

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE



N° d'ordre :

Série :

Université

Mohammed Seddik

BENYAHIA - Jijel

Faculté des Sciences et de la Technologie

Département : Automatique.

Thèse Présentée

Pour l'obtention du Diplôme de Doctorat 3^{ème} Cycle

Spécialité : Automatique.

Par :

Tarek BENSIDHOUM

Thème :

Contribution à la commande des systèmes répétitifs

Présentée et soutenue le : 08/04/2021.

Devant le Jury composé de :

Mr. Toufik BOUDEN	Professeur	U. M.S.B. Jijel	Président
Mr. Farah BOUAKRIF	Professeur	U. M.S.B. Jijel	Rapporteur
Mr. Michel ZASADZINSKI	Professeur	U/Lorraine, France	Co-Rapporteur
Mr. Naâmane BOUNAR	MCA	U. M.S.B. Jijel	Examineur
Mr. Boubekeur BOUKHEZZAR	Professeur	U. Constantine1	Examineur
Mr. Sofiane GHERBI	Professeur	U. Annaba	Examineur

REMERCIEMENT

Ce travail de thèse a été réalisé au sein de Laboratoire d'Automatique de Jijel (LAJ) à l'université Mohamed Seddik Benyahya, Jijel.

Je tiens à remercier au premier lieu, Dieu, qui m'a donné le courage, la patience et la force de continuer ce travail de thèse, car sans son aide, je n'aurais jamais eu le courage de la finir et ce doctorat n'aurait pas été possible.

*Je tiens à exprimer mes plus vifs remerciements à Mr. **Farah Bouakrif**, professeur à l'université de Jijel, qui fut pour moi un directeur de thèse attentif et très disponible. Ses compétences et sa rigueur scientifique, ses qualités humaines et sa claire voyance m'ont beaucoup apporté et appris.*

*Mes remerciements vont aussi à mon co-directeur de thèse Mr. **Michel Zasadzinski**, professeur à l'université de Lorraine en France, pour sa disponibilité, ses encouragements, sa qualité scientifique et pour m'avoir aidé et bien accueilli au sein du Centre de Recherche en Automatique de Nancy (CRAN-Longwy).*

*Je remercie l'ensemble des membres de jury qui m'ont fait l'honneur de bien vouloir étudier avec attention mon travail : Mr. **Toufik Bouden**, professeur à l'université de Jijel, de m'avoir fait l'honneur d'accepter de présider ce jury, Mr. **Naâmane Bounar**, Maître de Conférences à l'université de Jijel, Mr. **Boubekeur Boukhezzar**, professeur à l'université de Constantine 1 et Mr. **Sofiane Gherbi**, professeur à l'université de Annaba, pour avoir accepté d'être examinateurs de cette thèse.*

*Un grand merci à Mr. **Abdelhamid Tayebi** directeur de laboratoire d'automatique à l'université de Lakehead (Ontario, Canada), pour m'avoir aidé et bien accueilli au sein de son laboratoire. Je lui en suis très reconnaissant.*

Je m'adresse mes sincères remerciements à ma famille : mes chers parents (que Dieu les préserve), mes frères et sœurs, mes neveux, mes nièces et toute la grande famille.

Enfin, pour terminer ces remerciements, une pensée pour mes amis de Mansoura, tous mes collègues de EPST-Annaba, de l'université de Bordj Bou Arreridj et de l'université de Jijel. Sans oublier tous les membres du département électromécanique de Bordj Bou Arreridj et l'équipe de Centre de Recherche en Technologies Industrielles (Chéraga-Bousmail).

Je dédie ce travail à mon père, à ma mère et toute ma famille.

TABLE DES MATIERES

Chapitre. I. Introduction générale.....	1
I.1 Motivations et objectifs de la thèse	1
I.2 Brève histoire.....	2
I.3 Organisation de la thèse.....	3
I.4 Publications associées aux résultats	4
Chapitre. II. Principe de la commande par apprentissage itératif.....	6
II.1 Introduction	6
II.2 Forme générale de la commande par apprentissage itératif	6
II.3 Classifications de la commande par apprentissage itératif	7
II.4 Analyse de la stabilité.....	9
II.4.1 Préliminaires.....	9
II.4.2 Méthode de la norme- λ	11
II.4.3 Méthode de Lyapunov	12
II.4.4 Exemple illustratif 1	13
II.4.5 Exemple illustratif 2	15
II.5 Réseaux de neurones	19
II.5.1 Neurone biologique	19
II.5.1.1 Corps cellulaire.....	20
II.5.1.2 Dendrites.....	20
II.5.1.3 Axone.....	20
II.5.2 Neurone formel (artificiel)	20
II.5.3 Fonction d'activation.....	21
II.5.4 Apprentissage des réseaux de neurones	22
II.5.5 Réseaux de neurones à fonctions de base radiales (RBF).....	22
II.6 Systèmes flous	24

