

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira – Béjaïa



Faculté de Technologie



Département de Génie Electrique

Mémoire de fin d'étude

En vue de l'obtention d'un diplôme de Master en électrotechnique

Option : Automatismes industriels

Thème

Commande guidée par les données appliquée aux robots manipulateurs.

Réalisé par :

BENMEDDOUR Mohand ouali

BEDIAR Mohamed

Encadré par ;

Mr. MELAHI Ahmed

Remerciements

Avant, toute personne, nous tenons à remercier notre Dieu Tout Puissant pour nous avoir éclairci le chemin de ce travail.

Nous souhaitons tout d'abord remercier notre encadreur monsieur **MELAHI AHMED**, pour avoir accepté de diriger ce mémoire et de sa patience durant la période de l'encadrement, pour son suivi continué tout le long de la réalisation de ce mémoire et pour ses précieux conseils

Nos remerciements vont aussi à tous les enseignants d'Electrotechnique qui ont contribué à notre formation par ailleurs, précisément monsieur **TAIB Nabil**, Nos remerciements à tous les membres du jury qui ont accepté d'examiner notre travail.

Enfin, Nous adressons nos remerciements à toute notre famille sans oublier nos amis et collègues qui nous ont apporté leur soutien moral et intellectuel tout au long de notre démarche.

Table des matières

Table des matières.....	
Liste des figures.....	
Introduction générale.....	1

CHAPITRE I Modélisation des robots manipulateurs

I.1	Introduction	2
I.2	Historique	2
I.3	Définition d'un robot.....	2
I.4	Définitions des robots manipulateurs	2
I.5	Constituants mécaniques d'un robot	3
I.5.1	Organe terminal.....	3
I.5.2	Système mécanique articulé	3
I.5.2.1	Articulations.....	4
I.5.2.1.1	Articulation rotative	4
I.5.2.1.2	Articulation prismatique.....	4
I.5.3	Actionneurs	5
I.5.4	Capteurs.....	5
I.5.5	Système de commande et de traitement de l'information	5
I.6	Degré de liberté	6
I.7	Modélisation des robots manipulateurs.....	6
I.7.1	Transformation homogène	7
I.7.1.1	Transformation des repères	7
I.7.1.2	Matrice de rotation	8
I.7.2	Modélisation géométrique.....	8
I.7.2.1	Modèle géométrique direct	9

I.7.2.2	Modèle géométrique inverse	9
I.7.3	Modélisation cinématique	9
I.7.3.1	Modèle cinématique direct.....	9
I.7.3.2	Modèle cinématique inverse	10
I.7.4	Modélisation dynamique	10
I.7.4.1	Modèle dynamique inverse	10
I.7.4.2	Modèle dynamique direct	10
I.7.4.3	Formalisme d'Euler-Lagrange	11
I.8	Modélisation d'un robot manipulateur type SCARA à 2d.d.l.....	11
I.8.1	Modèle géométrique direct.....	12
I.8.2	Modèle géométrique inverse	12
I.8.3	Modèle cinématique direct	13
I.8.4	Modèle cinématique inverse.....	13
I.8.5	Modèle dynamique	13
I.9	Conclusion.....	15

CHAPITRE II Commande guidée par les données

II.1	Introduction	16
II.2	Contrôleur PID piloté par les données	16
II.2.1	Technique piloté par les données	16
II.2.1.1	Description du système	16
II.2.2	Conception du contrôleur	16
II.2.3	Conception d'un régulateur PID piloté par les données	18
II.2.3.1	Génération de la base de données initiale (étape 1).....	19
II.2.3.2	Calcul de la distance et sélectionner des voisins (étape 2)	19
II.2.3.3	Calcul des paramètres PID (étape 3).....	20
II.2.3.4	Réglage des paramètres PID (étape 4)	20

II.2.3.5	Suppression des données redondantes (étape 5)	23
II.3	Conclusion.....	24
Chapitre III Application au robot SCARA		
III.1	Introduction	25
III.2	Structure de régulateur sous Matlab.....	25
III.2.1	Génération de trajectoire	25
III.3	Simulation et interprétation des résultats	27
III.3.1	Simulation avec régulateur PID simple.....	28
III.3.1.1	Interprétation générale de la première application.....	32
III.3.2	Régulateur PID guidée par les données sans apprentissage	32
III.3.2.1	Interprétation générale de la deuxième application	37
III.3.3	Régulateur PID guidée par les données avec apprentissage.....	37
III.3.3.1	Interprétation générale de la troisième application	43
III.4	Conclusion.....	44
Conclusion générale		45
Références bibliographiques.....		

Liste des figures

Chapitre I

Figure I. 1 Constituants mécaniques d'un robot	3
Figure I. 2 Représentation d'une articulation rotative	4
Figure I. 3 Représentation d'une articulation prismatique	5
Figure I. 4 Constituants d'un robot	6
Figure I. 5 Transformation des repères	7
Figure I. 6 Robot SCARA à deux degré de liberté (2DDL RR)	12
Figure I. 7 Modélisation géométrique d'un robot SCARA à 2DDL.....	12

Chapitre II

Figure II. 1 Schéma fonctionnel du système de contrôle DD-PID	24
--	----

Chapitre III

Figure III. 1 Schéma de simulation du système globale.....	25
Figure III. 2 Génération de trajectoire.....	26
Figure III. 3 Trajectoire utilisée.....	27
Figure III. 4 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.....	29
Figure III. 5 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.....	30
Figure III. 6 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.....	31
Figure III. 7 Schéma bloc de simulation de régulateur PID basé sur les données sans apprentissage.....	32
Figure III. 8 Base des données pour le régulateur 1 et le régulateur 2.....	33
Figure III. 9 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.....	34

Figure III. 10 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.	35
Figure III. 11 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.	36
Figure III. 12 Schéma bloc de simulation de régulateur PID basé sur les données avec apprentissage.	37
Figure III. 13 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.	38
Figure III. 14 Base des données pour le régulateur 1 et le régulateur 2 à l’instant initial.	39
Figure III. 15 Base des données pour le régulateur 1 et le régulateur 2 à l’instant final.	39
Figure III. 16 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.	40
Figure III. 17 Base des données pour le régulateur 1 et le régulateur 2 à l’instant initial.	41
Figure III. 18 Base des données pour le régulateur 1 et le régulateur 2 à l’instant final.	41
Figure III. 19 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.	42
Figure III. 20 base des données pour le régulateur 1 et le régulateur 2 à l’instant initial.	43
Figure III. 21 base des données pour le régulateur 1 et le régulateur 2 à l’instant final.	43

Introduction générale

Introduction générale

Ces dernières années, la robotique occupe un grand rôle dans l'industrie. Elle représente un champ d'application intéressant pour des disciplines telles que la mécanique, l'électronique, l'automatique, l'informatique...

La robotique a marqué l'évolution du monde technologique. Son objectif de base est de pouvoir déplacer des objets conformément aux trajectoires prescrites, dans le but de remplacer l'homme lors de réalisations des tâches pénibles et dangereuses ou déplacement d'objets lourds...

Le robot SCARA est un type de robot industriel, il est disposé des bras articulés à deux liaisons similaires aux bras humains, cette caractéristique permet au bras de s'étendre dans des zones confinées, puis de se rétracter ou de se "plier" hors du chemin.

La commande guidée par les données est une technique de commande avancée de l'automatique. Le principe de cette technique est d'utiliser les données historique mesurées sur le processus à l'intérieur du contrôleur en temps réel afin d'anticiper le futur comportement du procédé, cette commande peut être utilisée pour commander des systèmes complexes comportant plusieurs entrées et sorties où le simple régulateur PID est insuffisant. Cette technique est particulièrement intéressante lorsque les systèmes possèdent des retards importants, des réponses inverses et de nombreuses perturbations. Les principaux utilisateurs de la commande guidée par les données sont les raffineries de pétrole, l'industrie chimique et agro- alimentaire, la métallurgie, l'aérospatiale... [1].

Notre travail subdivisé en trois chapitres.

Dans le premier chapitre, nous présenterons une description générale et quelques définitions sur les robots manipulateurs ainsi nous nous focaliserons particulièrement sur la modélisation dynamique du robot SCARA à deux degrés de liberté.

Dans le deuxième chapitre, nous présenterons une conception pour les contrôleurs PID basés sur la technique pilotée par les données qui se déroulera en quelques étapes.

Dans le troisième chapitre, nous allons appliquer la commande guidée par les données sur un robot manipulateur de deux degrés de liberté (SCARA) en utilisant le Matlab, et Nous effectuons des simulations et comparerons les résultats obtenus en utilisant trois méthodes et essais à base de régulateurs PID.

Nous terminerons par une conclusion générale.

CHAPITRE I

Modélisation des robots manipulateurs

I.1 Introduction

Aujourd'hui la robotique est considérée comme un grand domaine d'activité scientifique et technique pour son impact sur l'industrie. Dans ce chapitre on va présenter des généralités sur les robots manipulateurs, les types de modèles (géométrique, cinématique, dynamique) qui existent et ensuite on passera à la modélisation du robot SCARA.

I.2 Historique

L'invention du robot industriel remonte à 1954 quand George Devol a déposé un brevet pour le premier robot. La première entreprise à produire un robot était Unimation, fondée en 1956 par George Devol et Joseph Engelberger. Le premier robot a été mis en service dans une usine de General Motors en 1961 pour extraire des pièces d'une machine de moulage sous pression (Siciliano and Khatib, 2008). Le Stanford Arm a été conçu en 1969 par Victor Scheinman (Scheinman, 1968) comme un prototype pour la recherche. La conception de robots a par la suite été fortement influencée par les concepts de Scheinman. En 1973, la société ASEA (ABB aujourd'hui) a présenté le premier robot commandé par un microprocesseur, l'IRB-6, ce qui a permis le mouvement en trajectoire continue, une condition préalable pour le soudage à l'arc ou l'usinage [2].

I.3 Définition d'un robot

Un robot est une machine automatique pouvant être programmée pour effectuer des tâches diverses. Le mot robot vient du tchèque robota ce qui signifie « travail forcé ».

Un robot est un objet capable de mouvement, qui comporte des capteurs lui fournissant une information sur son environnement : images, lumières, sons, textures...

Enfin, cet objet mobile intègre un ordinateur et cette intelligence électronique sert à lui dicter un comportement donné en fonction de ce qui est perçu par les capteurs [3].

I.4 Définitions des robots manipulateurs

C'est un manipulateur qui doit être reprogrammable multifonctionnel conçu pour déplacer des matériaux, des pièces, des outils ou tout autre dispositif spécialisé au moyen d'une série de mouvements programmés et d'accomplir une variété d'autres tâches[4].

L'ISO (International Standard Organization) l'a défini comme étant une machine mue par un mécanisme incluant plusieurs degrés de libertés, ayant souvent l'apparence d'un ou plusieurs

bras se terminant par un poignet capable de tenir des outils, des pièces ou un dispositif d'inspection[5].

I.5 Constituants mécaniques d'un robot

C'est le squelette du robot. Le choix de l'articulation déterminera les mouvements possibles et guidera le type d'utilisation. Chaque axe de rotation et translation a un degré de liberté le long duquel le membre (bras, jambe) peut se déplacer.

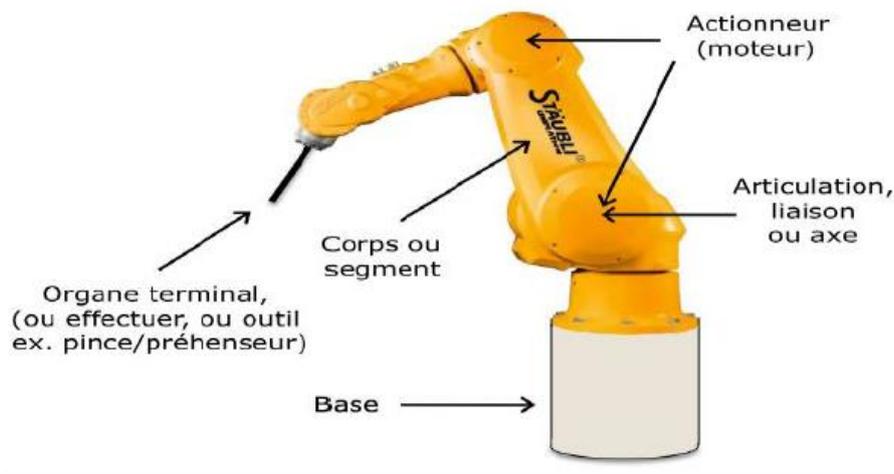


Figure I. 1 Constituants mécaniques d'un robot

I.5.1 Organe terminal

Sous le terme organe terminal, on regroupe tout dispositif destiné à manipuler des objets (dispositifs de serrage, dispositifs agnatiques, à dépression,...), ou à les transformer (outils, torche de soudage, pistolet de peinture,...). En d'autres termes, il s'agit d'une interface permettant au robot d'interagir avec son environnement.

Un organe terminal peut être multifonctionnel, au sens où il peut être équipé de plusieurs dispositifs ayant des fonctionnalités différentes. Il peut aussi être monofonctionnel, mais interchangeable. Un robot, enfin, peut-être multi-bras, chacun des bras portant un organe terminal différent. On utilisera indifféremment le terme organe terminal, préhenseur, outil ou effecteur pour nommer le dispositif d'interaction fixé à l'extrémité mobile de la structure mécanique [6].

I.5.2 Système mécanique articulé

Un système mécanique articulé (S.M.A.) est un mécanisme dont la structure est plus ou moins similaire à celle d'un bras humain. Il permet de remplacer ou de prolonger son mouvement. Son

rôle est d'amener l'organe terminal dans une situation donnée (position et orientation) en fonction de caractéristiques de vitesse et d'accélération données. Sa structure est une chaîne cinématique interconnectée par des articulations. Sa motorisation est assurée par des actionneurs électriques pneumatiques ou hydrauliques, qui transmettent leur mouvement aux articulations.

I.5.2.1 Articulations

Une articulation lie deux corps successifs, en limitant le nombre de degré de liberté de l'un par rapport à l'autre. Le degré de liberté m de l'articulation est entre 0 et 6 ($0 \leq m \leq 6$), lorsque $m=1$, ce qui est le cas le plus fréquent en robotique, On distingue principalement deux types d'articulations: soit rotative soit prismatique.

I.5.2.1.1 Articulation rotative

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

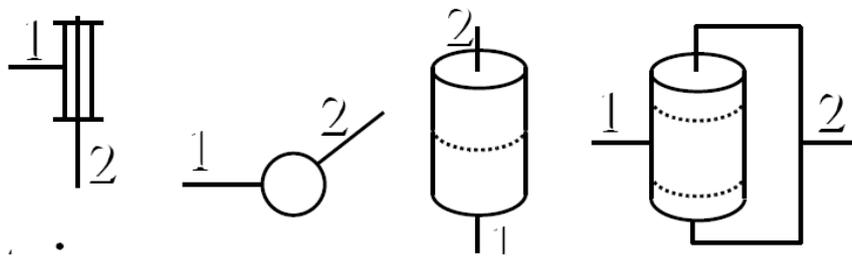


Figure I. 2 Représentation d'une articulation rotative

I.5.2.1.2 Articulation prismatique

Il s'agit d'une articulation de type glissière, notée P réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe [7].

