 ; le temps de courbure est de 300msec

X60 Y30 Z1800 ; 60 unités de mouvement (degrés dans notre cas) pour X, 30 pour Y et 1800 pour Z (correspondant à cinq tours du troisième moteur et donc 10cm en translation)

X-60 Y-30 Z-1800 ; mêmes unités de mouvement dans le chemin inverse

CLOSE ; ferme le programme 25001

IV.5 CONCLUSION

Deux approches de la planification de tâches ont été traitées au cours de ce chapitre. La première utilise la génération de mouvement entre deux points avec trajectoire quelconque entre les deux. Nous avons pu planifier la tâche Pick and place pour son illustration.

La seconde approche explorée est celle de la génération de mouvement entre deux points avec trajectoire imposée entre les deux (dans notre cas il s'agit de droites). Pour cela, nous avons pu simuler des tâches d'écriture avec succès.

Nous avons ensuite fait une étude matérielle et logicielle de l'interface UMAC utilisée pour la commande du robot et expliqué la démarche à adopter pour l'élaboration des programmes de mouvement.

CONCLUSION GENERALE

CONCLUSION GENERALE

Dans ce projet, nous avons traité le problème de conception, de réalisation et de commande d'un robot SCARA.

Le premier problème auquel nous nous sommes heurtés a été l'absence d'une méthodologie claire de conception et de réalisation de robots manipulateurs. Notre démarche a été de définir une feuille de route dans laquelle nous avons tracé les grandes lignes de notre travail. Le peu de documentation détaillée sur la conception de robots -les fabricants étant peu communicatifs sur la technologie mécanique employée- a pu être pallié par le travail en équipe que nous avons mené, ce genre de projet étant pluridisciplinaire par nature.

Le choix de la morphologie SCARA constitue une solution optimale dans notre cas compte tenu de l'absence de freins. Le maintien d'une configuration donnée est assuré.

Les solutions mécaniques adoptées, en accord avec des ingénieurs mécaniciens sont simples - Les plans du robot donnés en annexe A.1 l'illustrent très bien. Cette simplicité est recherchée car elle permet de faciliter la réalisation.

Nous avons ensuite entamé la réalisation du robot au niveau de la SNVI où les moyens humains et matériels adéquats ont été mis à notre disposition. Ce stage nous a permis

d'améliorer la rigidité, la précision et la fiabilité du robot et ce, en tenant compte des propositions des professionnels. On a pu choisir des solutions mieux adaptées et moins contraignantes. Par exemple, le contrepoids envisagé au départ pour l'équilibre du robot a été écarté au profit des billes de maintien car celles-ci jouent le même rôle sans introduire pour autant des modifications importantes.

Les frottements de la structure finale ont été sensiblement réduits et ce, par l'utilisation d'une solution mécanique simple consistant à utiliser des bagues et des rondelles graissées. L'emploi de roulements- nécessitant un usinage de précision- a donc été volontairement écarté.

Le robot une fois réalisé, nous avons calculé ses différents modèles géométriques, cinématiques et dynamique, mais aussi ceux des moteurs. Pour le modèle dynamique, une modélisation analytique par l'approche d'Euler-Lagrange a été effectuée. Le modèle des connaissances élaboré a pour avantage de comporter des termes ayant une signification physique. Aussi, et contrairement au modèle de représentation calculé par identification et qui est une solution très utilisée aussi, notre modèle ne dépend pas d'une plage d'entrées sorties particulière mais est beaucoup plus général. Cependant, le modèle identifié est généralement beaucoup plus précis car déduit d'expérimentations réelles. Il ne fait pas appel aux hypothèses simplificatrices. L'identification du modèle serait très intéressante à l'avenir. On pourra ainsi le comparer au modèle que nous avons utilisé.

A partir des modèles développés au chapitre 2, nous avons pu synthétiser différentes lois de commande au cours du chapitre 3. La stratégie adoptée se base sur trois techniques de commandes différentes.

Les deux premières sont la commande linéarisante par géométrie différentielle et la commande par couple calculé. Ces deux techniques utilisent les équations du MD. Le système commandé est donc de type MIMO. Elles se basent sur une linéarisation du système puis une commande dans l'espace opérationnel pour la première et articulaire pour la seconde. Les résultats de poursuite simulés sont très satisfaisants tant du point de vue précision que celui des énergies et des profils des signaux de commandes.

L'avantage de la géométrie différentielle est que la synthèse de la commande est faite directement dans l'espace opérationnel, ce qui est plus naturel surtout pour la planification de tâches. Son inconvénient est qu'elle fait appel au MGI contrairement au couple calculé qui nécessite un temps de calcul plus réduit. Les résultats étant identiques, le choix d'une technique de commande au profit d'une autre doit faire l'objet d'un compromis et dépend surtout de l'application.

La seconde approche est beaucoup plus simple. C'est une commande de type décentralisé où l'on commande trois systèmes linéaires de type SISO. L'intérêt des simulations que nous avons faites est qu'elles s'approchent du modèle réel car la structure du correcteur PID utilisé est identique à celle implémentée dans le contrôleur du robot. De même que les signaux de perturbation qui utilisent les équations du modèle dynamique. Les résultats obtenus ne sont pas aussi bons que ceux des techniques précédentes. Cependant, cette technique est très utilisée car la loi de commande est très simple et l'optimisation des gains du régulateur conduit à des résultats satisfaisants.