II.7	Conclusion	26
Chapitre. III. Commande par apprentissage itératif d'une classe des systèmes non linéaires multivariables..... 27		
III.1	Introduction	27
III.2	Position du Problème.....	28
III.3	Synthèse d'une loi de commande par apprentissage itératif.....	30
III.4	Résultats de simulation.....	37
III.4.1	Exemple d'un robot manipulateur.....	37
III.4.2	Comparaison avec la commande robuste par apprentissage itératif.....	43
III.5	Conclusion.....	48
Chapitre. IV. Commande par apprentissage itératif des systèmes non linéaires en présence des non linéarités à leurs entrées..... 49		
IV.1	Introduction	49
IV.2	Commande adaptative par apprentissage itératif type-P des robots manipulateurs avec zone-morte à l'entrée.....	51
IV.2.1	Formulation du problème	51
IV.2.1.1	Dynamique des robots manipulateurs	51
IV.2.1.2	Analyse de la zone-morte.....	52
IV.2.2	Synthèse d'une loi de commande adaptative par apprentissage itératif.....	56
IV.2.3	Résultats de simulation.....	64
IV.2.3.1	Résultats de simulation sur le robot manipulateur Puma 560.....	64
IV.2.3.2	Comparaison entre la commande proposée et la CAI de type-D.....	66
IV.3	Commande par apprentissage itératif d'ordre élevé des systèmes non linéaires multivariables avec saturation à l'entrée	73
IV.3.1	Formulation du problème	73
IV.3.2	Synthèse de la loi de commande	74
IV.3.3	Résultats de Simulation.....	78
IV.3.3.1	Exemple 1 : Bras manipulateur.....	78

IV.3.3.2	Exemple 2 : Système MIMO non linéaire	78
IV.4	Conclusion	83
Chapitre. V. Commande par apprentissage itératif à gains variables		84
V.1	Introduction	84
V.2	Position du problème	85
V.3	Synthèse d'une loi de CAI Type -P-.....	86
V.4	Synthèse d'une loi de CAI à gains adaptatifs	90
V.5	Résultats de simulation.....	93
V.5.1	Cas 1 : Utilisation de l'algorithme de PSO	94
V.5.2	Cas 2 : Utilisation de la loi d'adaptation des gains	95
V.6	Conclusion.....	101
Chapitre. VI. Commande par apprentissage itératif des systèmes non linéaires en présence des perturbations non répétitives.....		102
VI.1	Introduction	102
VI.2	Position du problème	103
VI.3	Synthèse de la loi de commande.....	104
VI.4	Résultats de Simulation	108
VI.5	Conclusion.....	112
Conclusion générale		113
Bibliographie.....		116

LISTE DES FIGURES

Figure II.1 Schéma bloc de la commande par apprentissage itératif.....	7
Figure II.2 Model d'un neurone formel.	21
Figure II.3 Différents types de fonctions d'activation, a) fonction à seuil, b) linéaire, c) sigmoïde.	22
Figure II.4 Architecture du réseau RBF.	23
Figure II.5 Structure interne d'un système flou.	25
Figure III.1 Diagramme fonctionnel de la commande proposée.....	30
Figure III.2 Trajectoires réelles et désirée de la première articulation.....	39
Figure III.3 Trajectoires réelles et désirée de la deuxième articulation.	39
Figure III.4 Erreurs de poursuite de la première articulation.	40
Figure III.5 Erreurs de poursuite de la deuxième articulation.....	40
Figure III.6 Norme -Sup de l'erreur de la première articulation au cours des itérations.	41
Figure III.7 Norme sup de l'erreur de la deuxième articulation au cours des itérations.....	41
Figure III.8 Signal de commande pour la première articulation.	42
Figure III.9 Signal de commande pour la deuxième articulation.	42
Figure III.10 Trajectoires réelles et désirée de la première articulation.....	44
Figure III.11 Trajectoires réelles et désirée de la deuxième articulation.	45
Figure III.12 Signal de commande pour la première articulation.	45
Figure III.13 Signal de commande pour la deuxième articulation.	46
Figure III.14 Norme Sup de l'erreur de la première articulation au cours des itérations pour les deux méthodes.....	46
Figure III.15 Norme Sup de l'erreur de la deuxième articulation au cours des itérations pour les deux méthodes.....	46
Figure IV.1 Fonction de la zone-morte.	52
Figure IV.2 Diagramme fonctionnel de la commande proposée.....	56
Figure IV.3 Robot Manipulateur PUMA 560.	66
Figure IV.4 Trajectoires réelles et désirée de la première articulation.	68
Figure IV.5 Erreurs de poursuite de la première articulation.....	68
Figure IV.6 Trajectoires réelles et désirée de la deuxième articulation.	69
Figure IV.7 Erreurs de poursuite de la deuxième articulation.	69
Figure IV.8 Trajectoires réelles et désirée de la troisième articulation.....	70
Figure IV.9 Erreurs de poursuite de la troisième articulation.	70