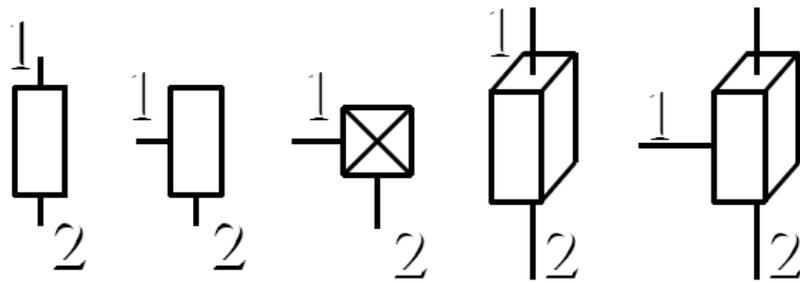


Figure I. 3 Représentation d'une articulation prismatique

I.5.3 Actionneurs

Les actionneurs sont les dispositifs de conversion d'énergie utilisée à l'intérieur d'un robot. La fonction principale des actionneurs est de convertir l'énergie en un phénomène physique utilisable comme par exemple les mouvements.

Ils utilisent fréquemment des moteurs. On trouve de plus en plus de moteurs à commutation électronique, ou pour de petits robots, des moteurs pas à pas.

Pour les robots devant manipuler de très lourdes charges (par exemple, une pelle mécanique), les actionneurs sont le plus souvent hydrauliques, agissant en translation (vérin, hydraulique) ou en rotation (moteur hydraulique). Les actionneurs pneumatiques sont d'un usage général pour les manipulateurs à cycles (robots tout ou rien) [8].

I.5.4 Capteurs

Les capteurs traduisent le phénomène physique caractérisant l'environnement en un signal électrique exploitable. Le robot peut être équipé par différents capteurs ; Les capteurs sont dits proprioceptifs lorsqu'ils mesurent l'état interne du robot (positions et vitesses des articulations), et extéroceptifs lorsqu'ils recueillent des informations sur l'environnement (détection de présence, de contact, mesure de distance, vision artificielle).

I.5.5 Système de commande et de traitement de l'information

La partie commande synthétise les consignes des asservissements pilotant les actionneurs, (génération des signaux de commande). Le choix du système de commande est guidé par :

- ❖ Complexité de la mécanique,
- ❖ L'ampleur des tâches à réaliser,

- ❖ Performances souhaitées.

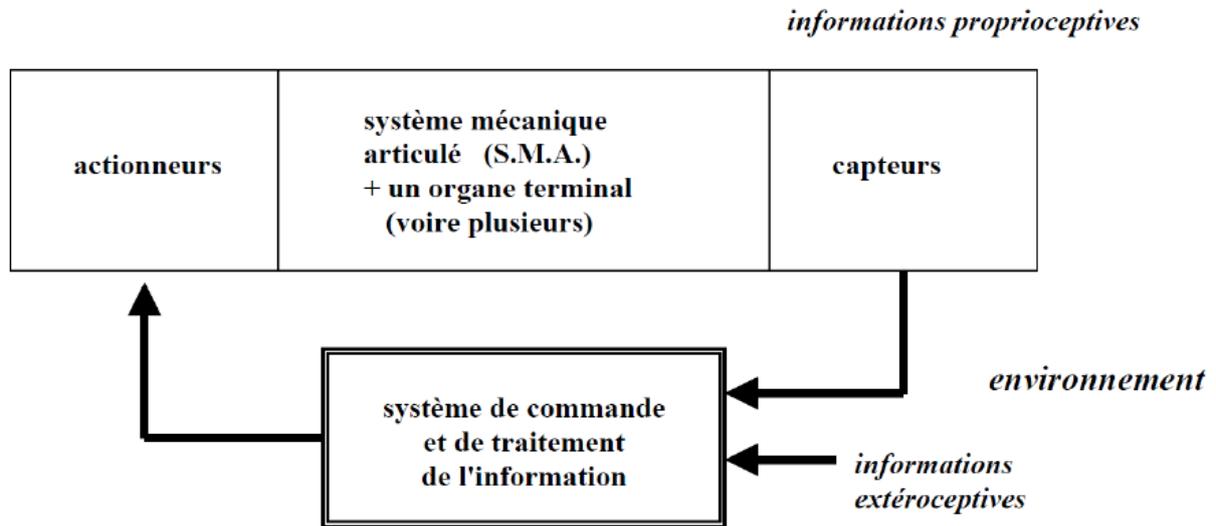


Figure I. 4 Constituants d'un robot

I.6 Degré de liberté

Le degré de liberté d'un robot manipulateur est la somme des degrés de liberté de ses articulations. Le degré de liberté d'une articulation est le nombre de mouvements du corps aval par rapport au corps amont.

I.7 Modélisation des robots manipulateurs

La conception et la commande des robots nécessitent le calcul de certains modèles mathématiques, 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 [8].

I.7.1 Transformation homogène

I.7.1.1 Transformation des repères

Le formalisme mathématique fait appel aux matrices de transformation homogènes de dimension (4x4). La matrice homogène ${}^i T_j$ représente la transformation permettant de passer du repère R_i au repère R_j telle que :

$${}^i T_j = \begin{bmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i P_j \end{bmatrix} = \begin{bmatrix} s^x & n^x & a^x & P^x \\ s^y & n^y & a^y & P^y \\ s^z & n^z & a^z & P^z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{I.1})$$

La matrice de transformation s'écrit aussi sous la forme :

$${}^i T_j = \begin{bmatrix} {}^i A_j & {}^i P_j \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^i s_j & {}^i n_j & {}^i a_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{I.2})$$

Avec :

- ${}^i A_j$ est la matrice de rotation.
- ${}^i P_j$ est le vecteur de translation.

où $({}^i s_j \quad {}^i n_j \quad {}^i a_j)$ désignent respectivement les vecteurs unitaires suivant les axes (x_j, y_j, z_j) du repère R_j exprimés dans le repère R_i , et ${}^i P_j$ est le vecteur exprimant l'origine du repère R_j dans le repère R_i . Les vecteurs ${}^i s_j \quad {}^i n_j \quad {}^i a_j$ de la matrice de rotation ${}^i A_j$ sont les cosinus directeurs.

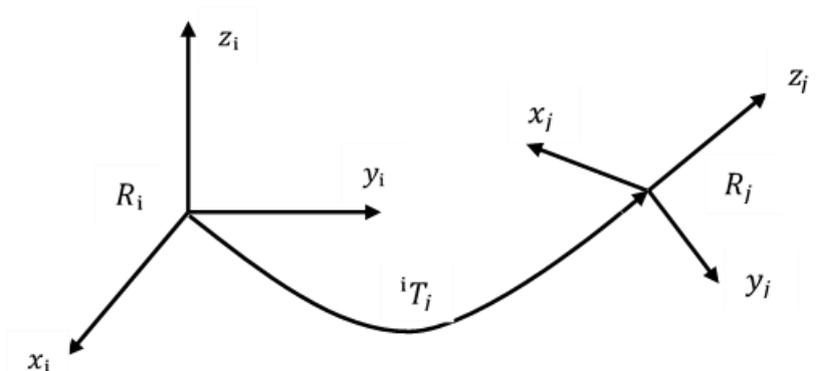


Figure I. 5 Transformation des repères

I.7.1.2 Matrice de rotation

Comme exemples de matrices de rotation, on peut présenter pour les rotations autour des axes X, Y et Z comme suit :

Matrice de rotation par rapport à l'axe x

$$ROT(x, \theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (I.3)$$

Matrice de rotation par rapport à l'axe y

$$ROT(y, \theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (I.4)$$

Matrice de rotation par rapport à l'axe z

$$ROT(z, \theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (I.5)$$

I.7.2 Modélisation géométrique

La modélisation géométrique permet d'exprimer la situation du dispositif terminal en fonction de la configuration du mécanisme, et inversement. Il existe deux modèles :

- Modèle géométrique direct.
- Modèle géométrique inverse.

Pour analyser le comportement mécanique de la structure mécanique d'un robot ou d'un bras manipulateur, il est nécessaire de lier un repère orthonormé à chacun de ses corps et un référentiel attaché à la base du robot ou du bras manipulateur.

I.7.2.1 Modèle géométrique direct

Le modèle géométrique direct (MGD) est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal, c'est-à-dire les coordonnées opérationnelles du robot, en fonction de ses coordonnées articulaires.

Le modèle géométrique direct du robot peut être représenté par la relation.

$$x = F(q) \quad (\text{I. 6})$$

q : Vecteur des variables articulaires.

x : Coordonnées opérationnelles.

I.7.2.2 Modèle géométrique inverse

Le modèle géométrique inverse permet de déterminer le vecteur des variables articulaires à partir du vecteur de coordonnées opérationnelles.

$$q = F^{-1}(x) \quad (\text{I. 7})$$

I.7.3 Modélisation cinématique

Le modèle cinématique exprime les relations entre les vitesses articulaires de chaque articulation et les vitesses cartésiennes d'un point de la chaîne cinématique ; généralement l'organe terminal.

I.7.3.1 Modèle cinématique direct

Le modèle cinématique direct décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires, Il est noté :

$$\frac{dx}{dt} = \frac{\partial F}{\partial q} \frac{dq}{dt} \quad (\text{I. 8})$$

$\frac{dx}{dt} = \dot{X}$: Vecteur de position et d'orientation de l'organe terminal.

$\frac{\partial F}{\partial q} = J(q)$: Matrice jacobéenne.

$\frac{dq}{dt} = \dot{q}$: Vecteur de vitesse articulaire.

I.7.3.2 Modèle cinématique inverse

Le modèle cinématique inverse nous permet l'obtention des vitesses qu'il faut appliquer aux actionneurs en fonction des vitesses désirées pour l'organe terminal dans l'espace de la tâche, il est donné sous la forme suivante :

$$\dot{q} = J^{-1}(q)\dot{X} \quad (\text{I. 9})$$

I.7.4 Modélisation dynamique

Le modèle dynamique joue un rôle important dans la conception et le fonctionnement des robots. Pour la conception, le modèle dynamique inverse peut être utilisé pour choisir les actionneurs, alors que le modèle dynamique direct est utilisé pour effectuer des simulations, afin de tester les performances du robot.

Plusieurs approches sont proposées pour obtenir le modèle dynamique des robots, les plus utilisées sont la formulation de Newton-Euler et la formulation de Lagrange. L'approche de Newton-Euler est basée sur l'équilibre des forces et les moments qui agissent entre les articulations. La formulation de Lagrange est une approche basée sur l'énergie. Pour le même manipulateur, les deux approches donnent les mêmes équations du mouvement [10].

I.7.4.1 Modèle dynamique inverse

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \quad (\text{I. 10})$$

q : Vecteur des variables articulaires de dimension $(n \times 1)$.

\dot{q} : Vecteur des vitesses articulaires de dimension $(n \times 1)$.

\ddot{q} : Vecteur des accélérations articulaires de dimension $(n \times 1)$.

τ : Vecteur des couples appliqués aux articulations de dimension $(n \times 1)$.

$M(q)$: Matrice d'inertie du robot de dimension $(n \times n)$ symétrique définie positive.

$C(q, \dot{q})\dot{q}$: Vecteur des forces de Coriolis et centrifuges de dimension $(n \times 1)$.

$G(q)$: Vecteur des forces de gravité de dimension $(n \times 1)$.

I.7.4.2 Modèle dynamique direct

Exprime la position, la vitesse et l'accélération des articulations en fonction des couples appliqués, il est obtenu par inversion du modèle précédent.

$$\ddot{q} = M(q)^{-1}[\tau - C(q, \dot{q})\dot{q} - G(q)] \quad (\text{I. 11})$$

I.7.4.3 Formalisme d'Euler-Lagrange

Le formalisme d'Euler-Lagrange amène à un algorithme composé d'équations dynamiques représentant le mouvement. La méthode d'Euler-Lagrange est utilisée pour modéliser et visualiser la dynamique d'un robot à travers des équations de mouvement. Elle est donnée par l'équation suivante :

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} + \frac{\partial E_d}{\partial \dot{q}_i} = \tau_i \quad (i = 1 \dots n) \quad (\text{I.12})$$

Avec :

$$L = E_c - E_p \quad (\text{I.13})$$

τ_i : La force où le couple généralisé d'articulation.

n : Le nombre de degré de liberté.

q_i : La coordonnée généralisée de l'articulation i .

\dot{q}_i : La dérivée de la coordonnée généralisée.

L : Lagrangien du système.

E_c : Énergie cinétique.

E_p : Énergie potentielle.

E_d : L'énergie de dissipation en cas de présence de frottement visqueux

I.8 Modélisation d'un robot manipulateur type SCARA à 2d.d.l

Le robot SCARA «Selective Compliance Arm for Robot Assembly» est un robot manipulateur à 2 axes rotatifs (RR). L'enveloppe de travail cylindrique permet une utilisation maximale de l'espace de travail.

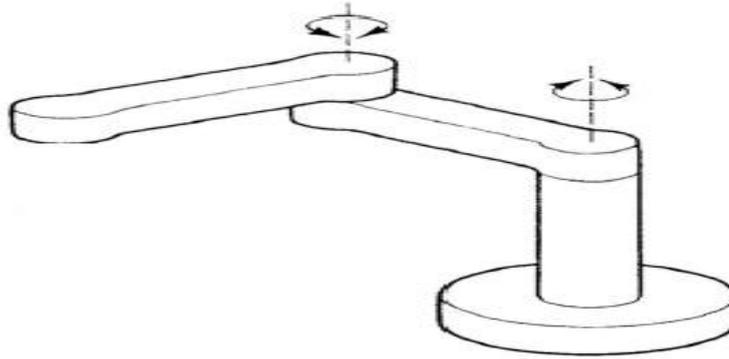


Figure I. 6 Robot SCARA à deux degré de liberté

I.8.1 Modèle géométrique direct

Ce modèle permet de savoir la position exacte de l'organe terminal à partir de les angles θ .

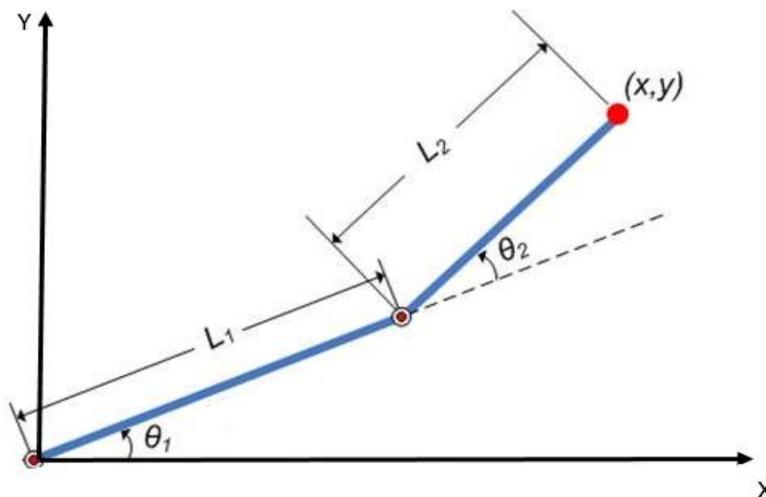


Figure I. 7 Modélisation géométrique d'un robot SCARA à 2DDL

Le modèle géométrique direct est donné par :

$$\begin{cases} x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) \\ y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) \end{cases} \quad (\text{I. 14})$$

I.8.2 Modèle géométrique inverse

Le modèle géométrique inverse est donné par les équations suivantes :

$$\theta_2 = \text{acos} \left(\frac{(x^2 + y^2 - l_1^2 - l_2^2)}{(2l_1l_2)} \right) \quad (\text{I. 15})$$

$$\theta_1 = \text{atg} \frac{y}{x} - \text{acos} \left(\frac{(x^2 + y^2 + l_1^2 - l_2^2)}{(2l_1\sqrt{x^2 + y^2})} \right) \quad (\text{I. 16})$$

I.8.3 Modèle cinématique direct

Le modèle cinématique direct du robot est donnée par la relation suivante :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -l_1 \sin(\theta_1) - l_2 \sin(\theta_1 + \theta_2) & -l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (\text{I. 17})$$

I.8.4 Modèle cinématique inverse

Le MCI est obtenu par l'inversion de la matrice $J(q)$:

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = J(q)^{-1} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad (\text{I. 18})$$

I.8.5 Modèle dynamique

Dans ce modèle on utilise le formalisme de Lagrange pour trouver l'équation qui décrit le modèle dynamique du robot :

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} + \frac{\partial E_d}{\partial \dot{\theta}_i} = \tau \quad (\text{I. 19})$$

Avec :

L est données par équation (2. 13).