Diverses tâches ont été planifiées et simulées, et des programmes de mouvement ont été implémentés sur le contrôleur permettant ainsi de commander le robot en boucle fermée. Ceci illustre bien les capacités de reprogrammation des robots qui s'adaptent très vite aux changements de tâches ce qui constitue leur force majeure.

Comme perspectives, nous proposons :

- La réalisation d'un poignet rotule pour obtenir une structure à six degrés de liberté et pouvoir ainsi contrôler en plus de la position de la pointe de l'outil, son orientation.
- L'amélioration de la structure mécanique existante pour augmenter sa rigidité et sa fiabilité.
- La réalisation d'autres types de structures de robots (anthropomorphes, mobiles...etc.) et ce en tirant parti du partenariat avec l'industrie et la SNVI en particulier.
- Concevoir un contrôleur spécifique autre que l'interface UMAC ou encore la mise en place d'une interface virtuelle permettant de commander le robot et de connaître sa position en temps réel.

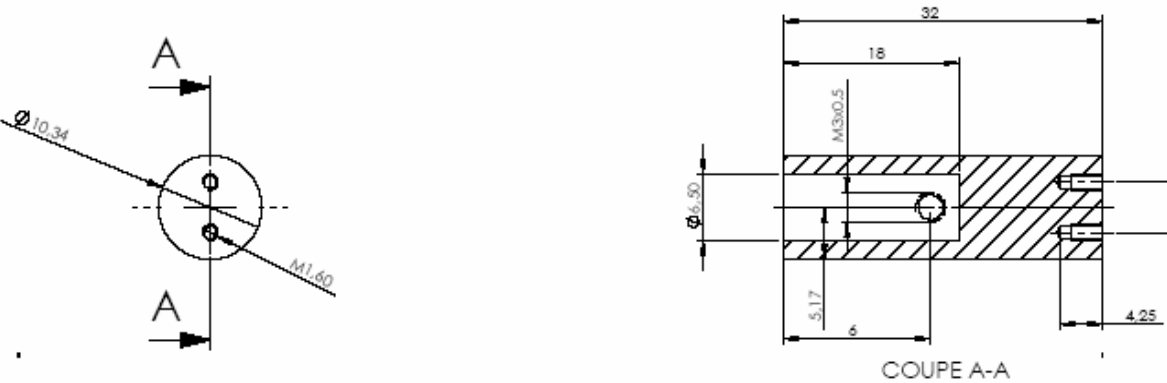
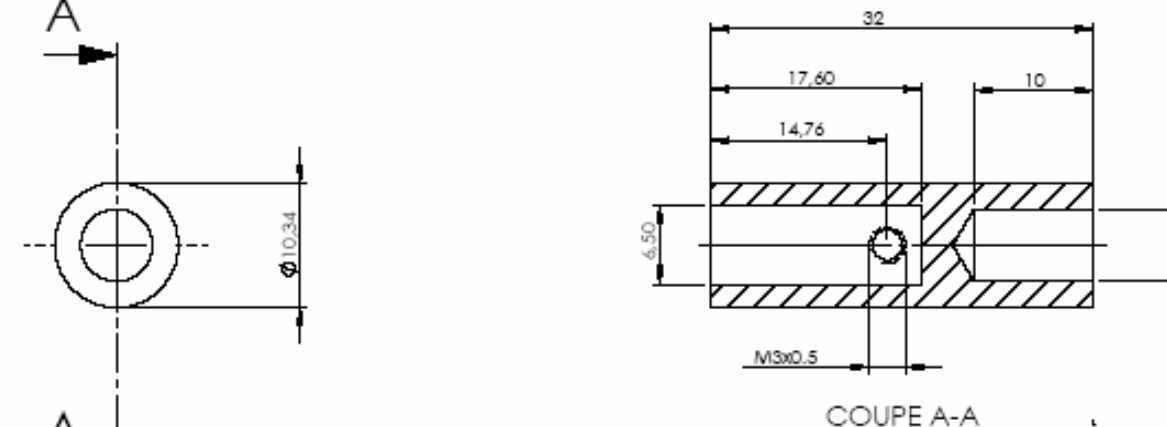
- D'autres types d'interfaces- qui seraient spécifiques aux tâches planifiées- peuvent être envisagées. Par exemple, nous pouvons très bien imaginer une interface de communication entre un clavier et le robot lui-même pour le transformer ainsi en robot écrivain.
- L'ajout de capteurs plus perfectionnés comme une caméra par exemple peut être envisagé pour un asservissement visuel du robot.