Figure IV.10 Trajectoires réelles et désirée de la première sortie.....	71
Figure IV.11 Trajectoires réelles et désirée de la deuxième sortie.	71
Figure IV.12 Norme sup de l'erreur du premier joint au cours des itérations pour les deux méthodes.....	72
Figure IV.13 Norme sup de l'erreur du deuxième joint au cours des itérations pour les deux méthodes.....	72
Figure IV.14 Trajectoires réelles et désirée a traves les itérations.....	79
Figure IV.15 Norme Sup de l'erreur au cours des itérations.	80
Figure IV.16 Signal de commande au cours des itérations.	80
Figure IV.17 Norme sup de l'erreur de la première sortie au cours des itérations.	81
Figure IV.18 Norme sup de l'erreur de la deuxième sortie au cours des itérations.	81
Figure IV.19 Premier signal de commande au cours des itérations.	82
Figure IV.20 Deuxième signal de commande au cours des itérations.	82
Figure V.1 Trajectoires réelle et désirée de la première articulation pour la 20 ^{ème} itérations..	96
Figure V.2 Trajectoires réelle et désirée de la deuxième articulation pour la 20 ^{ème} itérations.	96
Figure V.3 Erreurs de poursuite de la première articulation.	97
Figure V.4 Erreurs de poursuite de la deuxième articulation.....	97
Figure V.5 Norme sup des erreurs de la première et la deuxième articulation au cours des itérations.....	98
Figure V.6 Trajectoires réelles et désirée de la première articulation.	98
Figure V.7 Trajectoires réelles et désirée de la deuxième articulation.	99
Figure V.8 Norme sup de l'erreur de la première articulation au cours des itérations.	99
Figure V.9 Norme sup de l'erreur de la première articulation au cours des itérations.	100
Figure V.10 Norme sup de la valeur de Pk au cours des itérations.	100
Figure VI.1 Trajectoires réelles et désirée à travers les itérations pour la première sortie. ...	109
Figure VI.2 Trajectoires réelles et désirée à travers les itérations pour la deuxième sortie. ...	110
Figure VI.3 Erreurs de poursuite à travers les itérations pour la première sortie.....	110
Figure VI.4 Erreurs de poursuite à travers les itérations pour la deuxième sortie	111
Figure VI.5 Norme Sup de l'erreur de la première sortie au cours des itérations.....	111
Figure VI.6 Norme Sup de l'erreur de la deuxième sortie au cours des itérations	112

LISTE DES TABLEAUX

Table III.1 Valeurs de $\sup \ e_1\ $ pour les deux méthodes (1 ^{ère} articulation).	47
Table III.2 Valeurs de $\sup \ e_2\ $ pour les deux méthodes (2 ^{ème} articulation).....	47
Table V.1 Valeurs de $\sup (\ e_1\ + \ e_2\)$ pour des différents gains.....	98