$E_p=0$, car le mouvement est dans le plan horizontal.

L'énergie cinétique du système est donnée par l'équation suivant :

$$E_c = E_{c_1} + E_{c_2} + E_{c_0} + E_{c_{m_1}} + E_{c_{m_2}} \quad (\text{I. 21})$$

Avec :

E_{c_1} : L'énergie cinétique d'articulation 1.

E_{c_2} : L'énergie cinétique d'articulation 2.

E_{c_0} : L'énergie cinétique de la masse, portée par l'organe terminal.

$E_{c_{m_1}}$ et $E_{c_{m_2}}$: L'énergie cinétique des deux moteurs.

E_d : L'énergie de dissipation en cas de présence de frottement visqueux

➤ L'énergie cinétique d'articulation 1 :

$$E_{c_1} = \frac{1}{2} I_1 \dot{\theta}_1^2 \quad (\text{I. 22})$$

Avec le moment d'inertie I_1 est :

$$I_1 = \frac{1}{3} m_1 l_1^2 \quad (\text{I. 23})$$

➤ L'énergie cinétique d'articulation 2 :

$$E_{c_2} = \frac{1}{2} m_2 v_2^2 + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \quad (\text{I. 24})$$

C'est-à-dire :

$$E_{c_2} = \frac{1}{2} m_2 \left[\left(-l_1 \sin(\theta_1) \dot{\theta}_1 - \frac{l_2}{2} \sin(\theta_1 + \theta_2) * (\dot{\theta}_1 + \dot{\theta}_2) \right)^2 + \left(l_1 \cos(\theta_1) \dot{\theta}_1 + \frac{l_2}{2} \cos(\theta_1 + \theta_2) * (\dot{\theta}_1 + \dot{\theta}_2) \right)^2 \right] + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \quad (\text{I. 25})$$

➤ L'énergie cinétique de la masse m_0 :

$$E_{c_0} = \frac{1}{2} m_0 v_0^2 \quad (\text{I. 26})$$

Ce qui donne :

$$E_{c_0} = \frac{1}{2} m_0 \left[\left(-l_1 \sin(\theta_1) \dot{\theta}_1 - l_2 \sin(\theta_1 + \theta_2) * (\dot{\theta}_1 + \dot{\theta}_2) \right)^2 + \left(l_1 \cos(\theta_1) \dot{\theta}_1 + l_2 \cos(\theta_1 + \theta_2) * (\dot{\theta}_1 + \dot{\theta}_2) \right)^2 \right] \quad (\text{I. 27})$$

➤ L'énergie cinétique des deux moteurs :

$$E_{c_{m_1}} = \frac{1}{2} n_1^2 I_{m_1} \dot{\theta}_1^2 \quad (\text{I. 28})$$

$$E_{c_{m_2}} = \frac{1}{2} n_2^2 I_{m_2} \dot{\theta}_2^2 \quad (\text{I. 29})$$

➤ L'énergie de dissipation en cas de présence de frottement visqueux :

$$E_d = \frac{1}{2} f_{v_1} \dot{\theta}_1^2 + \frac{1}{2} f_{v_2} \dot{\theta}_2^2 \quad (\text{I. 30})$$

En appliquant le formalisme de Lagrange, on trouve :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} \frac{d}{dt} \frac{\partial E_{c_1}}{\partial \dot{\theta}_1} + \frac{d}{dt} \frac{\partial E_{c_2}}{\partial \dot{\theta}_1} + \frac{d}{dt} \frac{\partial E_{c_0}}{\partial \dot{\theta}_1} + \frac{d}{dt} \frac{\partial E_{c_m}}{\partial \dot{\theta}_1} \\ \frac{d}{dt} \frac{\partial E_{c_1}}{\partial \dot{\theta}_2} + \frac{d}{dt} \frac{\partial E_{c_2}}{\partial \dot{\theta}_2} + \frac{d}{dt} \frac{\partial E_{c_0}}{\partial \dot{\theta}_2} + \frac{d}{dt} \frac{\partial E_{c_m}}{\partial \dot{\theta}_2} \end{bmatrix} - \begin{bmatrix} \frac{\partial E_{c_1}}{\partial \theta_1} + \frac{\partial E_{c_2}}{\partial \theta_1} + \frac{\partial E_{c_0}}{\partial \theta_1} + \frac{\partial E_{c_m}}{\partial \theta_1} \\ \frac{\partial E_{c_1}}{\partial \theta_2} + \frac{\partial E_{c_2}}{\partial \theta_2} + \frac{\partial E_{c_0}}{\partial \theta_2} + \frac{\partial E_{c_m}}{\partial \theta_2} \end{bmatrix} + \begin{bmatrix} \frac{\partial E_d}{\partial \dot{\theta}_1} \\ \frac{\partial E_d}{\partial \dot{\theta}_2} \end{bmatrix} \quad (\text{I. 31})$$

I.9 Conclusion

Nous avons présenté dans ce chapitre une description générale et quelques définitions sur les robots manipulateurs. Nous avons donné, également, les outils mathématiques utilisés dans la robotique pour trouver le modèle géométrique, le modèle cinématique et dynamique des robots manipulateurs. Nous nous sommes focalisés particulièrement sur la modélisation dynamique du robot SCARA à deux degré de liberté (2ddl).

Dans le chapitre suivant on va s'intéresser à la commande guidée par des données.

CHAPITRE II

Commande guidée par les données

II.1 Introduction

Ce chapitre décrit un schéma de conception pour les contrôleurs PID, basé sur la technique pilotée par les données. La principale caractéristique du schéma de contrôle DD-PID est que les paramètres PID sont mis à jour en fonction de l'erreur de contrôle en ligne et qu'ils sont stockés dans la base de données. Par conséquent, l'effet d'apprentissage est progressivement accumulé dans la base de données. Afin d'éviter l'accumulation excessive des données stockées, un algorithme est proposé qui réduit le besoin de stockage en mémoire et les coûts de calcul [11].

II.2 Contrôleur PID piloté par les données

II.2.1 Technique piloté par les données

Afin de créer la technique pilotée par les données, plusieurs considérations mathématiques doivent d'abord être effectuées.

II.2.1.1 Description du système

Tout d'abord, on suppose qu'un système non linéaire est décrit par l'équation suivante :

$$y(t) = f(\varnothing(t - 1)) \quad (\text{II.1})$$

$y(t)$: Sortie du système.

$f(\cdot)$: Fonction non linéaire.

$\varnothing(t - 1)$: Vecteur d'information, est défini par ce qui suit :

$$\varnothing(t - 1) := [y(t - 1), \dots, y(t - n_y), u(t - 1), \dots, u(t - n_u)] \quad (\text{II.2})$$

$u(t)$: Entrée de commande.

n_y, n_u : Les ordres de la sortie et de l'entrée du système.

Comme l'objet contrôlé est décrit à l'aide d'éléments de $\varnothing(t - 1)$, on suppose que le signe de la dérivée partielle de $y(t)$ par rapport à $u(t - 1)$ est connu. De plus, cela signifie que le signe de la matrice jacobienne du système est aussi connu.

II.2.2 Conception du contrôleur

La loi de commande avec une structure PID est considérée comme suite :

$$\begin{aligned} \Delta u(t) &= \frac{k_c T_c}{T_I} e(t) - k_c \left(\Delta + \frac{T_D}{T_s} \Delta^2 \right) y(t) \\ &= K_I e(t) - K_p \Delta y(t) - K_D \Delta^2 y(t) \end{aligned} \quad (\text{II.3})$$

$e(t)$: Signal d'erreur de commande défini par :

$$e(t) = r(t) - y(t) \quad (\text{II.4})$$

$r(t)$: Signal de référence.

k_c : Gain proportionnel.

T_I : Temps d'intégration.

T_D : Temps de dérivé.

T_s : Période d'échantillonnage.

Les gains K_p , K_I et K_D sont obtenus à partir des relations suivantes :

$$K_p = k_c$$

$$K_I = \frac{k_c T_s}{T_I}$$

$$K_D = \frac{k_c T_D}{T_s}$$

Δ : L'opérateur de différenciation défini par : $\Delta = 1 - z^{-1}$

Cependant, il est assez difficile d'obtenir une bonne performance de contrôle en raison des non linéarités si les paramètres PID (K_p , K_I et K_D) en (II.3) sont fixes. Pour pouvoir ajuster les paramètres PID en ligne correspondant aux propriétés du système. On utilise la loi de contrôle PID suivante avec des paramètres PID variant dans le temps :

$$\Delta u(t) = K_I(t)e(t) - K_p(t)\Delta y(t) - K_D(t)\Delta^2 y(t) \quad (\text{II.5})$$

L'équation (II.5) peut être réécrite en utilisant les relations suivantes :

$$u(t) = g(\phi'(t)) \quad (\text{II.6})$$

$$\phi'(t) := [\mathbf{K}(t), r(t), y(t), y(t-1), y(t-2), u(t-1)] \quad (\text{II.7})$$

$$\mathbf{K}(t) := [K_p(t), K_I(t), K_D(t)] \quad (\text{II.8})$$

Où $g(\varphi'(t))$ désigne une fonction linéaire composée de termes qui sont donnés par (II.7) En substituant (II.6) et (II.7) dans (II.1) et (II.2), on obtient l'équation suivante :

$$y(t+1) = h(\tilde{\varphi}(t)) \quad (\text{II.9})$$

$$\tilde{\varphi}(t) := [y(t), \dots, y(t-n_y+1), \mathbf{K}(t), r(t), u(t-1), \dots, u(t-n_u+1)] \quad (\text{II.10})$$

$h(\tilde{\varphi}(t))$ désigne une fonction non linéaire qui a des éléments donnés par (II.10).

Par conséquent, le vecteur du paramètres du PID, $\mathbf{K}(t)$, est lié au vecteur d'information, il s'exprime par :

$$\mathbf{K}(t) = F(\bar{\varphi}(t)) \quad (\text{II.11})$$

$$\bar{\varphi}(t) := \begin{bmatrix} y(t+1), y(t), \dots, y(t-n_y+1), r(t), u(t-1), \dots, \\ u(t-n_u+1) \end{bmatrix} \quad (\text{II.12})$$

La sortie $y(t+1)$ inclus dans (II.12) ne peut pas être obtenu en temps, donc on la remplace par $r(t+1)$ donc $\bar{\varphi}(t)$ est nouvellement réécrit comme suit :

$$\bar{\varphi}(t) := \begin{bmatrix} r(t+1), r(t), y(t), \dots, u(t-1), y(t-n_y+1), \dots, \\ u(t-n_u+1) \end{bmatrix} \quad (\text{II.13})$$

II.2.3 Conception d'un régulateur PID piloté par les données

Après avoir effectué les préparations ci-dessus, il est possible de concevoir un nouveau contrôleur PID basé sur la technique "piloté par les données". La procédure de conception du contrôleur est résumée comme suit.

II.2.3.1 Génération de la base de données initiale (étape 1)

La technique basée sur les données ne peut pas fonctionner si les données historiques ne sont pas enregistrées, car il n'y aura aucune information avec laquelle travailler.

Il est nécessaire de créer d'abord une base de données basée à partir des données historiques du système sous contrôle.

Les paramètres PID sont initialement calculés. la matrice $\phi(j)$ dans l'équation suivante est générée comme base de données initiale :

où

$$\phi(j) = [\bar{\phi}(j), \mathbf{K}(j)], \quad j = 1, 2, \dots, N(0) \quad (\text{II.14})$$

$\bar{\phi}(j)$ et $\mathbf{K}(j)$ sont données par les équations (II.13) et (II.8).

$N(0)$: Nombre de vecteurs d'information stockés dans la base de données initiale.

Un ensemble de paramètres d'un PID fixe sont choisis comme étant typiques. Puis tous les paramètres PID inclus dans les vecteurs d'information initiaux peuvent être égaux. Exprimé numériquement, c'est-à-dire que $\mathbf{K}(1) = \mathbf{K}(2) = \dots = \mathbf{K}(N(0))$ à l'étape initiale.

II.2.3.2 Calcul de la distance et sélectionner des voisins (étape 2)

Une distance entre la requête (c'est le vecteur d'information qui indique l'état actuel du système) $\bar{\phi}(t)$ et le vecteur d'information $\bar{\phi}(j)$ dans une base de données, est calculée par la formule suivante :

$$d(\bar{\phi}(t), \bar{\phi}(j)) = \sum_{l=1}^{n_y+n_u+1} \left| \frac{\bar{\phi}_l(t) - \bar{\phi}_l(j)}{\max_m \bar{\phi}_l(m) - \min_m \bar{\phi}_l(m)} \right|, \quad (j = 1, 2, \dots, N(t)) \quad (\text{II.15})$$

où

$N(t)$: Nombre de vecteurs d'information stockés dans la base de données lorsque la requête $\bar{\phi}(t)$ est donnée.

$\bar{\phi}_l(j)$: Désigne l' $l^{\text{ème}}$ élément du $j^{\text{ème}}$ vecteur d'information.

$\bar{\phi}_l(t)$: Désigne l' $l^{\text{ème}}$ élément de la requête à t .

Parmi les $l^{\text{ème}}$ éléments de tous les vecteurs d'information ($\bar{\phi}_l(j), (j = 1, 2, \dots, N(t))$) stockés dans la base de données, l'élément maximum est noté $\max_m \bar{\phi}_l(m)$, et de même $\min_m \bar{\phi}_l(m)$ est l'élément minimum. Dans cette méthode les k -éléments avec les plus petites distances entre eux sont choisis parmi tous les vecteurs d'information.

II.2.3.3 Calcul des paramètres PID (étape 3)

En utilisant les k -voisins sélectionnés dans l'**étape 2**, l'ensemble approprié de paramètres PID est calculé autour de la requête en utilisant la moyenne pondérée suivante :

$$\mathbf{K}(t) = \sum_{i=1}^k w_i \mathbf{K}(i) \quad , \quad \sum_{i=1}^k w_i = 1 \quad (\text{II. 16})$$

w_i Désigne le poids correspondant au $i^{\text{ème}}$ vecteur d'information $\bar{\phi}(i)$, dans les voisins sélectionnés. Celle-ci est calculée par :

$$w_i = \sum_{l=1}^{n_y+n_u+1} \left(1 - \frac{(\bar{\phi}_l(t) - \bar{\phi}_l(i))^2}{(\max_m \bar{\phi}_l(m) - \min_m \bar{\phi}_l(m))^2} \right) \quad (\text{II. 17})$$

En utilisant les paramètres PID calculés en (II.16), l'entrée de commande est générée, et la sortie $y(t + 1)$ est mesurée.

II.2.3.4 Réglage des paramètres PID (étape 4)

Dans le cas où les performances de régulation adéquates ne peuvent pas être obtenues à l'aide des paramètres PID calculés en **étape 3**, ces paramètres de contrôle doivent être mis à jour et stockés dans la base de données. Il est nécessaire d'ajuster les paramètres PID pour que l'erreur de régulation soit diminuée. La méthode suivante est utilisée afin de modifier les paramètres du PID :

$$\mathbf{K}^{new}(t) = \mathbf{K}^{old} - \eta \frac{\partial J(t+1)}{\partial \mathbf{K}(t)^{old}} \quad (\text{II. 18})$$

η est un vecteur de coefficients d'apprentissage donné par :

$$\eta = \text{diag}\{\eta_P, \eta_I, \eta_D\} \quad (\text{II. 19})$$

η_P, η_I et η_D indiquent les coefficients d'apprentissage pour chaque gains du PID.