ANNEXES

ANNEXES

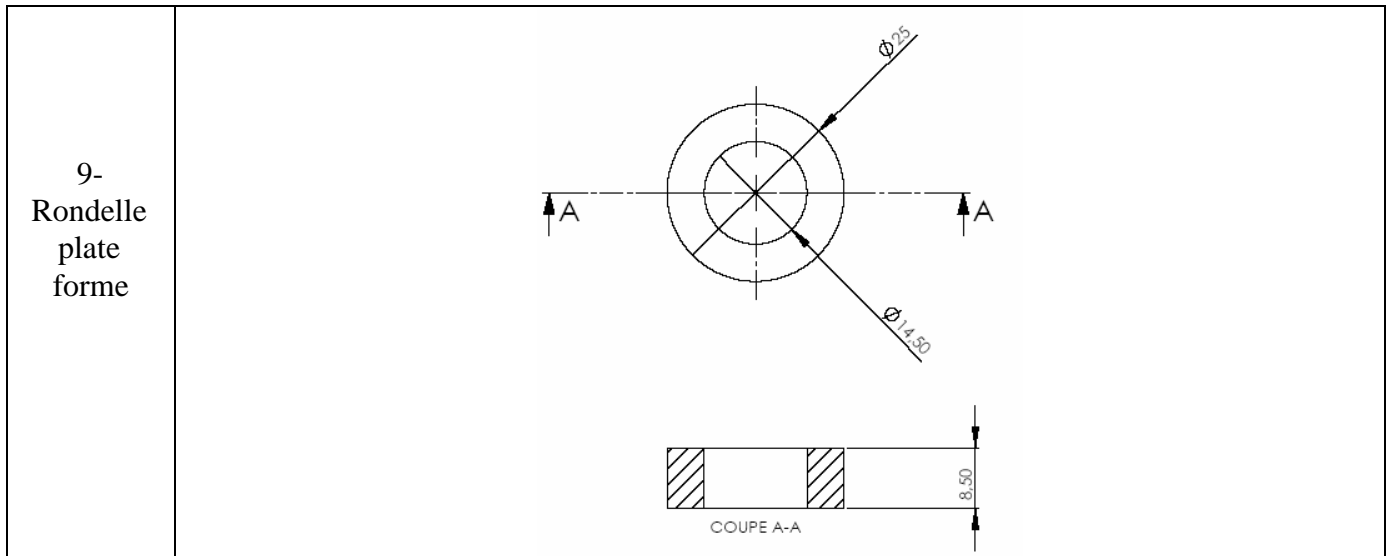
ANNEXES A

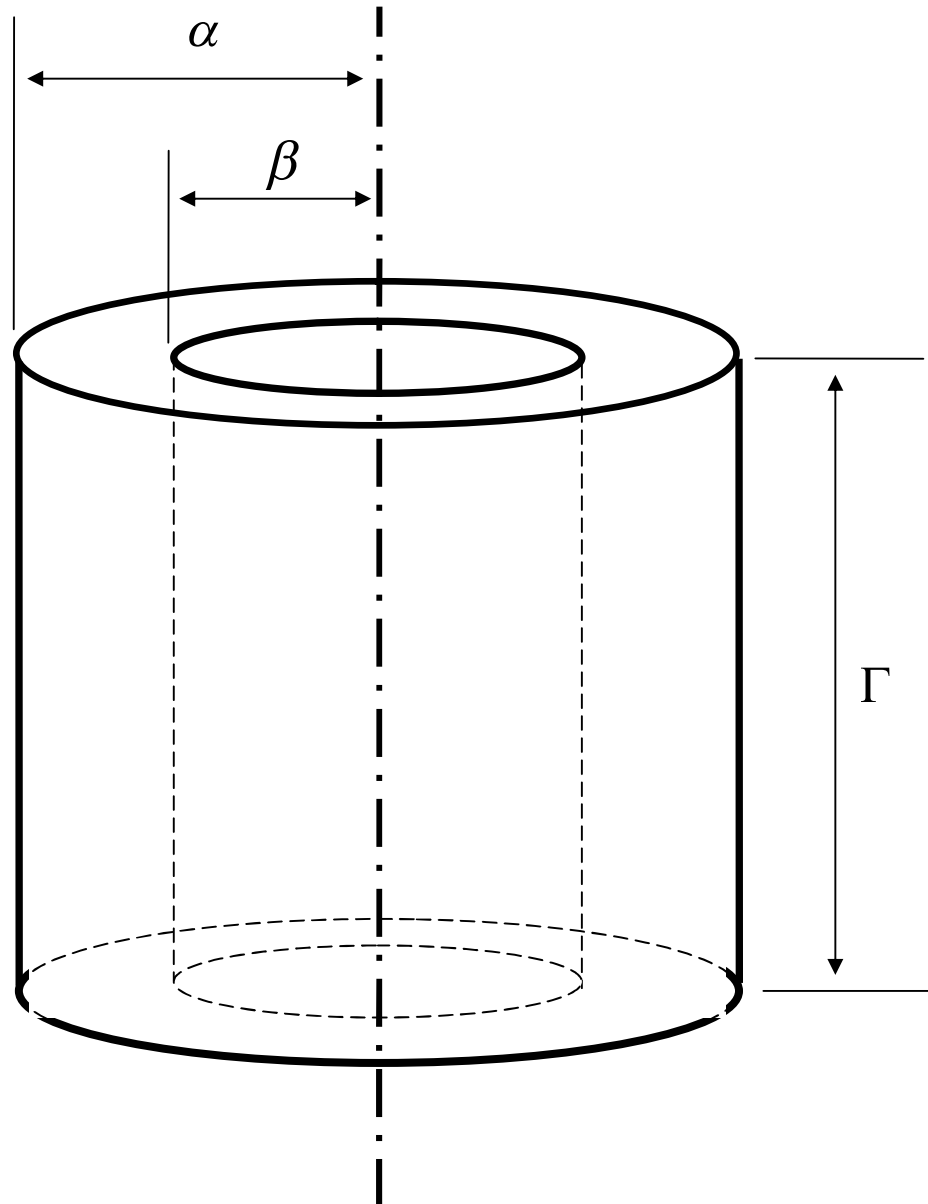
Annexe A.1 Plans de coupe.