SYMBOLES ET NOTATIONS

\mathbb{R}, \mathbb{N}	L'ensemble des nombres réels (resp. naturels).
y_d	Sortie désirée du système.
x_d	Etat désiré du système.
k	Nombre des itérations.
y_k	Sortie réelle du système à la $k^{\text{ième}}$ itération.
x_k	Etat réelle du système à la $k^{\text{ième}}$ itération.
d_k	Perturbations externes à la $k^{\text{ième}}$ itération.
e_k	Erreur de poursuite à la $k^{\text{ième}}$ itération.
V_k	Fonction de Lyapunov à la $k^{\text{ième}}$ itération.
λ	Scalaire positif (largement grand).
$\ *\ $	Norme Euclidienne de (*).
$\ x\ _{L_2}$	Norme L_2 de (*).
$\ x\ _\lambda$	Norme λ de (*).
q_d	Vecteur de position désirée des robots manipulateurs.
\dot{q}_d	Vecteur de vitesse désirée des robots manipulateurs.
q_k	Vecteur de position articulaire des robots manipulateurs à la $k^{\text{ième}}$ itération.
\dot{q}_k	Vecteur de vitesse articulaire des robots manipulateurs à la $k^{\text{ième}}$ itération.
$\lambda_{\min}(*)$	Valeur propre minimale de (*).
$\lambda_{\max}(*)$	Valeur propre maximale de (*).
$Sat(*)$	Saturation de (*).
$DZ(*)$	Zone morte de (*).

ABREVIATION

CAI	Commande par apprentissage itératif.
SISO	Mono entrée Mono sortie.
MIMO	Multi entrées Multi sorties.
RBF	Réseaux de neurones à base de fonction radiale.
GLC	Condition globale de Lipschitz.
LLC	Condition locale de Lipschitz.
PSO	L'optimisation par essaim particulière.

Chapitre. I.

Introduction générale

1.1 Motivations et objectifs de la thèse

Beaucoup de systèmes existant dans le monde industriel sont répétitifs. On appelle système répétitif un système qui exécute une tâche donnée d'une manière répétitive dans un intervalle de temps fixe. Les robots manipulateurs et les machines industrielles (les véhicules convoyeurs, presses automatiques...) programmés pour fonctionner de manière répétitive constituent des exemples typiques de ce genre de systèmes. La commande de tels systèmes a suscité énormément d'intérêt, et beaucoup de travaux ont été publiés dans les grands journaux et les congrès importants de la discipline Automatique. En effet, lorsqu'une machine exécute la même tâche à plusieurs reprises (cycles), il devient intéressant d'utiliser les informations des cycles précédents en vue d'améliorer les performances du système à la prochaine exécution de cette tâche. Ceci caractérise l'approche de Commande par Apprentissage Itératif (CAI).

Il est à noter que nous avons constaté plusieurs problématiques qui existent lors de la conception de la commande des systèmes non linéaire répétitifs. Ceci nous a motivé d'orienter notre investigation dans ce sens, c.à.d. de développer des nouvelles lois de commande par apprentissage itératifs pour résoudre ces problèmes. En effet, nous avons développé plusieurs schémas de commande par apprentissage itératif afin de résoudre le problème de poursuite des systèmes non linéaires multivariables. Nous avons pu proposer des solutions aux quelques contraintes rencontrées lors de la synthèse de cette technique de commande sur les systèmes non linéaires multivariables. Parmi ces contraintes, on trouve la nécessité, pour les schémas classiques de la CAI, de satisfaire la condition de continuité globale de Lypschitz. Dans ce sens, nous avons surmonté cet handicap en proposant un schéma de CAI sans utiliser cette hypothèse. Un autre problème rencontré est la nécessité de connaître le signe de la matrice de commande. Dans cette thèse, on trouve la synthèse d'une loi de CAI sans supposer la connaissance de ce signe. De plus, dans la pratique, l'existence des non-linéarités de type zone-morte et saturation

sur les entrées des systèmes à commander peuvent déstabiliser et détruire les performances des systèmes. Dans le but de donner des solutions à ces problèmes, nous avons développé deux lois de commande CAI en tenant compte ces deux contraintes. La première est appliquée sur un robot manipulateur avec la contrainte de la zone-morte à l'entrée, alors que la deuxième est appliquée sur un système non linéaire avec la contrainte de saturation en entrée. Il y a, également, d'autres contributions dans cette thèse, telles que, la proposition des schémas de commande à gain variable et la synthèse de loi de commande avec un estimateur des perturbations. En fait, il est connu que l'utilisation de l'action dérivative dans les applications pratiques peut générer des bruits, et à cause de la nature itérative du système, ces bruits s'accumulent à travers les itérations ce qui pose un problème. Afin de surmonter ce problème, nous avons proposé des commandes de type P, sans utiliser l'action dérivative. Pour toutes les commandes proposées dans cette thèse, la stabilité de système en boucle fermée est démontrée en utilisant une des deux méthodes, la méthode de la norme- λ et la méthode de Lyapunov.