Le critère $J(t + 1)$ est donnée par :

$$J(t + 1) = \frac{1}{2} \varepsilon(t + 1)^2 \quad (\text{II. 20})$$

Avec

$$\varepsilon(t) = y_r(t) - y(t) \quad (\text{II. 21})$$

et $y_r(t)$ sortie du modèle de référence donnée par :

$$y_r(t) = \frac{z^{-1}T(1)}{T(z^{-1})} r(t) \quad (\text{II. 22})$$

où

$$T(z^{-1}) = 1 + t_1 z^{-1} + t_2 z^{-2} \quad (\text{II. 23})$$

$T(z^{-1})$ est basé sur le temps de montée et propriété d'amortissement.

Le polynôme $T(z^{-1})$ est le dénominateur de la version à temps discret de la fonction de transfert en temps continu du second ordre $G(s)$ suivante :

$$G(s) = \frac{1}{1 + \sigma s + \mu(\sigma s)^2} \quad (\text{II. 24})$$

$T(z^{-1})$ est défini comme suit :

$$t_1 = -2 \exp\left(-\frac{\rho}{2\mu}\right) \cos\left(\frac{\sqrt{(2\mu - 1)}}{2\mu} \rho\right) \quad (\text{II. 25})$$

$$t_2 = \exp\left(-\frac{\rho}{\mu}\right) \quad (\text{II. 26})$$

$$\rho = \frac{T_s}{\sigma} \quad (\text{II. 27})$$

$$\mu = 0.25(1 - \delta) + 0.51\delta \quad (\text{II. 28})$$

σ : Temps de montée.

μ : Coefficient d'amortissement.

Le coefficient d'amortissement est ajusté en modifiant δ .

De plus, chaque dérivée partielle de la relation (II.18) se développe comme suit :

$$\begin{aligned}
 \frac{\partial J(t+1)}{\partial K_p(t)} &= \frac{\partial J(t+1)}{\partial \varepsilon(t+1)} \frac{\partial \varepsilon(t+1)}{\partial y(t+1)} \frac{\partial y(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial K_p(t)} \\
 &= \varepsilon(t+1)(y(t) - y(t-1)) \frac{\partial y(t+1)}{\partial u(t)} \\
 \frac{\partial J(t+1)}{\partial K_i(t)} &= \frac{\partial J(t+1)}{\partial \varepsilon(t+1)} \frac{\partial \varepsilon(t+1)}{\partial y(t+1)} \frac{\partial y(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial K_i(t)} \\
 &= -\varepsilon(t+1)e(t) \frac{\partial y(t+1)}{\partial u(t)} \\
 \frac{\partial J(t+1)}{\partial K_d(t)} &= \frac{\partial J(t+1)}{\partial \varepsilon(t+1)} \frac{\partial \varepsilon(t+1)}{\partial y(t+1)} \frac{\partial y(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial K_d(t)} \\
 &= \varepsilon(t+1)((y(t) - 2y(t-1) + y(t-2))) \frac{\partial y(t+1)}{\partial u(t)}
 \end{aligned} \tag{II.29}$$

Il est à noter que des informations a priori sur la matrice jacobienne représentée par $\partial J(t+1)/\partial u(t)$ sont nécessaires pour le calcul(II.29).

Ici, en utilisant la relation $x = |x| \text{sign}(x)$, la matrice jacobienne peut être obtenue par l'équation suivante :

$$\frac{\partial y(t+1)}{\partial u(t)} = \left| \frac{\partial y(t+1)}{\partial u(t)} \right| \text{sign} \left(\frac{\partial y(t+1)}{\partial u(t)} \right) \tag{II.30}$$

Où $\text{sign}(x) = 1(x > 0)$, $-1(x < 0)$. De l'hypothèse mentionnée ci-dessus, le signe de la matrice jacobienne est disponible. Et en incluant $\partial y(t+1)/\partial u(t)$ dans η , l'utilisation de la matrice jacobienne peut être facilitée.

Enfin, la base de données est mise à jour en utilisant les gains PID mis à jour. Cette procédure est exécutée de manière itérative jusqu'à ce que la quantité de correction en (II.18) devienne suffisamment petite.

II.2.3.5 Suppression des données redondantes (étape 5)

Lors de la mise en œuvre de la technique basée sur les données dans des systèmes réels, il faut tenir compte du temps nécessaire pour effectuer les calculs entre (l'étape 1) et (l'étape 2). En effet, toute donnée redondante stockée dans une base de données ferait perdre du temps informatique qui pourrait être utilisé pour ces processus. Pour éviter cela, un algorithme est présenté qui évite de stocker des données en excès. Cet algorithme est implémenté en deux étapes :

a) Première condition

Pour les vecteurs d'information dans lesquels les k -voisins sont exceptés d'avance, les vecteurs d'information qui satisfont la condition suivante sont extraits :

$$d(\bar{\varphi}(t), \bar{\varphi}(i)) \leq \alpha_1, \quad i = 1, 2, \dots, N(t) - k \quad (\text{II. 31})$$

Il est à noter que la distance est uniquement calculée à l'aide des données d'entrée/sortie $\bar{\varphi}$ contenues dans le vecteur d'information. Le fait de remplir la première condition signifie que seuls les vecteurs d'information ayant une courte distance par rapport à la requête $\bar{\varphi}(t)$ sont extraits de la base de données.

b) Deuxième condition

Pour les vecteurs d'information extraits dans la première condition, les vecteurs d'information qui remplissent la condition suivante sont extraits :

$$\sum_{l=1}^3 \left\{ \frac{K_l(i) - K_l^{new}(t)}{K_l^{new}(l)} \right\}^2 \leq \alpha_2 \quad (\text{II. 32})$$

K_1 , K_2 et K_3 signifient respectivement K_P , K_I et K_D

En utilisant la procédure ci-dessus, les données redondantes peuvent être extraites et supprimées de la base de données. Ceci est réalisé en extrayant et en supprimant les vecteurs d'informations de la base de données qui présentent une forte similarité dans la relation entre les gains PID nouvellement générés et les gains PID contenus dans le vecteur d'informations extrait. S'il existe plusieurs vecteurs d'informations qui satisfont à la deuxième condition, seul le vecteur d'informations avec la plus petite valeur dans la deuxième condition est supprimé.

Il est à noter que la requête et les paramètres PID mis à jour correspondants sont toujours stockés dans la base de données. Dans la pratique, α_1 et α_2 devraient être fixés entre 0,1 et 1, mais certains essais et erreurs peuvent être nécessaires.

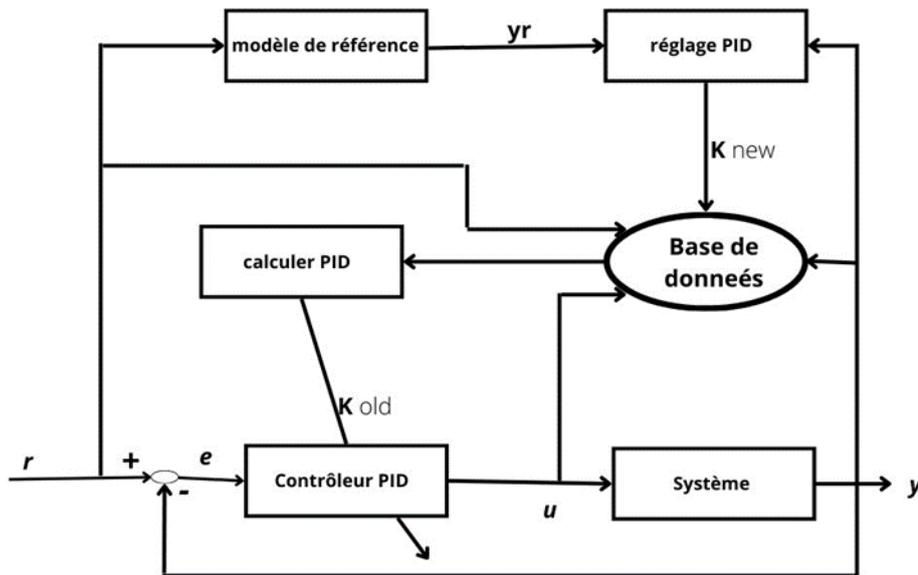


Figure II. 1 Schéma fonctionnel du système de contrôle DD-PID

II.3 Conclusion

Dans ce chapitre, on a décrit un schéma de conception pour les contrôleurs PID, basée sur la technique pilotée par les données, peut être mise à jour en utilisant un ensemble de données d'exploitation, les performances de contrôle sont améliorées à partir d'un contrôleur PID fixe, même si le système sous contrôle a une forte non-linéarité, La méthode proposée peut être appliquée aux systèmes réels tels que des machines, dans le chapitre suivant on va l'appliquer au robot manipulateur.

Chapitre III

Application au robot SCARA

III.1 Introduction

Le système considéré dans ce travail est un robot manipulateur à deux degrés de liberté (SCARA). L'objectif est de faire une régulation en position du robot afin de suivre certaines trajectoires.

Dans ce chapitre, nous allons utiliser trois méthodes et essais à base de régulateurs PID pour simuler la commande guidée par les données. Nous effectuons des simulations et comparerons les résultats obtenus.

III.2 Structure de régulateur sous Matlab

Le schéma suivant représente la structure de la commande d'un robot SCARA. Il regroupe trois blocs, le premier est utilisé pour générer la trajectoire de consigne, le deuxième pour le régulateur et le troisième bloc représente le robot.

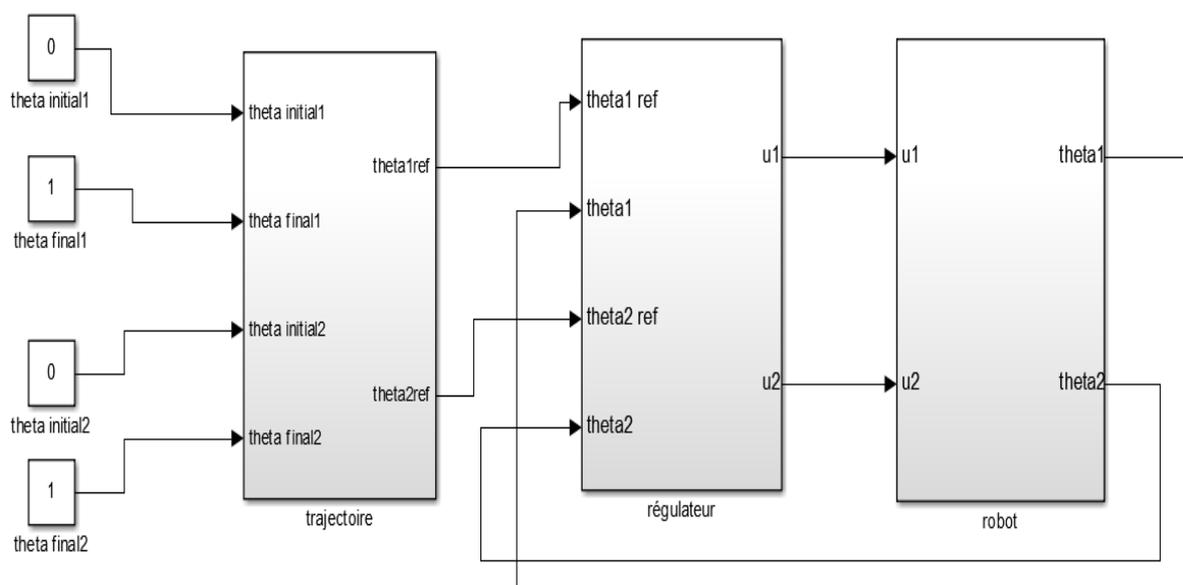


Figure III. 1 Schéma de simulation du système globale.

III.2.1 Génération de trajectoire

Les générateurs de trajectoires ont pour fonction de calculer l'évolution des positions des articulations ou de la situation du robot en fonction du temps. Cette trajectoire de référence définit l'entrée du système de contrôle. Le générateur présente une grande importance car la continuité de la trajectoire produite et de ses dérivées dépend la qualité du mouvement produit.

Cette qualité impacte non seulement la précision de suivi du chemin mais aussi la durée de vie du système mécanique. Il existe plusieurs types de trajectoires et dans ce travail on a utilisé des trajectoires point à point.

Le schéma suivant présente le bloc de simulation de génération trajectoire :

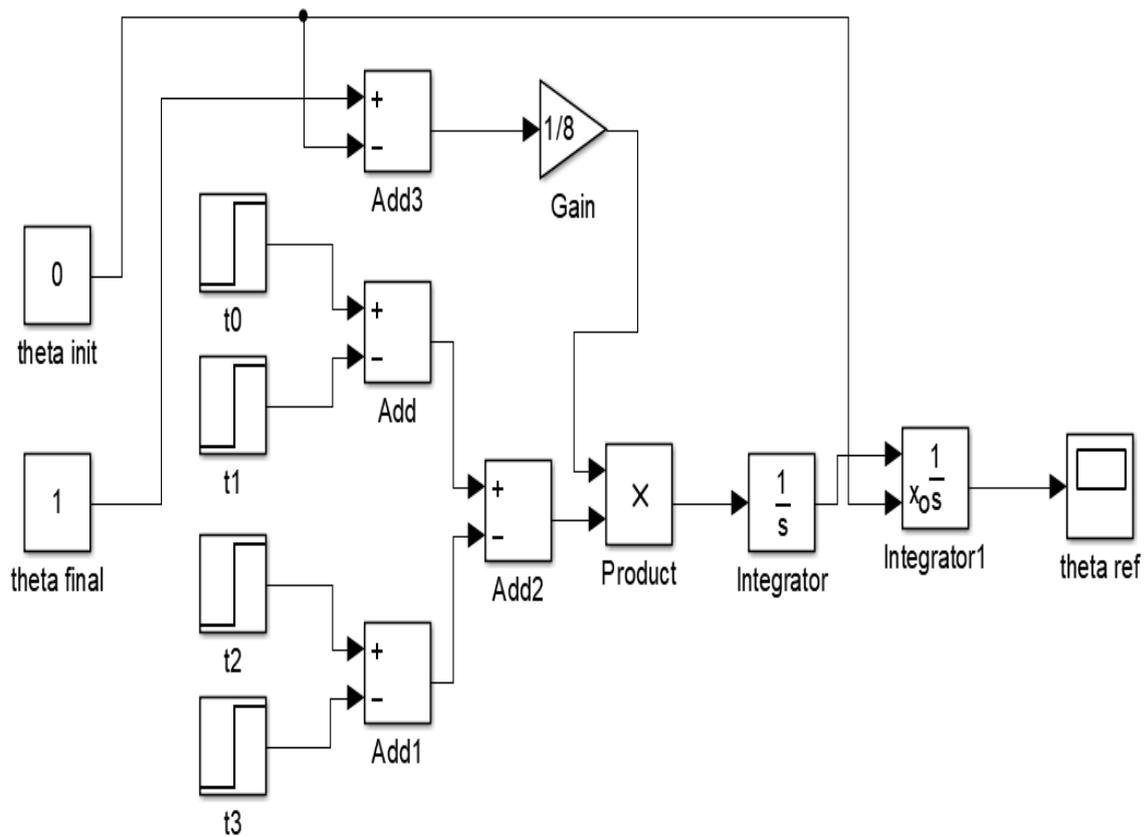


Figure III. 2 Génération de trajectoire.