N° et nom de la pièce	Plans de la pièce
1- Arbre	 <p>The drawing for '1- Arbre' consists of two views. The front view on the left shows a circular shaft with a diameter of $\phi 10,34$ and a thread of $M1,60$. Section lines 'A-A' are shown with arrows pointing to the right. The cross-section view on the right, labeled 'COUPE A-A', shows a shaft with a total length of 32. It features a central hole with a diameter of $\phi 6,50$ and a depth of 6. A thread of $M3 \times 0,5$ is located at a distance of 18 from the left end. The shaft has a diameter of 6 at the left end and 5 at the right end. A chamfered end is shown with a chamfer angle of $5,17$ and a chamfer width of 4,25.</p>
2- Arbre outil	 <p>The drawing for '2- Arbre outil' consists of two views. The front view on the left shows a hollow shaft with an outer diameter of $\phi 10,34$. Section lines 'A-A' are shown with arrows pointing to the right. The cross-section view on the right, labeled 'COUPE A-A', shows a shaft with a total length of 32. It has a central hole with a diameter of $\phi 6,50$ and a depth of 6. A thread of $M3 \times 0,5$ is located at a distance of 17,60 from the left end. The shaft has a diameter of 6 at the left end and 5 at the right end. A chamfered end is shown with a chamfer angle of $14,76$ and a chamfer width of 10.</p>

<p>3-Bague</p>	<p>Technical drawing of a ring (Bague). The front view shows an outer diameter of $\phi 34,50$ and an inner diameter of $\phi 10,50$. The cross-section (COUPE A-A) shows a thickness of 12 and a central hole diameter of $\phi 14,50$.</p>
<p>4 - Plat</p>	<p>Technical drawing of a plate (Plat). The front view shows a circular plate with a diameter of $\phi 35$, four holes of diameter $\phi 2$, and a central hole of diameter $\phi 3$. The cross-section (COUPE A-A) shows a thickness of 5.</p>
<p>5- Bras outil</p>	<p>Technical drawing of a tool arm (Bras outil). The front view shows a total length of 250, a diameter of 38.10, and various hole positions. The side view (COUPE A-A) shows a diameter of $\phi 12$ and a hole of diameter $\phi 14,50$. The cross-section (COUPE B-B) shows a diameter of $\phi 25,5$ and a hole of diameter $\phi 5,11$.</p>

<p>6- Bras</p>	<p>Usiner le trou tarodé sur un seul bras (Pour la fixation du guide de la tige filetée)</p>
<p>7-Plate forme</p>	
<p>8-rondelle</p>	



Annexe A. 2 Espace de travail du robot SCARA.**Figure A.2.1** *Espace de travail du robot SCARA*

Avec :

$$\Gamma = l_3$$

$$\alpha < a_1 + a_2$$

$$\beta > 0.17 \text{ m}$$

ANNEXES B

Annexe B. 1 Algorithme DH.

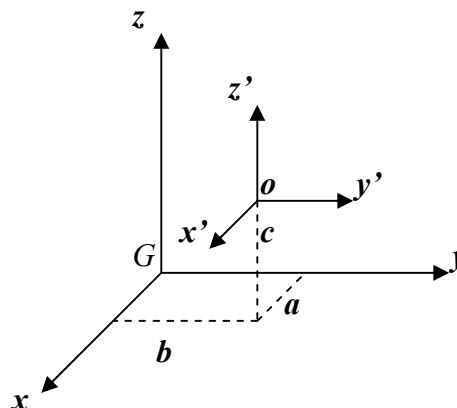
0. Numéroté les articulations de 1 à n en commençant par la base jusqu'à l'outil (les angles de lacet, tangage et roulis devant être numérotés dans cet ordre)
1. Attribuer à la base du robot un repère orthonormé noté L_0 , tout en s'assurant que z_0 est aligné avec l'axe de l'articulation 1. Poser $k = 1$.
2. Aligner z_k avec l'axe de l'articulation $k + 1$.
3. Placer l'origine de chaque repère L_k au point d'intersection des axes z_k et z_{k-1} . S'ils ne se coupent pas, alors utiliser l'intersection de z_k et une normale commune à z_k et z_{k-1} .
4. Choisir x_k de sorte à ce qu'il soit orthogonal à z_k et z_{k-1} . Si z_k et z_{k-1} sont parallèles, diriger x_k loin de z_{k-1} .
5. Choisir y_k de manière à obtenir un repère orthonormé L_k respectant la règle de la main droite.
6. Poser $k = k + 1$. Si $k < n$, aller vers l'étape 2. ; sinon continuer.
7. Placer l'origine du repère L_n à l'extrémité de l'outil. Aligner z_n avec le vecteur d'attaque « The approach vector », y_n avec le vecteur de glissement « The sliding vector » et x_n avec le vecteur normal « The normal vector » de l'outil. Poser $k = 1$.
8. Placer b_k à l'intersection des axes x_k et z_{k-1} . S'ils ne se coupent pas, alors utiliser l'intersection de x_k avec une normale commune à x_k et z_{k-1} .
9. Calculer θ_k : angle de rotation de x_{k-1} à x_k et mesuré autour de z_{k-1} .
10. Calculer d_k : distance entre l'origine du repère L_{k-1} et le point b_k et mesurée le long de l'axe z_{k-1} .
11. Calculer a_k : distance entre le point b_k et l'origine du repère L_k et mesurée le long de x_k .