1.2 Brève histoire

Bien que l'idée de base de la commande par apprentissage itératif a été breveté pour la première fois par Garden en 1971, ce travail a été déjà réalisé en 1967 aux Etats-Unis, intitulé « Commande par apprentissage des actionneurs dans un système de commande ». L'approche consiste à stocker un signal de commande dans la mémoire de l'ordinateur et actualiser itérativement le signal de commande en utilisant l'erreur entre la réponse réelle et la réponse désirée. En 1978, une première publication académique sur CAI a été introduite au Japon par Uchiyama [UCH 78]. Malheureusement, cet article a été publié en Japonais et le travail n'a pas connu de continuité. Et ce n'est qu'à partir de 1984 et après les travaux de [ARI 84] que la commande par apprentissage itératif avait été donnée d'une manière explicite. Après et à base de ce papier, la commande est devenue la plus populaire pour la commande des systèmes répétitif et beaucoup des travaux ont été publiés sur la CAI, nous citons entre autres [BRI 06], [AHN 07], [WAN 09], [BOU 08] et [SEB 19].

Il est à noter que les robots industriels représentent les systèmes les plus adéquats pour l'implémentation pratique de cette commande. En effet, le problème principal pour la commande des robots manipulateurs c'est comment forcer les sorties réelles des articulations à suivre plus proche possible les sorties désirées, l'objectif principal de la commande est de traiter et résoudre ce problème sous différentes contraintes. Pour voir l'intérêt de la commande par apprentissage itératif dans le domaine industriel, plusieurs applications ont été faites sur

différents systèmes robotiques. Des applications pratiques ont été réalisées sur un robot manipulateur à deux degrés de liberté par [WAN 18], sur un robot de type ABB IRBI1400 par [NOR 02], sur un robot à cinq degrés de liberté de type CRS 465 par [TAY 08], sur un robot manipulateur à six degrés de liberté de type CRS255 (CATALYST5) par [TAY 06] et sur un robot mobile de type ALIVE3 par [KAN 05]. Par ailleurs, plusieurs implémentations pratiques de la CAI sur d'autres systèmes industriels ont été faites, par exemple, l'application de cette technique sur le Microscopie à force atomique de système piézoélectrique par [LEA 061], pour la Machine CNC (commande numérique par ordinateur) par [KIM 96], pour contrôler la température d'un four de thermoformage [DER 00], pour commander les arrêts du train [HOU 11] et pour le lecteur de disque industrielle [CHE 06] et sur l'exothermique pour Les réacteurs chimique semi-discontinus [MEZ 02].

D'autre part, pour atteindre les performances de système répétitif tel que la stabilité et la convergence, et pour maintenir la robustesse de système, la CAI a été combinée avec d'autres techniques de commande afin d'améliorer ces performances à travers les itérations. On note encore qu'il y a plusieurs publications dans ce sens, par exemple, la commande par apprentissage itératif avec la commande à structure variable (Xu 1998), avec la commande adaptative [TAY 04], avec la commande par backstepping [WAN 13], à base de la logique floue [CHI 08] et à base des réseaux de neurones artificiels [CHI 02].

En général, on distingue deux méthodes différentes pour l'analyse de la convergence des systèmes répétitifs. La première méthode c'est lambda norme, qui est basée sur l'utilisation des propriétés des normes du système à commander [BIE 89]. La deuxième méthode c'est la fonction de Lyapunov, qui est basée sur la théorie développée par le mathématicien russe Alexandre Mikhailovich Lyapunov [QU 02].

1.3 Organisation de la thèse

Le manuscrit est divisé en deux parties :

La première partie porte sur les notions élémentaires et fondamentales sur la commande des systèmes répétitifs. Elle est composée de deux chapitres :

- Dans le premier chapitre, une introduction décrit brièvement les motivations et objectifs de la thèse, ainsi qu'une brève histoire sur la commande par apprentissage itératif.

- Le deuxième chapitre présente le principe de base de la commande par apprentissage itératif, différentes classifications de cette approche et les méthodes utilisées pour l'analyse de la stabilité.