La trajectoire est présentée ci-dessus :

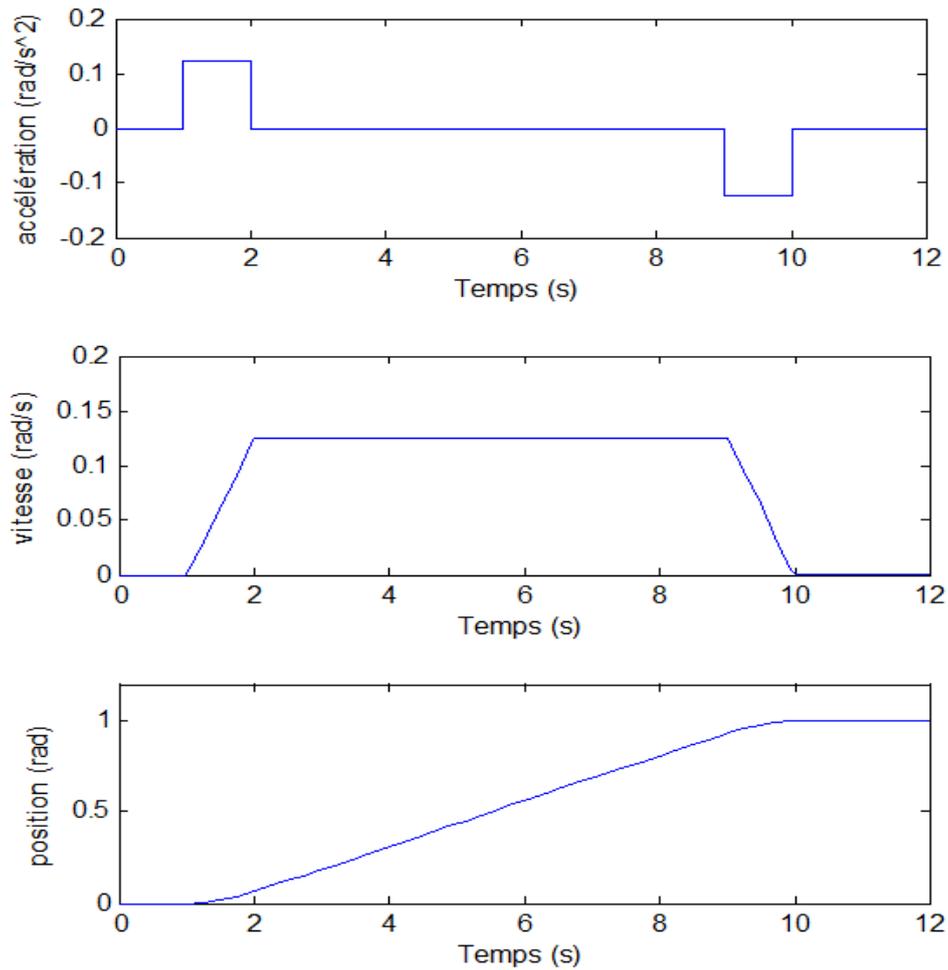


Figure III. 3 Trajectoire utilisée

III.3 Simulation et interprétation des résultats

Afin d'obtenir les résultats souhaités, on simule le système premièrement, avec un régulateur PID simple ensuite avec régulateur PID guidée par les données sans apprentissage, enfin avec régulateur PID guidée par les données avec apprentissage.

III.3.1 Simulation avec régulateur PID simple

Dans cette partie, nous utilisons un régulateur PID, avec des paramètres constants, qui agit pour minimiser l'erreur entre la sortie du système et la sortie souhaitée, sachant que l'entrée est une position de deux points, et on va effectuer dans cette partie 3 essais.

Essai 1 : $\theta_{1C}=1$

$\theta_{2C}=0$

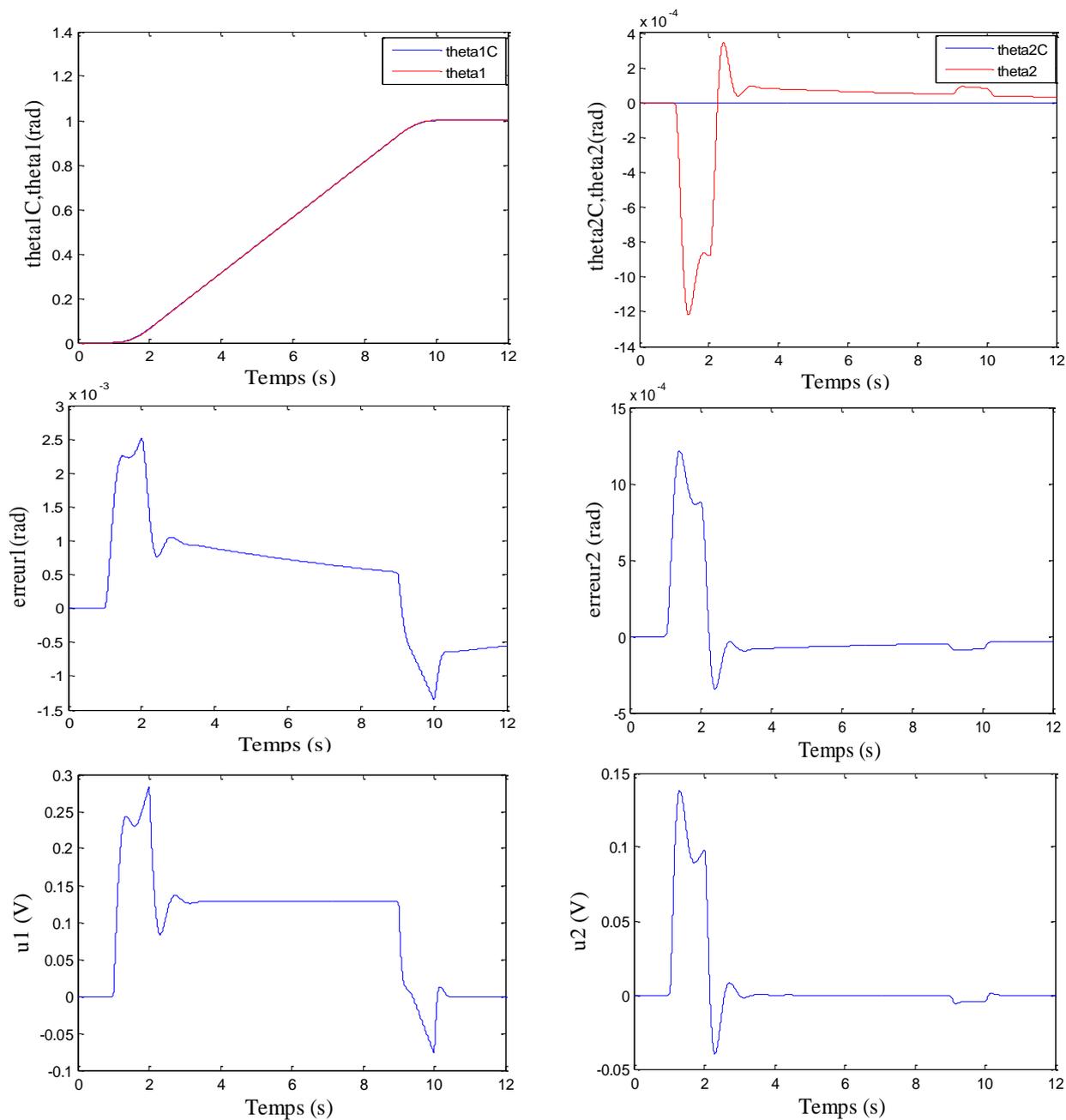


Figure III. 4 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

La figure (III. 4) présente les consignes et les réponses de la trajectoire en fonction de temps ainsi que les erreurs de la poursuite, on remarque que la poursuite est assurée avec des faibles erreurs pour la première trajectoire, mais elle a une faible influence sur la deuxième articulation.

Essai 2 : $\theta_{1C}=0$

$\theta_{2C}=1$

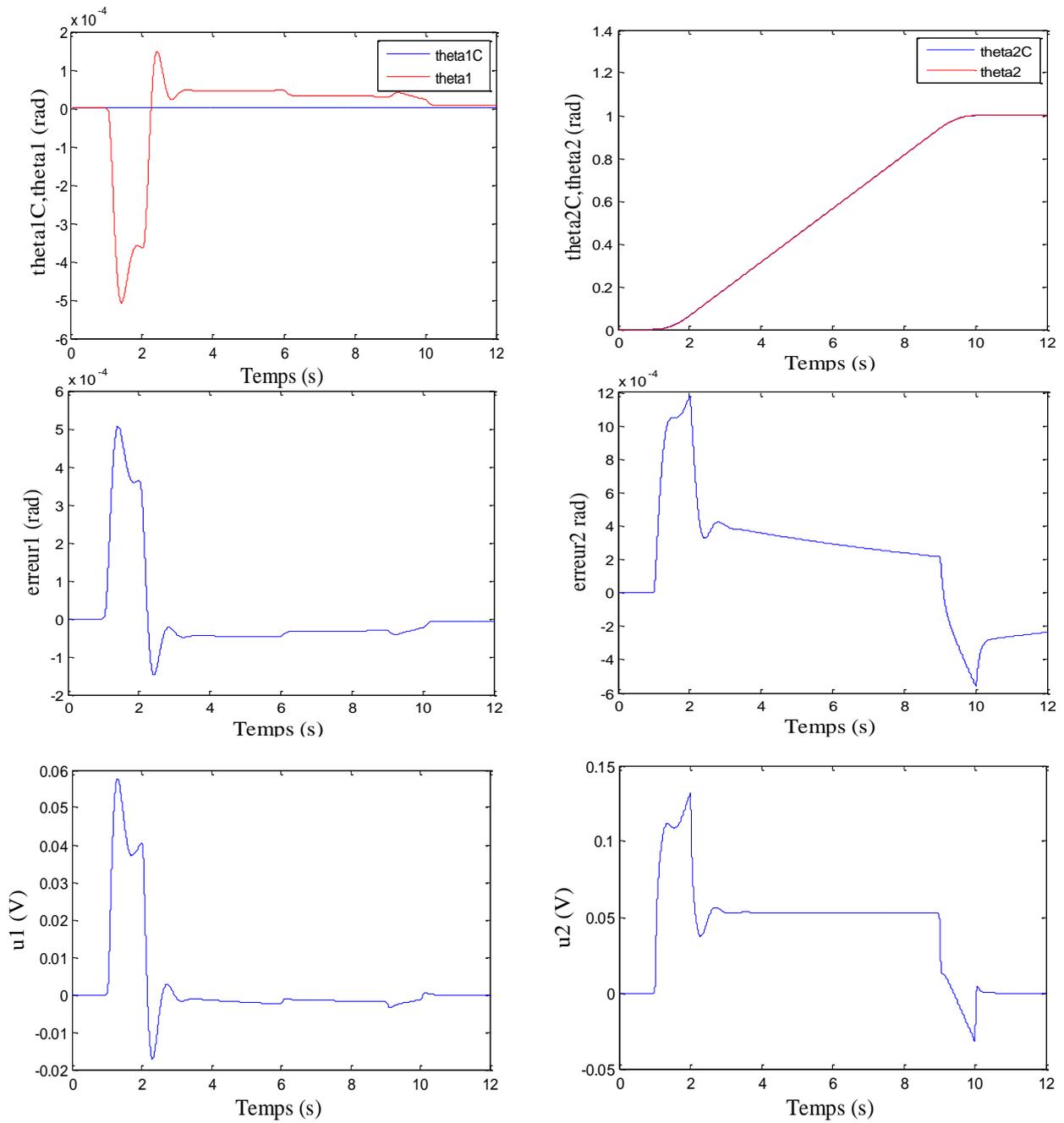


Figure III. 5 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

D'après la figure (III. 5) on remarque que la deuxième trajectoire est assurée avec des faibles erreurs, mais elle a une faible influence sur la première articulation.

Essai 3 : $\theta_{1C}=1$

$\theta_{2C}=1$

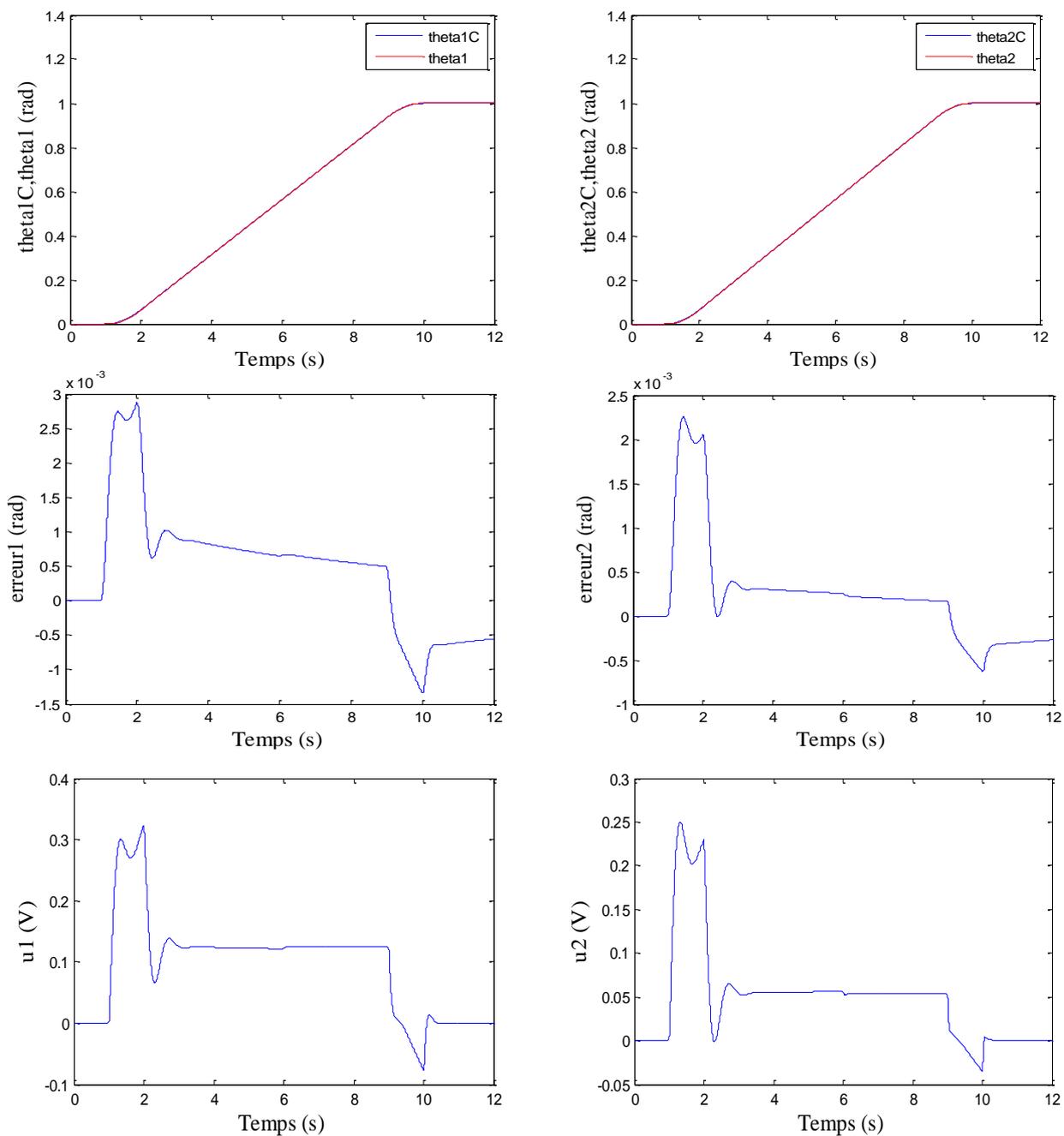


Figure III. 6 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

D'après la figure (III. 6) on remarque que les deux trajectoires suivent leur référence avec des faibles erreurs.