12. Calculer α_k : angle de rotation de z_{k-1} à z_k et mesuré autour de x_k .

13. Poser $k = k + 1$. Si $k < n$, aller vers l'étape 8. ; sinon arrêter.

Annexe B. 2 Théorème de Huygens généralisé.

Soit la figure B. 2.1 suivante :



Avec :

Figure B.2.1 Centre de Masse du corps étudié

G : le centre de masse du corps étudié

Notons par : $I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{xz}, I_{yz}$ les moments et produits d'inertie par rapport au centre de masse et exprimés dans le repère R_0 , et $I_{xx}^*, I_{yy}^*, I_{zz}^*, I_{xy}^*, I_{xz}^*, I_{yz}^*$ ces moments et produits d'inertie exprimés dans un repère R_G dont l'origine est placée au centre de masse G du corps étudié.

Alors, le théorème de Huygens pour les moments d'inertie est donné par :

$$\begin{cases} I_{xx} = I_{xx}^* - (b^2 + c^2)m \\ I_{yy} = I_{yy}^* - (a^2 + c^2)m \\ I_{zz} = I_{zz}^* - (a^2 + b^2)m \end{cases} \quad (\text{B.2.1})$$

Et le théorème de Huygens pour les produits d'inertie est :

$$\begin{cases} I_{xy} = I_{xy}^* - abm \\ I_{xz} = I_{xz}^* - acm \\ I_{yz} = I_{yz}^* - bcm \end{cases} \quad (\text{B.2.2})$$

Telle que :

m : la masse du corps étudié

Si l'on note par J_G : le tenseur d'inertie par rapport au centre de masse et exprimé dans le repère R_G et tel que :

$$J_G = \begin{pmatrix} I_{xx}^* & -I_{xy}^* & -I_{xz}^* \\ -I_{xy}^* & I_{yy}^* & -I_{yz}^* \\ -I_{xz}^* & -I_{yz}^* & I_{zz}^* \end{pmatrix} \quad (\text{B.2.3})$$

Alors, le tenseur d'inertie J_0 par rapport au centre de masse et exprimé dans le repère R_0 est donné par :

$$J_0 = J_G - m \begin{pmatrix} (b^2 + c^2) & -ab & -ac \\ -ab & (a^2 + c^2) & -bc \\ -ac & -bc & (a^2 + b^2) \end{pmatrix} \quad (\text{B.2.4})$$

Annexe B. 3 Calcul de la matrice Jacobienne notée $J(q)$. (THE MANIPULATOR JACOBIAN)

Une variation infinitésimale dans l'espace articulaire notée dq engendre une variation infinitésimale dx dans l'espace opérationnel.

Cette variation dx est une combinaison d'une variation infinitésimale de la position dp et de l'orientation $d\phi$:

$$dx = \begin{bmatrix} dp \\ d\phi \end{bmatrix} \quad (\text{B.3.1})$$

Ceci est illustré dans la figure B.3.1 :

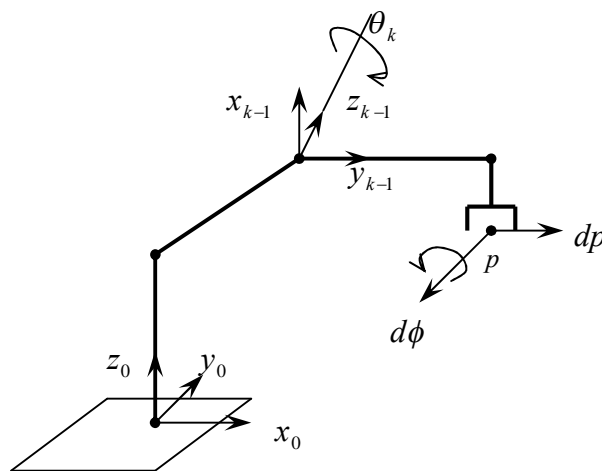


Figure B.3.1. *Rotation et translation infinitésimales*

dx et dq sont liés par la matrice jacobienne J de dimension $6 \times n$ comme spécifié par la relation (B.3.2) et la figure (B.3.2) :

$$dx = J dq \quad (\text{B.3.2})$$

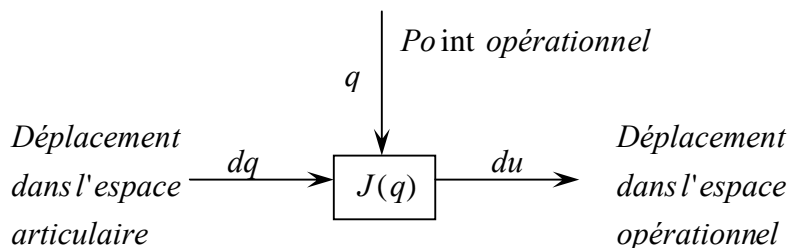


Figure B.3.2. Matrice Jacobienne

Cette matrice peut être décomposée en deux sous matrices de dimension $3 \times n$, $A(q)$ et $B(q)$ telles que :

$$J(q) = \begin{bmatrix} A(q) \\ \text{-----} \\ B(q) \end{bmatrix} \quad (\text{B.3.3})$$

$A(q)$: partie de la matrice jacobienne associée aux variations linéaires de l'outil.