La deuxième partie de ce manuscrit concerne aux développements des nouveaux schémas de commande par apprentissage itératif des systèmes non linéaires multi-variables, ce qui présente notre contribution dans ce domaine de recherche. Elle est composée de quatre chapitres :

- Le troisième chapitre aborde la commande par apprentissage itératif à base des réseaux de neurones à fonctions de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des systèmes non linéaires multivariés répétitifs sans la connaissance du signe de la matrice de commande.
- Le quatrième chapitre présente la synthèse de deux approches de commandes par apprentissage itératif. Dans le premier schéma, nous avons présenté une loi de commande adaptative par apprentissage itératif de type -P- à base des réseaux de neurones à fonction de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des robot manipulateurs possédants des non-linéarités de type zone-morte à leurs entrées et soumis à des perturbations externes non répétitives. Dans la deuxième approche, une loi de commande par apprentissage itératif de Type -P- d'ordre élevé est proposée pour traiter le problème de poursuite des systèmes non linéaires multivariés MIMO en présence des saturations à leurs entrées.
- Dans le cinquième chapitre, une loi de commande par apprentissage itératif à gains variables est synthétisée. Le choix des gains est fait par la technique de l'optimisation PSO.
- Le sixième chapitre est consacré à la synthèse de commande par apprentissage itératif avec un estimateur des perturbations est proposé pour une classe des systèmes non linéaires soumis à des perturbations externes non répétitives et inconnues.

Enfin, nous terminons la thèse par une conclusion générale et des perspectives pour la continuité de ce travail.

1.4 Publications associées aux résultats

Articles publiés

- Bensidhoum, T., Bouakrif, F. et Zasadzinski, M. (2019). Iterative learning radial basis function neural networks control for unknown multi input multi output nonlinear

systems with unknown control direction. *Transactions of the Institute of Measurement and Control*, 41(12), 3452-3467.

- Bensidhoum, T., et Bouakrif, F. (2020). Adaptive P-type iterative learning radial basis function control for robot manipulators with unknown varying disturbances and unknown input dead zone. *International Journal of Robust and Nonlinear Control*, vol. 30, no. 10, pp.4075-4094.

Communications dans le cadre des manifestations scientifiques

- Bensidhoum, T., Bouakrif, F. et Zasadzinski, M. (2019). Iterative learning fuzzy control with optimal gains for a class of nonlinear systems. In : *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2019. p. 210-215. Paris, France.
- Bensidhoum, T., Bouakrif, F. et Zasadzinski, M. (2019). Iterative learning neural networks control with disturbances estimation for nonlinear systems. In : *2019 4th World Conference on Complex Systems (WCCS)*. IEEE, 2019. p. 1-5. Ouarzazate, Morocco.
- Bouakrif, F., Bensidhoum, T., et Zasadzinski, M. (2020, June). Trajectory tracking controller for nonlinear systems with disturbances using iterative learning algorithm without resetting condition. In : *2020 7th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, Vol. 1, pp. 517-522. Prague, Czech Republic.

Chapitre. II.

Principe de la commande par apprentissage itératif

II.1 Introduction

Dans le domaine industriel, on trouve souvent des systèmes non linéaires qui exécutent des tâches répétitives. L'utilisation d'un régulateur conventionnel pour commander ce type de systèmes fournira la même erreur à chaque itération. D'où la nécessité de concevoir un régulateur capable d'utiliser les informations (commande, erreurs...) des itérations antérieures afin d'exécuter progressivement mieux le nouveau cycle jusqu'il arrive à une erreur nulle. Ceci le principe de la commande par apprentissage itératif.

Dans ce chapitre, nous donnons la philosophie sur laquelle s'appuie la CAI, en mettant en évidence le principe de base de cette technique, les différents types de la CAI ainsi que les deux méthodes utilisées dans l'analyse de stabilité lors de la conception de cette commande, à savoir, la méthode de la norme-Lambda et la méthode de Lyapunov.

II.2 Forme générale de la commande par apprentissage itératif

Au cours des vingt dernières années, plusieurs types d'algorithme de la commande par apprentissage itératif ont été développés. La forme générale de cette commande est donnée par :

$$u_{k+1}(t) = f(u_k(t), \dots, u_{k-m}(t), e_{k+1}(t) \dots, e_{k-m}(t)), m \geq 1. \quad (\text{II.1})$$

Avec t présente le temps, k désigne l'indice des itérations, $u_k(t)$ désigne la commande à l'itération k et $e_k(t)$ désigne l'erreur à l'itération k . f est une fonction qui peut dépendre à la fois:

- Des commandes utilisées dans les itérations précédentes.
- Des erreurs obtenues durant les différentes itérations, et/ou ses dérivées et/ou ses intégrales.

Le but principal est de déterminer une fonction simple qui assure la convergence de l'erreur à travers les itérations. Le schéma bloc traduisant cette approche est donné dans la figure II.1.