III.3.1.1 Interprétation générale de la première application

D'après les résultats obtenus, on constate que le PID simple assure une bonne poursuite pour les deux trajectoires avec des légères erreurs.

III.3.2 Régulateur PID guidée par les données sans apprentissage

Dans cette partie, nous allons utiliser un régulateur PID avec une base des données sans mise à jour, qui agit pour minimiser l'erreur entre la sortie du système et la sortie souhaitée, sachant que l'entrée est une position de deux points, et on va effectuer dans cette partie 3 essais.

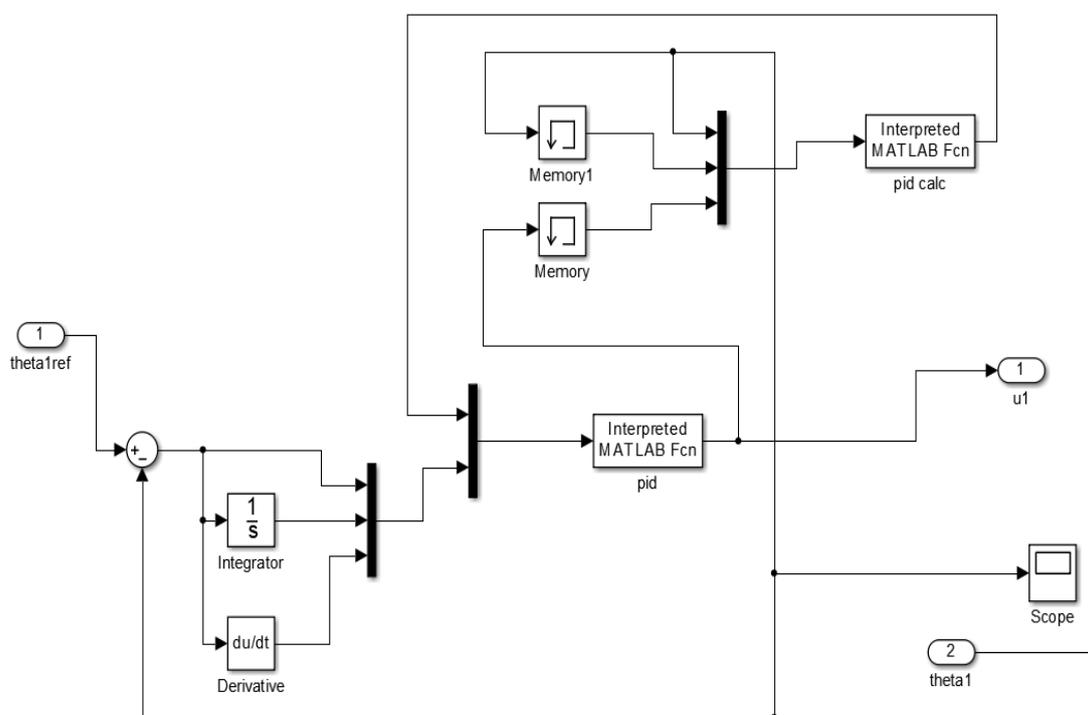


Figure III. 7 Schéma bloc de simulation de régulateur PID basé sur les données sans apprentissage.

La base des données est présentée ci-dessus :

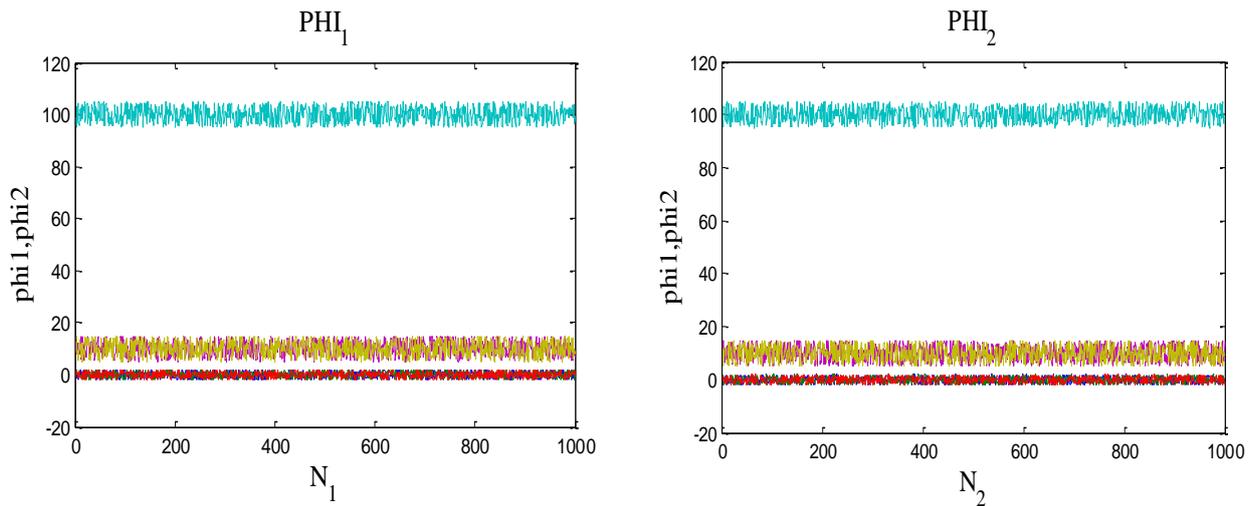


Figure III. 8 Base des données pour le régulateur 1 et le régulateur 2.

La figure (III. 8) présente la base des données pour régulateurs 1 et 2, dans ce cas notre base des données elle ne change pas pendant toute la simulation pour le fait des paramètres de régulateur PID qui sont fixées.

Essai 1 : $\theta_{1C}=1$

$\theta_{2C}=0$

Les résultats de simulation sont donnés comme suite :

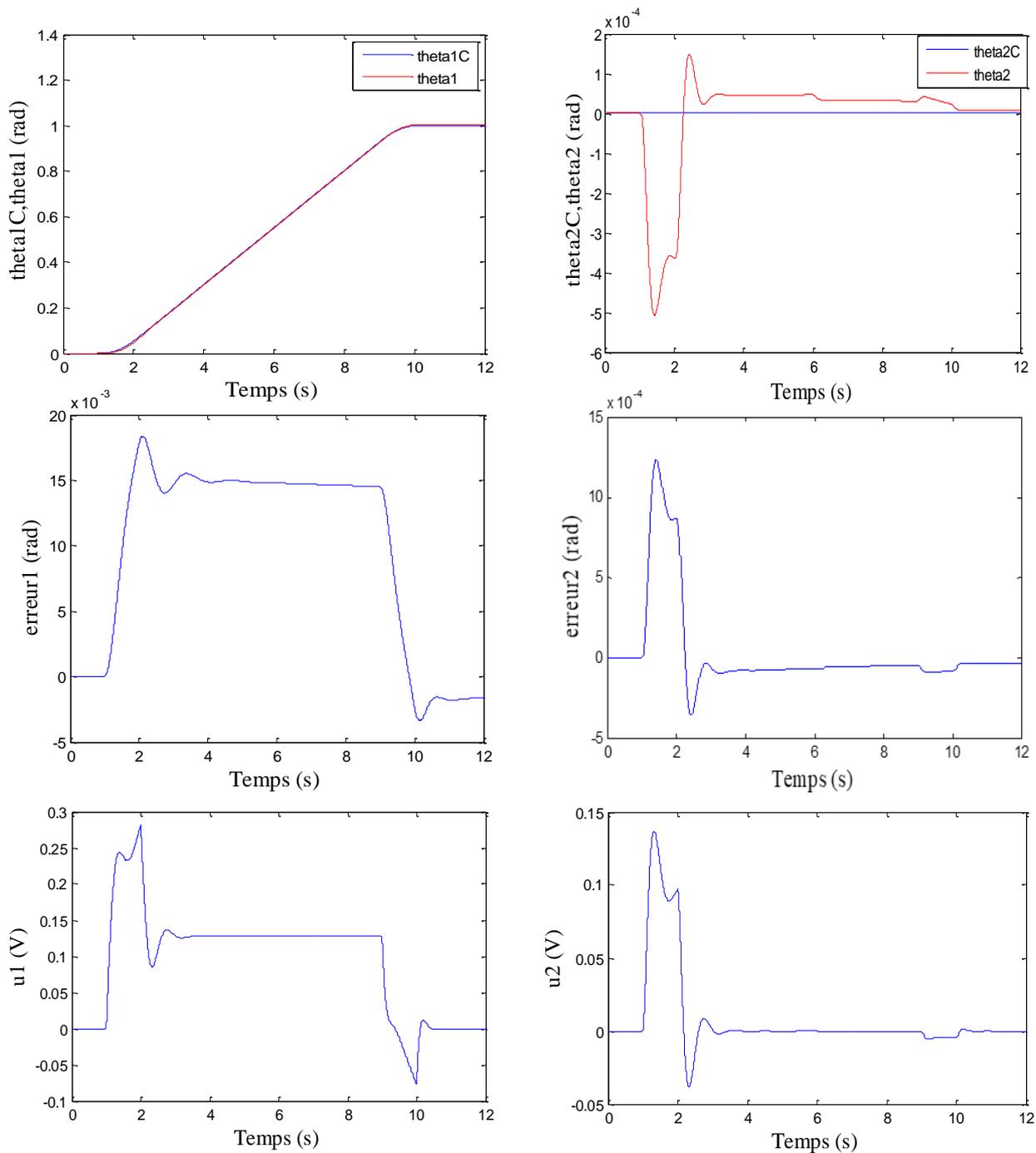


Figure III. 9 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

Essai 2 : $\theta_{1C}=0$

$\theta_{2C}=1$

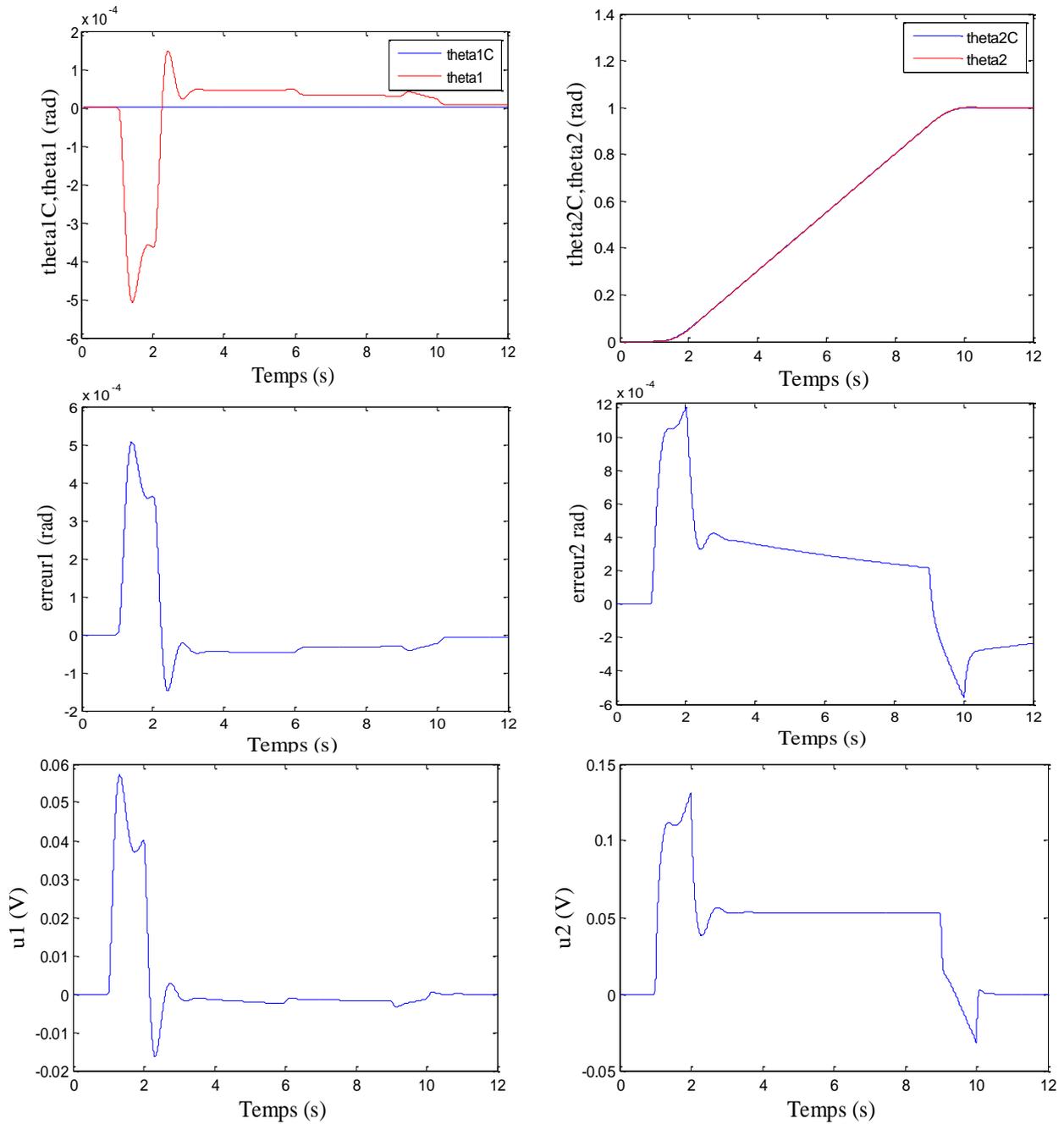


Figure III. 10 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

Essai 3 : $\theta_{1C}=1$

$\theta_{2C}=1$

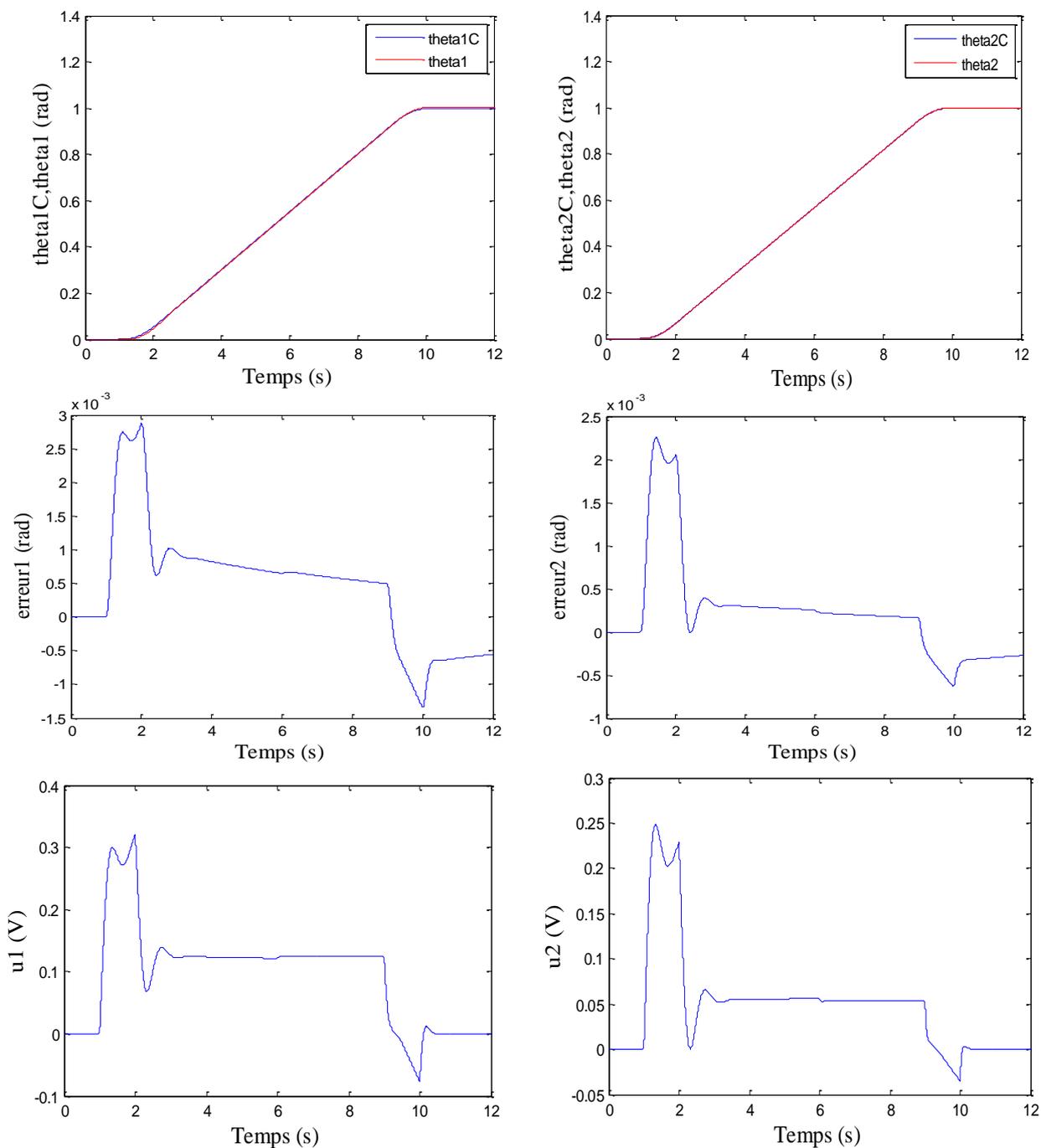


Figure III. 11 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

Essai 1 : $\theta_{1C}=1$

$\theta_{2C}=0$

Les résultats de simulation sont donnés comme suite :