$B(q)$: partie de la matrice jacobienne associée aux variations angulaires de l'outil.

Ceci est donné par les relations (B.3.4) et (B.3.5) :

$$dp = A(q) dq \quad (\text{B.3.4})$$

$$d\phi = B(q) dq \quad (\text{B.3.5})$$

En divisant les deux membres de la relation (B.3.2) par une variation infinitésimale du temps dt , nous obtenons la relation (B.3.6) représentant le MCD :

$$\dot{x} = J \dot{q} \quad (\text{B.3.6})$$

Tel que $\dot{x} = (\dot{P}_x \ \dot{P}_y \ \dot{P}_z \ \dot{\phi}_x \ \dot{\phi}_y \ \dot{\phi}_z)^T$ est le vecteur des vitesses opérationnelles.

Ainsi les relations (B.3.4) et (B.3.5) deviennent :

$$\dot{p} = A(q) \dot{q} \quad (\text{B.3.7})$$

$$\dot{\phi} = B(q)\dot{q} \quad (\text{B.3.8})$$

Etape 1 : Calcul de la matrice $A(q)$ liant la vitesse de translation de l'outil $\dot{P}(q)$ et le vecteur des vitesses articulaires \dot{q} : (Relation B.3.7)

Tels que :

$$\dot{P}(q) = (\dot{P}_x(q) \ \dot{P}_y(q) \ \dot{P}_z(q))^T \quad \text{et} \quad \dot{q} = [\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3]^T$$

Le calcul de $A(q)$ se fait en dérivant le vecteur position de la matrice de transformation homogène représentant le MGD, et ce par rapport aux variables articulaires θ_1 , θ_2 et d pour former les colonnes de la matrice, comme indiqué par la relation (B.3.9) :

$$A(q) = \begin{pmatrix} \frac{\partial Px}{\partial \theta_1} & \frac{\partial Px}{\partial \theta_2} & \frac{\partial Px}{\partial d} \\ \frac{\partial Py}{\partial \theta_1} & \frac{\partial Py}{\partial \theta_2} & \frac{\partial Py}{\partial d} \\ \frac{\partial Pz}{\partial \theta_1} & \frac{\partial Pz}{\partial \theta_2} & \frac{\partial Pz}{\partial d} \end{pmatrix} \quad (\text{B.3.9})$$

Nous obtenons donc :

$$A(q) = \begin{pmatrix} -a_1 S_1 - a_2 S_{1-2} & -a_2 S_{1-2} & 0 \\ a_1 C_1 + a_2 C_{1-2} & -a_2 C_{1-2} & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (\text{B.3.10})$$

Etape 2 : Détermination de la matrice $B(q)$, partie de la matrice jacobienne liant les vitesses

angulaires $[\dot{\phi}_1, \dot{\phi}_2, \dot{\phi}_3]^T$ au vecteur des vitesses articulaires \dot{q} : (Relation B.3.8)

De la relation (B.3.5), on a :

$$d\phi = \sum_{k=1}^n (dq_k) b^k(q) \quad (\text{B.3.11})$$

Tels que : $b^k(q)$ est la $k^{\text{ème}}$ colonne de $B(q)$. Elle est constituée de gains de proportionnalité liant les déplacements de l'articulation k aux déplacements de l'outil quand toutes les autres articulations sont immobiles.

Et afin de déterminer les $b^k(q)$, nous nous intéressons à deux cas particuliers. Le premier cas est celui pour lequel l'articulation k est prismatique. De la figure (B.3.1), nous

déduisons que le corps k ainsi que tous les corps subséquents vont avoir des mouvements de translation le long de z_{k-1} . Par conséquent, à chaque mouvement de translation le long de z_{k-1} n'est associé aucune variation angulaire de l'outil. Ainsi $d\phi = 0$ ce qui veut dire que $b^k(q) = 0$. Dans le cas où l'articulation k est rotoïde. La rotation autour de l'axe de cette articulation entraînera avec elle la rotation autour de z_{k-1} de tous les corps subséquents y compris l'outil. Dans ce cas, le déplacement angulaire de l'outil induit par le mouvement autour de l'axe z_{k-1} est donné par la relation :

$$d\phi = (dq_k) z^{k-1} \quad (\text{B.3.12})$$

En comparant la relation (B.3.11) avec la relation (B.3.12), nous arrivons au résultat suivant : $b^k(q) = z^{k-1}(q)$ et ce quand l'articulation k est rotoïde.