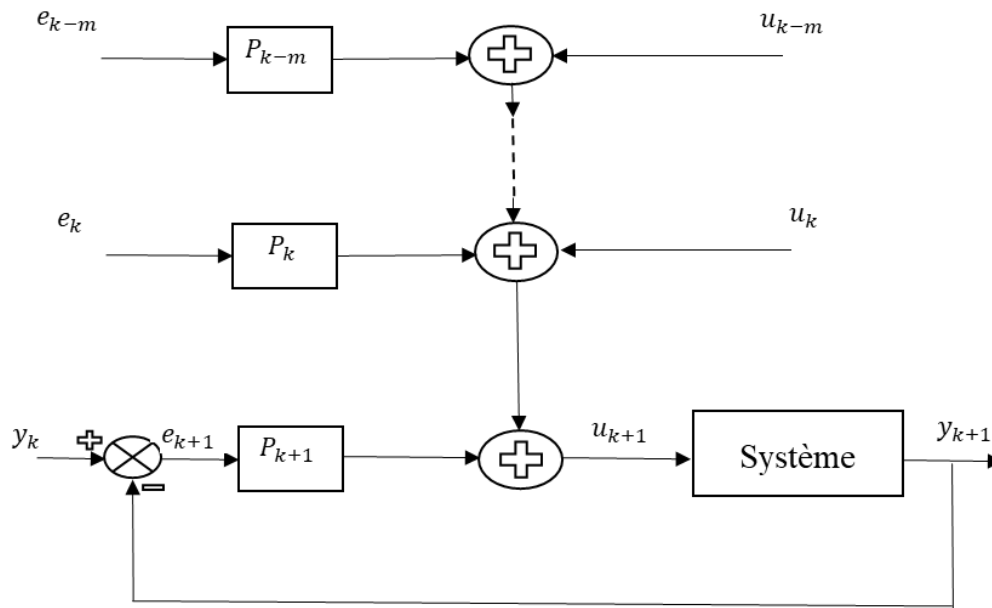


Figure II.1 Schéma bloc de la commande par apprentissage itératif.

II.3 Classifications de la commande par apprentissage itératif

Il est à noter que suivant les informations utilisées dans la loi de commande par apprentissage itératif proposée, la CAI peut être classée en deux types :

- CAI off-line : les informations utilisées sont obtenues à partir des itérations précédentes. Ce type de commande est en boucle ouverte vis-à-vis aux itérations, la loi de commande de ce type se présente sous la forme :

$$u_{k+1}(t) = u_k(t) + \dots + u_{k-m}(t) + P_k e_k(t) \dots + P_{k-m} e_{k-m}(t), m \geq 1. \quad (\text{II.2})$$

- CAI on-line : les informations utilisées sont obtenues à partir des itérations précédentes et de l'itération courante en même temps. Ce type de commande est en boucle fermée vis-à-vis aux itérations, la loi de commande de ce type se présente sous la forme :

$$u_{k+1}(t) = u_k(t) + \dots + u_{k-m}(t) + P_{k+1} e_{k+1}(t) + P_k e_k(t) \dots + P_{k-m} e_{k-m}(t), m \geq 1. \quad (\text{II.3})$$

On note encore que suivant le nombre des informations utilisées dans les lois de commande par apprentissage itératif proposées, on peut distinguer plusieurs catégories de la CAI tel que :

- CAI d'ordre un : les informations utilisées sont provenues à partir d'une seule itération.
- CAI d'ordre élevé : les informations utilisées sont provenues à partir de plusieurs itérations. Si le nombre des itérations utilisé c'est n , on dit CAI d'ordre n .

En effet, une comparaison a été faite entre les deux types de commande dans [CHI 18], basant sur les résultats de simulation, il a été montré que l'utilisation de la commande d'ordre élevé est meilleure en termes de performances de convergence de système par rapport à l'utilisation de la commande d'ordre un.

En général, la commande par apprentissage itératif est décrite par une lettre suivant le type de l'erreur utilisée dans la loi de commande. Soit l'action proportionnelle "P", l'action dérivée "D" et l'action intégrale "I". Pour cela, on peut distinguer quatre types de la CAI :