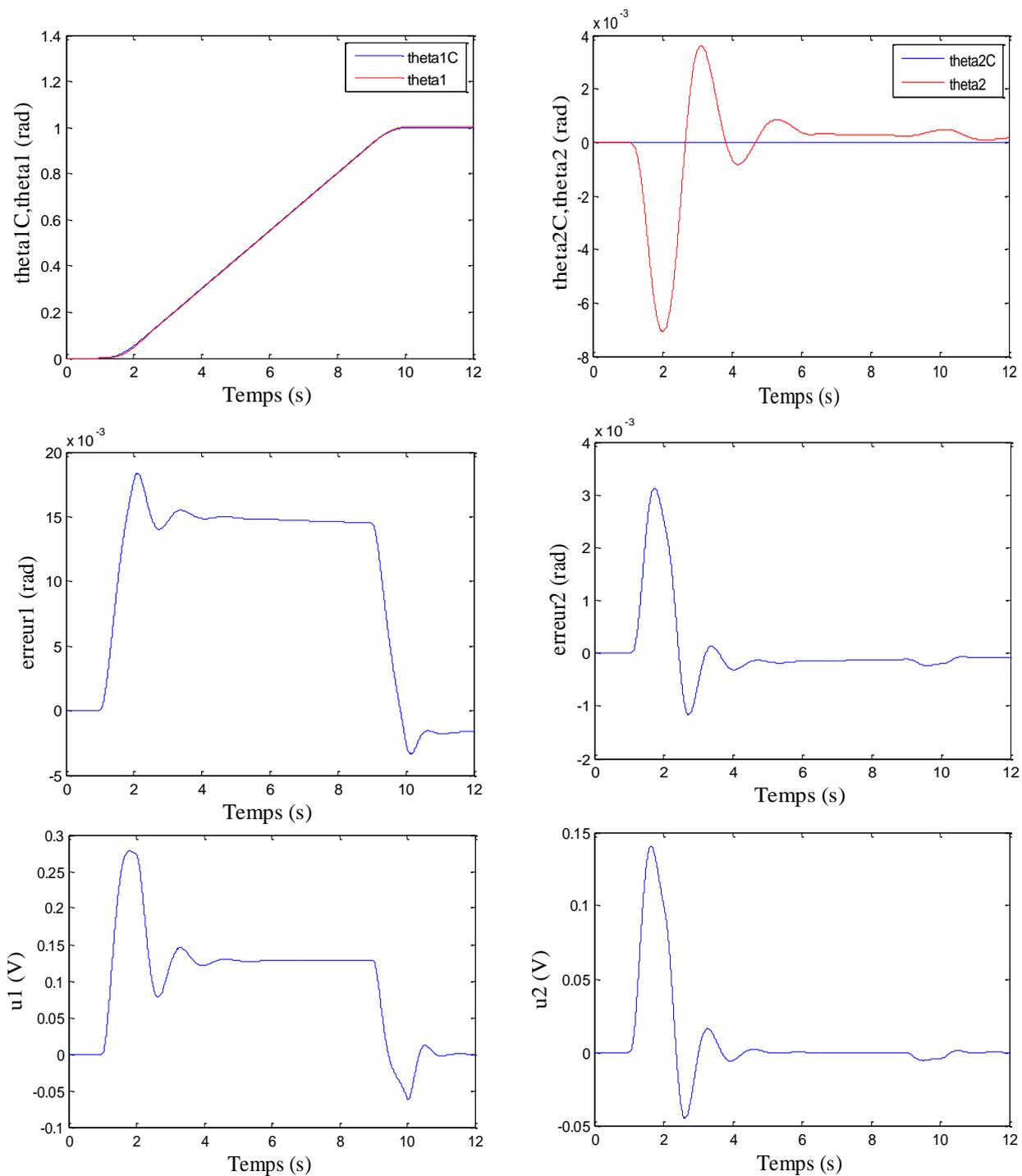


Figure III. 13 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

La figure (III. 13) présente les consignes et les réponses de la trajectoire en fonction de temps ainsi que les erreurs de la poursuite, on remarque que la première trajectoire suit parfaitement sa référence, mais elle a une faible influence sur la deuxième trajectoire.

La base des données est présentée ci-dessus :

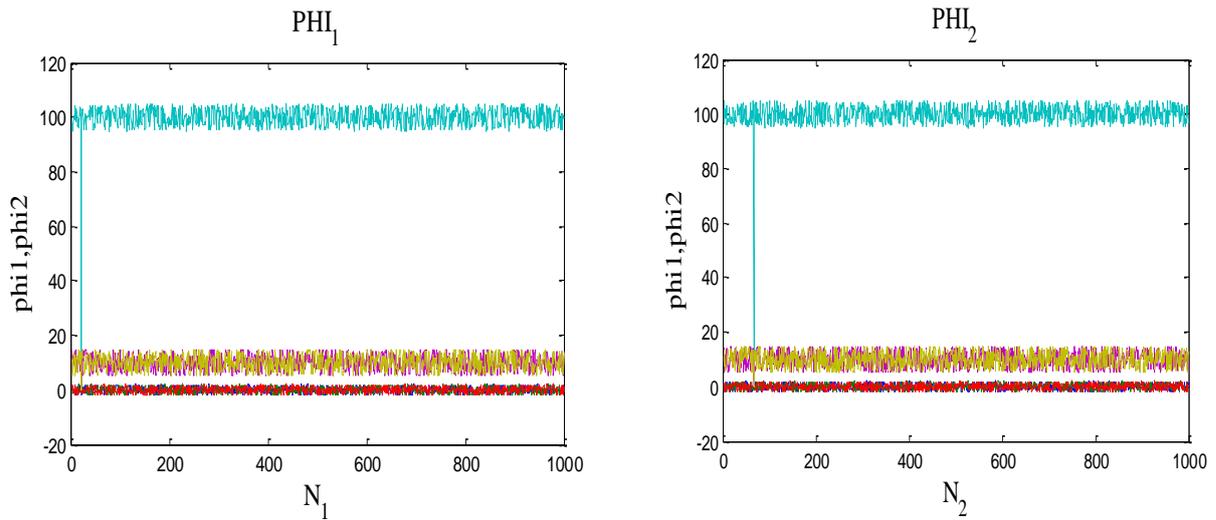


Figure III. 14 Base des données pour le régulateur 1 et le régulateur 2 à l'instant initial.

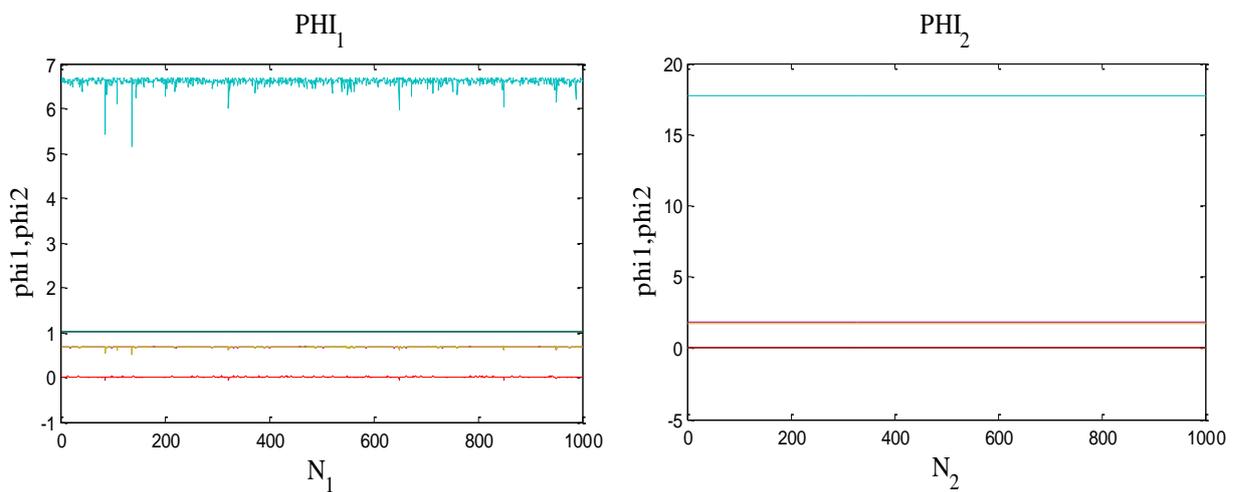


Figure III. 15 Base des données pour le régulateur 1 et le régulateur 2 à l'instant final.

Essai 2 : $\theta_{1C}=0$

$\theta_{2C}=1$

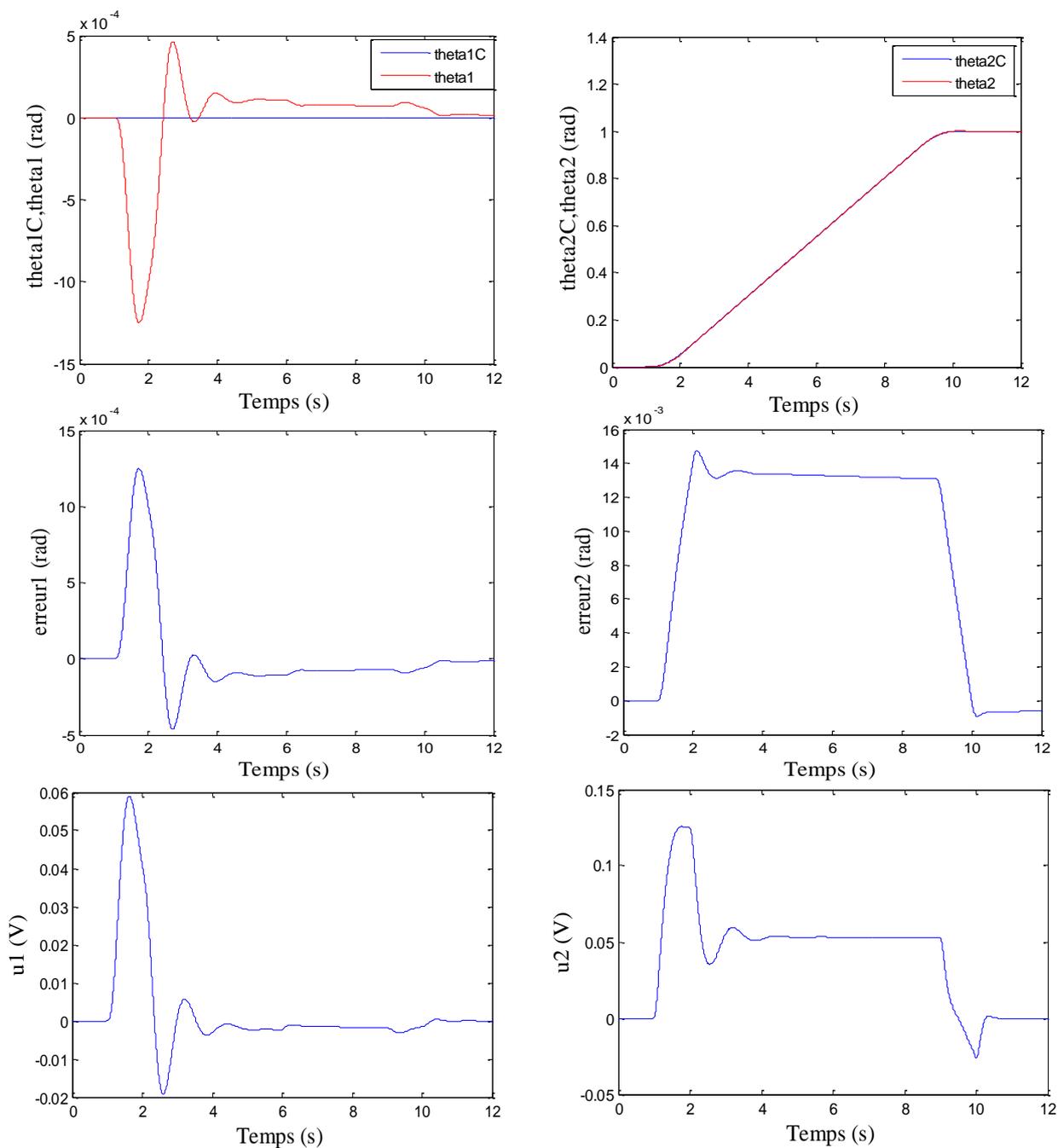


Figure III. 16 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

D'après la figure (III. 16) on remarque que la deuxième trajectoire suit parfaitement sa référence, mais elle a une faible influence sur la première trajectoire.

La base des données est présentée ci-dessus :

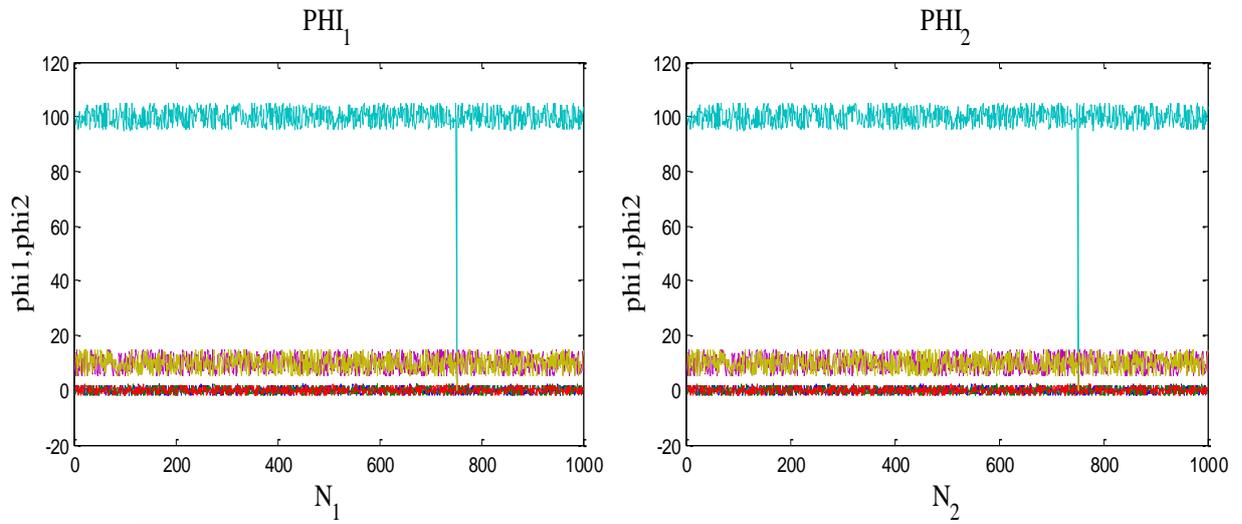


Figure III. 17 Base des données pour le régulateur 1 et le régulateur 2 à l'instant initial.