D'une manière générale, quelle que soit le type de l'articulation mise en jeu, nous avons la relation suivante :

$$b^k(q) = \xi_k z^{k-1}(q) \quad 1 \leq k \leq n \quad (\text{B.3.13})$$

Avec :

$$\begin{cases} \xi_k = 0 & \text{si } k \text{ est prismatique.} \\ \xi_k = 1 & \text{si } k \text{ est rotoïde.} \end{cases}$$

Reste à déterminer uniquement l'expression de $z^{k-1}(q)$ qui est égale à :

$$z^{k-1}(q) = R_0^{k-1} z^3 \quad (\text{B.3.14})$$

Avec :

$$z^3 = [0 \quad 0 \quad 1]^T$$

D'où, l'expression de $B(q)$ qui est donnée par :

$$B(q) = [\xi_1 z^0(q) \quad \xi_2 z^1(q) \quad \dots \quad \xi_n z^{n-1}(q)] \quad (\text{B.3.15})$$

Dans notre cas ;

$$B(q) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & -1 & 0 \end{bmatrix} \quad (\text{B.3.16})$$

Finalement, nous obtenons l'expression générale de la matrice jacobienne de dimension $6 \times n$:

$$J(q) = \begin{bmatrix} \frac{\partial P(q)}{\partial q_1} & \frac{\partial P(q)}{\partial q_2} & \cdots & \frac{\partial P(q)}{\partial q_n} \\ \text{-----} \\ \xi_1 z^0(q) & \xi_2 z^1(q) & \cdots & \xi_n z^{n-1}(q) \end{bmatrix} = \begin{bmatrix} A(q) \\ \text{-----} \\ B(q) \end{bmatrix} \quad (\text{B.3.17})$$

Et à partir des équations (B.3.10) et (B.3.16), et en utilisant la relation (B.3.17), nous aboutissons à la matrice Jacobienne $J(q)$ de notre robot donnée par la relation (B.3.18) :

$$J(q) = \begin{pmatrix} -a_1 S_1 - a_2 S_{1-2} & -a_2 S_{1-2} & 0 \\ a_1 C_1 + a_2 C_{1-2} & -a_2 C_{1-2} & 0 \\ 0 & 0 & -1 \\ \text{-----} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & -1 & 0 \end{pmatrix} \quad (\text{B.3.18})$$

Annexe B. 4 Algorithme d'Euler Lagrange.

Nous définissons les deux grandeurs suivantes qui sont utilisées dans le développement des calculs de l'algorithme :

$$C_{kj}^i(q) \triangleq \frac{\partial}{\partial q_k} D_{ij}(q) - \frac{1}{2} \frac{\partial}{\partial q_i} D_{kj}(q) \quad 1 \leq i, j, k \leq n \quad (\text{B.5.1})$$

$$h_i(q) \triangleq - \sum_{k=1}^3 \sum_{j=i}^n g_k m_j A_{ki}^j(q) \quad 1 \leq i \leq n \quad (\text{B.5.2})$$

$$J^k(q) = \begin{bmatrix} \frac{\partial \bar{c}^k}{\partial q_1} & \dots & \frac{\partial \bar{c}^k}{\partial q_k} & 0 \\ \xi_1 z^0 & \dots & \xi_k z^{k-1} & 0 \end{bmatrix} \triangleq \begin{bmatrix} A^k(q) \\ B^k(q) \end{bmatrix} \quad (\text{B.5.3})$$

Nous définissons aussi par souci de rigueur l'équation caractérisant les frottements et que nous avons négligé dans nos calculs :

$$b_k(\dot{q}) = b_k^v \dot{q}_k + \text{sign}(\dot{q}_k) \left[b_k^d + (b_k^s - b_k^d) \exp \frac{-|\dot{q}_k|}{\varepsilon} \right] \quad 1 \leq k \leq n \quad (\text{B.5.4})$$

Avec :

b_k^v , b_k^d et b_k^s : représentent les coefficients de frottement visqueux, dynamique et statique respectivement.

Et :

ε : un paramètre positif infinitésimal.

1. Définir les différents repères $\{L_0, L_1, \dots, L_n\}$ du robot en utilisant l'algorithme DH.

2. Poser $T_0^0 = I, i = 1, D(q) = 0$.
3. Déterminer Δc^i , les coordonnées homogènes du centre de masse du corps i dans le repère L_i .
4. Noter par L_c le repère résultant de la translation du repère L_i vers le centre de masse du corps i . Le calcul de \bar{D}_i , le tenseur d'inertie du corps i par rapport à son centre de masse exprimé dans le repère L_c , s'effectue à l'aide de l'équation (II.4.3)

5. Calculer :

$$\begin{aligned} z^{i-1}(q) &= R_0^{i-1}(q) i^3 \\ T_0^i(q) &= T_0^{i-1}(q) T_{i-1}^i(q) \\ \bar{c}^i(q) &= H_1 T_0^i(q) \Delta c^i \\ D_i(q) &= R_0^i(q) \bar{D}_i [R_0^i(q)]^T \end{aligned}$$

6. Calculer $J^i(q)$ en utilisant l'équation (B.5.3).
7. Diviser $J^i(q)$ en s'imprégnant de la notation utilisée dans l'équation (B.5.3) et calculer:

$$D(q) = D(q) + [A^k(q)]^T m_k A^k(q) + [B^k(q)]^T D_k B^k(q)$$

8. Poser $i = i + 1$. Si $i \leq n$, aller à l'étape 3 ; sinon, poser $i = 1$ et continuer.
9. Calculer $C^i(q), h_i(q)$ et $b_i(\dot{q})$ en utilisant les équations (B.5.1), (B.5.2) et (B.5.4), respectivement.
10. Formuler la $i^{\text{ème}}$ équation sous la forme :

$$\sum_{j=1}^n D_{ij}(q) \ddot{q}_j + \sum_{k=1}^n \sum_{j=1}^n C_{kj}^i(q) \dot{q}_k \dot{q}_j + h_i(q) + b_i(\dot{q}) = \tau_i$$