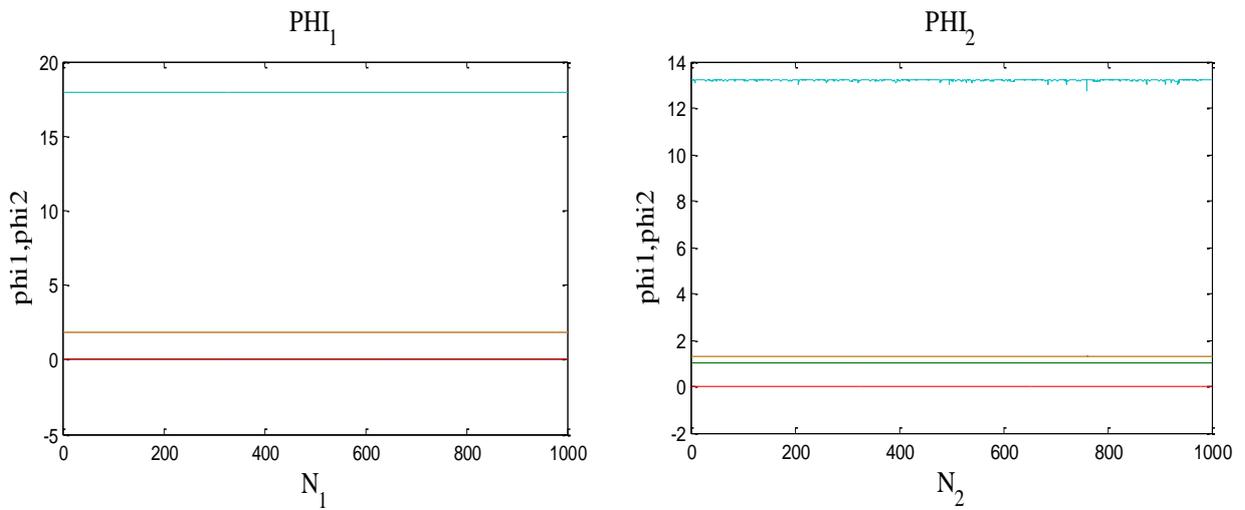


Figure III. 18 Base des données pour le régulateur 1 et le régulateur 2 à l'instant final.

Essai 3 : theta1C=1

theta2C=1

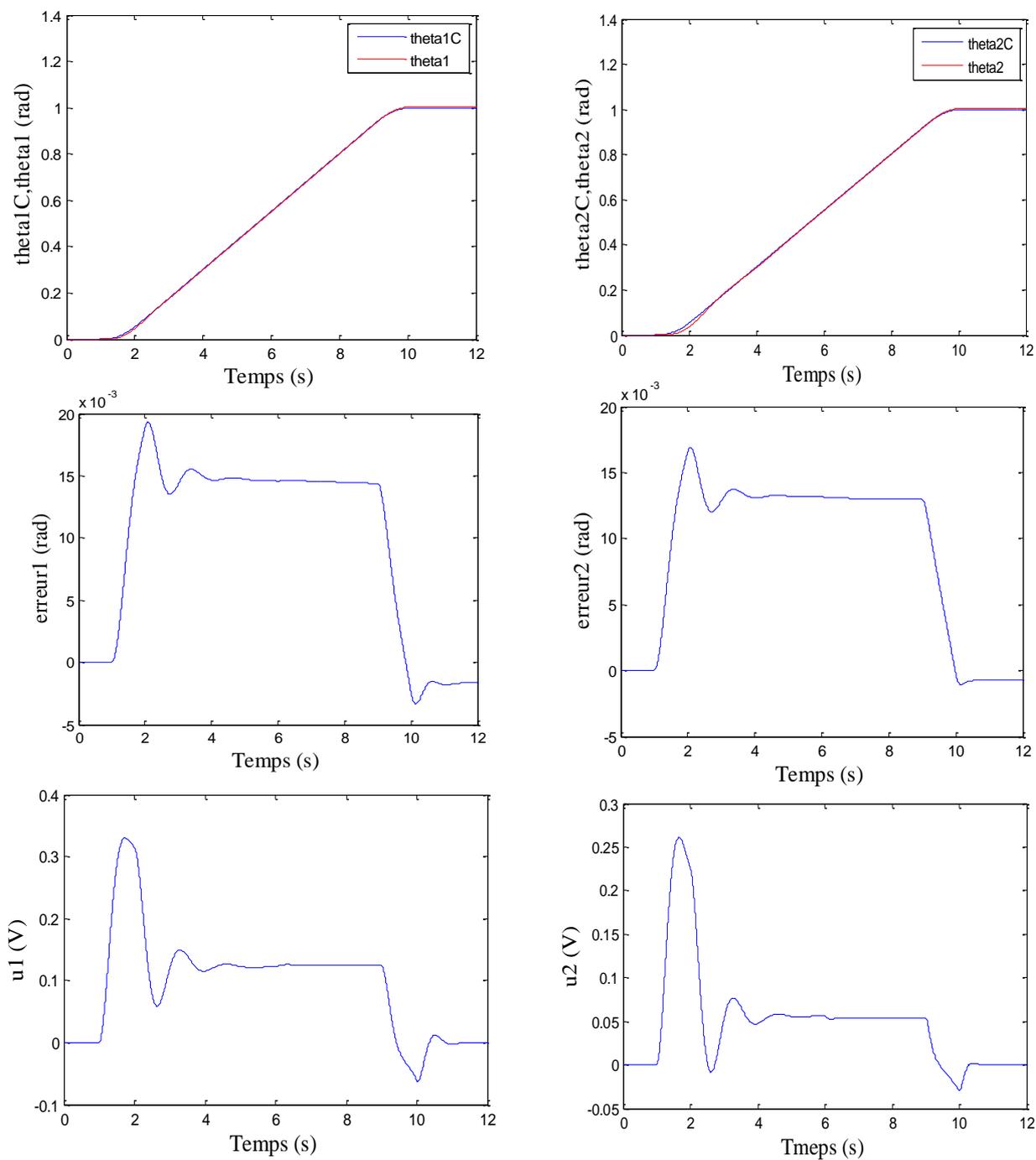


Figure III. 19 Evolution temporelle de la position (consignes et réponses), et les erreurs de poursuite de la trajectoire.

D’après la figure (III.19) on remarque que les deux trajectoires suivent parfaitement leur référence avec des faibles erreurs d’ordre 10^{-3} .

La base des données est présentée ci-dessus :

Figure III. 20 base des données pour le régulateur 1 et le régulateur 2 à l’instant initial.

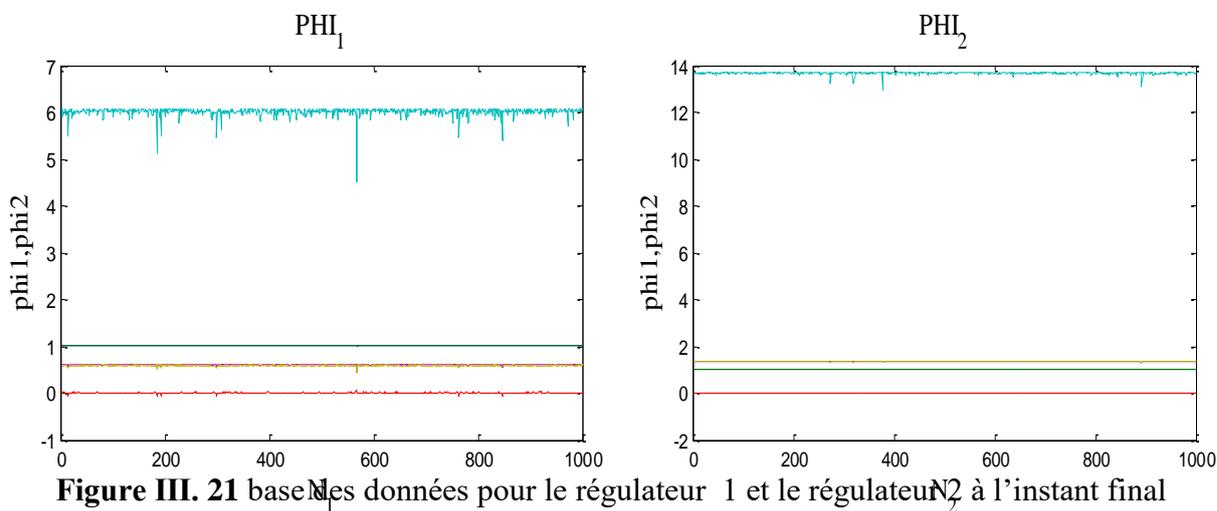
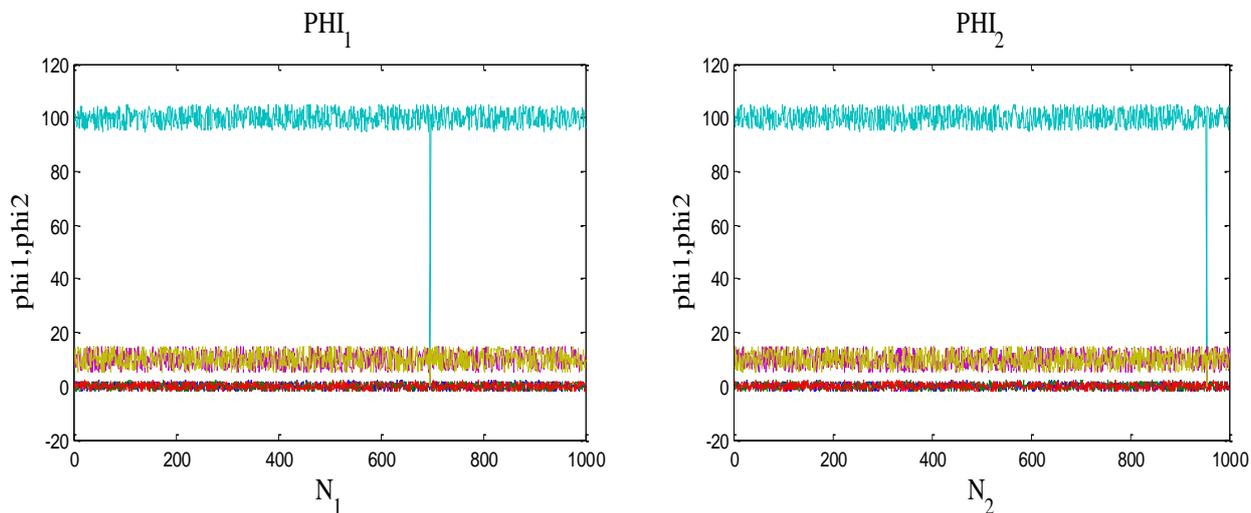


Figure III. 21 base des données pour le régulateur 1 et le régulateur 2 à l’instant final

III.3.3.1 Interprétation générale de la troisième application

D’après les résultats obtenus, constate que le régulateur PID avec base de données avec apprentissage est efficace il assure une bonne poursuite pour les deux trajectoires avec des faibles erreurs.

III.4 Conclusion

Dans ce chapitre, nous avons appliqué la commande guidée par les données à un robot manipulateur. Nous avons présenté le modèle globale du robot et puis nous sommes passées à la simulation. Les résultats obtenus sont satisfaisants en terme de bonne poursuite de référence. Nous concluons que la commande guidée par les données appliquée aux robots manipulateurs est efficace et performante.

Conclusion générale

Conclusion générale

Une commande avancée à un système complexe doit répondre aux exigences en matière de robustesse et performance.

Ce travail porte sur l'application d'une commande avancée qui est une commande guidée par les données dans notre cas sur un système composé d'un robot manipulateur du type SCARA.

Au début, nous avons présenté une généralité sur les robots manipulateurs puis nous nous sommes focalisé sur le robot SCARA et sa modélisation.

Ensuite, nous avons expliqué et présenté les principes de la commande guidée par les données et ses étapes pour la conception des contrôleurs PID piloté par les données.

À la fin, nous avons appliqué la commande guidée par les données, nous avons simulé le système en passant par trois essais et puis nous avons interprété les résultats obtenus.

D'après les résultats obtenus, on trouve que cette commande est très utile, elle donne des résultats satisfaisants, elle répond aux exigences de performances.

Nous proposons comme perspective, d'utiliser ce type de commande guidée par les données sur un robot SCARA réel et d'autres robots manipulateurs, et d'étudier ses différentes variantes.

Références bibliographiques

Références bibliographiques

- [1] **F.KLAOUA**, «Commande prédictive généralisée», mémoire de fin d'étude master en Automatique, Université Mohamed khider Biskra.2014.
- [2] **G. ABBA**, « Commande hybride position/force robuste d'un robot manipulateur utilisé en usinage et/ou en soudage », thèse pour obtenir le grade de docteur, l'École Nationale Supérieure d'Arts et Métiers, 02 décembre 2013.
- [3] Laville Testaud Elsa 3èmeE « L'évolution de la robotique ».
- [4] **P. Fisette, H. Buyse and J.C. Samin**, MECA 2732 : « Introduction à la Robotique», 10 novembre 2004.
- [5] **DJ. GUELLAI**, « Modélisation, contrôle et simulation du robot manipulateur à 2ddl mémoire fin d'étude Master, Université Mohamed Khider Biskra.2018.
- [6] **P.Fisette, H.Buyse, J.C.Samin**. « Introduction à la robotique ». MECA 2732. 2004.
- [7] **A. BENSOUILAH**, «Modélisation et commande par mode glissant et réseau de neurones du robot manipulateur SCARA à 4 ddl», mémoire de fin d'étude master en Automatique, Université Mohamed Sadik Benyahia de Jijel.2020.
- [8] **A. BENSOUILAH**, «Modélisation et commande par mode glissant et réseau de neurones du robot manipulateur SCARA à 4 ddl», mémoire de fin d'étude master en Automatique, Université Mohamed Sadik Benyahia de Jijel.2020.
- [9] **W. KHALI, E. DOMBRE**, «Bases de la modélisation et des robots de la des robots-manipulateurs de type série commande».2012.
- [10] **A. CHEBBAH**, «commande des Robots Manipulateurs», Mémoire de fin d'étude Pour l'obtention du diplôme de Master en Automatique, Université de Jijel.2019.
- [11] **R.Vilanova, A.Visiali**, « PID Control in the third millennium (chapitre 17 page 528) ».

Annexe

Annexe

Paramètres du robot

Masse du premier bras : $m_1=15.91$

Masse du deuxième bras : $m_2=11.361$

Longueur du premier bras : $l_1=0.432$

Longueur du deuxième bras : $l_2=0.432$

Coefficient de frottement visqueux du premier bras : $f_{v1}=0.0085$

Coefficient de frottement visqueux du deuxième bras : $f_{v2}=0.0085$

Coefficient de frottement sec du premier bras : $f_{s1}=0.000$

Coefficient de frottement sec du deuxième bras : $f_{s2}=0.000$

Moment d'inertie du moteur 1 : $im_1=0.0048$

Moment d'inertie du moteur 2 : $im_2=0.0048$

Rapport de réduction de vitesse pour le premier bras : $n_1=120$

Rapport de réduction de vitesse pour le deuxième bras : $n_2=50$