11. Poser $i = i + 1$. Si $i \leq n$, revenir à l'étape 9.

BIBLIOGRAPHIE

BIBLIOGRAPHIE CONSULTEE

- [1] : L. Badji et F. Bouziani, “ Etude et mise en fonction de l’interface d’axes industriels Turbo Umac”, *Projet de Fin d’Etudes, ENP, Alger, 2002.*
- [2] : F. Boudjema, “Modélisation Analytique Et Simulation Des Processus Industriels” *Notes de Cours de 5^{ème} Année Automatique de l’Ecole Nationale Polytechnique, Année universitaire 2005-2006.*
- [3] : A. CHEVALIER et J. BOHAN, “ Guide du technicien en productique ”, *Pour maîtriser la production industrielle, Editions Hachette Technique, 1995.*
- [4] : B. Gorla et M. Renaud, “Modèles des Robots Manipulateurs”, *Application à leur commande, Cepadues Editions, 1984.*
- [5] : C. Hazard, “Mémotech dessin industriel”, *Editions Casteilla, 1984.*
- [6] : W. Khalil et E. Dombre, “Modélisation, identification et commande des robots”, *2^{ème} édition revue et augmentée, Hermès Sciences, 1999.*
- [7] : D. Leroux, “ Les robots, stratégie industrielle”, *Hermès Editions, 1986.*
- [8] : P. Marsh, “ Le temps des robots, de la fiction aux industries de pointe ”, *Paris, Bordas, 1986.*
- [9] : M. Priel, “ Les robots industriels ”, *Caractéristiques, performances et choix, Afnor Technique, 1990.*

- [10]: R.J. Schilling, “Fundamentals of robotics: Analysis and Control”, *Prentice Hall*, 1990.
- [11]: M. W. Spong et M. Vidyasagar, “Robot Dynamics And Control”, *John Wiley & Sons*, 1989.
- [12]: M. Tadjine, “Automatique Avancée” *Notes de Cours de 5^{ème} Année Automatique de l’Ecole Nationale Polytechnique*, Année universitaire 2005-2006.
- [13]: M. Tadjine, “Commande Des Systèmes Multivariable” *Notes de Cours de 4^{ème} Année Automatique de l’Ecole Nationale Polytechnique*, Année universitaire 2004-2005.
- [14]: http://www.linde-gas.fr/International/Web/LG/FR/likelgfr.nsf/DocByAlias/ind_TIG
- [15]: <http://www.snvi-dz.com/pageLibre00010085.html>
- [16]: http://eavr.u-strasbg.fr/~bernard/education/master_it/slides1_master_it.pdf

BIBLIOGRAPHIE GENERALE

- [17] : S. Abdelaziz et S. Issaoui, “Réalisation et Commande d’un bras manipulateur par l’Interface d’Axe Turbo Umac”, *Projet de Fin d’Etudes, ENP, Alger, 2005.*
- [18] : M.O Demaurex, “Approche théorique de la conception de la structure mécanique d’un robot industriel”, *Thèse présentée au Département de mécanique de l’Ecole Polytechnique Fédérale De Lausanne pour l’obtention du grade de Docteur es sciences techniques.*
- [19] : L. Marais, “Régulation Automatique”, *Systèmes Analogiques. Presses Polytechniques Romandes, 1987.*
- [20] : R. Quatremer et J. P. Trotignan, “La Construction Mécanique”, *I-Dessin, Conception, Normes. Edition de l’Usine Nouvelle, 1982.*
- [21] : G. Lacroux, “Les actionneurs électriques pour la robotique et les asservissements”, *technique et documentation, Paris, 1985.*

NOTES

Résumé :

Ce travail a pour but la conception, la réalisation et la commande d'un robot manipulateur à trois degrés de liberté. La structure conçue est de type SCARA. Elle a été réalisée au cours d'un stage pratique à la SNVI. Les modèles du robot ont été calculés analytiquement et différentes commandes ont été simulées. En dernier lieu, diverses tâches ont été planifiées, implémentées à l'aide du contrôleur UMAC puis exécutées par le robot.

Mots clés : Conception, réalisation, usinage, robot manipulateur, SCARA, modélisation, commande, planification de tâches.

ملخص:

إنّ الهدف من هذا العمل يكمن في تصميم و إنجاز ذراع آلية بغرض التحكم فيها. الذراع من نوع SCARA. تمّ إنجازها خلال تربيص بالشركة الوطنية للسيّارات الصناعيّة SNVI بالروبية. بعد ذلك عيّنت و حسبت مختلف الأمثلة الرياضيّة اللازمة لتحديد قوانين التحكم التي تمّ محاكاتها. أخيراً، خطّطت الذراع لإنجاز وظائف عدّة، ثمّ برمجت باستعمال جهاز التحكم UMAC.

الكلمات الأساسية: تصميم، إنجاز، تصنيع، ذراع آلية، SCARA، تمثيل، تحكم، تخطيط الوظائف.

Abstract :

Work's purpose is design, realization and control of a three degrees of freedom manipulator arm. The SCARA structure has been realized during a training course at the SNVI. Mathematical models have been elaborated and different types of commands were designed and simulated. Finally, some tasks were planed and implemented in UMAC controler and have been done by the robot.

Key words: Design, realization, machining, manipulator arm, SCARA, modeling, control, task planning.