- P-type CAI : la loi de commande contient juste l'erreur. L'équation la plus simple de ce type est :

$$u_{k+1}(t) = u_k(t) + P_k e_k(t) \quad (\text{II.4})$$

- D-type CAI : la loi de commande contient juste la dérivée de l'erreur. L'équation la plus simple de ce type est :

$$u_{k+1}(t) = u_k(t) + D_k \dot{e}_k(t) \quad (\text{II.5})$$

- PD-type CAI : la loi de commande contient l'erreur et sa dérivée. L'équation la plus simple de ce type est :

$$u_{k+1}(t) = u_k(t) + P_k e_k(t) + D_k \dot{e}_k(t) \quad (\text{II.6})$$

- PID-type CAI : la loi de commande contient l'erreur, son intégral et sa dérivée. L'équation la plus simple de ce type est :

$$u_{k+1}(t) = u_k(t) + P_k e_k(t) + D_k \dot{e}_k(t) + I_k \int e_k(t) d\tau \quad (\text{II.7})$$

En outre, il est important de combiner les actions de l'erreur dans la loi de commande proposée afin d'obtenir les avantages et les caractéristiques de chaque action. Cependant, le choix des gains de commande reste toujours difficile.

II.4 Analyse de la stabilité

II.4.1 Préliminaires

Les notations mathématiques préliminaires et les lemmes nécessaires sont établis dans cette section pour l'utiliser dans le reste de la thèse.

Les notations \mathbb{R} et \mathbb{N} présentent l'ensemble des nombres réels et naturels, respectivement.

$x \in \mathbb{R}^n$ est un vecteur de dimension n . $A \in \mathbb{R}^{n \times n}$ est une matrice de dimension $(n \times n)$.

La matrice A est dit définie positive (semi définie positive) si

$$A > 0 \quad (A \geq 0) \quad (\text{II.8})$$

$\lambda_{\min}(A)$ et $\lambda_{\max}(A)$ représentent la valeur propre minimale et maximale de la matrice A , respectivement.

Si la matrice A est symétrique et définie positive, les relations suivantes sont vérifiées

$$A^T = A \text{ et } x^T A x \geq 0 \quad (\text{II.9})$$

Si la matrice A est antisymétrique, les relations suivantes sont vérifiées

$$A^T = -A \text{ et } x^T A x = 0 \quad (\text{II.10})$$

Avec A^T est la matrice transposée de A .

La matrice I_n est de dimension $(n \times n)$, elle présente la matrice identité.

Dans le reste de la thèse, $k \in \mathbb{N}$ désigne le nombre des itérations. $(*)_k$ indique le signal à l'itération k^{th} .

Définition II.1. (L_2 norm) : Pour toute $x \in \mathbb{R}^n$ définie sur $[0, T]$, la norme de x dans L_2 est définie comme suit

$$\|x\|_{L_2} \triangleq \left(\int_0^{+\infty} \|x\|^2 d\tau \right)^{\frac{1}{2}} \quad (\text{II.11})$$

Définition II.2. L_2 est l'espace des fonctions $x: \mathbb{R} \rightarrow \mathbb{R}$ telle que

$$\left(\int_0^{+\infty} \|x\|^2 d\tau \right)^{\frac{1}{2}} < \infty \quad (\text{II.12})$$

On donne les définitions de certaines normes qui seront utilisées dans la thèse.

La norme d'un vecteur X est définie par

$$\|X\| = \sqrt{X^T X} \quad (\text{II.13})$$

La norme d'une matrice est définie par

$$\|A\| = \sqrt{\lambda_{\max}(A^T A)} \quad (\text{II.14})$$

Définition II.3 (norme-Sup) : Pour toute $x \in \mathbb{R}^n$ définie sur $[0, T]$, la norme-Sup, $\|x(t)\|_{\infty} : [0, T] \times \mathbb{R}^n \rightarrow \mathbb{R}$ est définie comme

$$\|x(t)\|_{\infty} = \sup_{0 \leq t \leq T} \|x(t)\| \quad (\text{II.15})$$

Définition II.4 (norme- λ) : Pour toute $x \in \mathbb{R}^n$ définie sur $[0, T]$, la norme- λ , $\|x(t)\|_{\lambda}$ est définie comme suit

$$\|x(t)\|_{\lambda} = \sup_{0 \leq t \leq T} \{e^{-\lambda t} \|x(t)\|\} \quad (\text{II.16})$$

Remarque II.1 : A partir des définitions précédentes, la propriété d'équivalence des normes est définie comme

$$\|x(t)\|_{\lambda} \leq \|x(t)\|_{\infty} \leq e^{\lambda t} \|x(t)\|_{\lambda}, \quad \lambda > 0 \quad (\text{II.17})$$

Les méthodes d'analyse de la convergence des systèmes répétitifs nécessitent d'utiliser les lemmes suivants.

Lemme II.1 [YAN 01]

Soit une fonction $w(t) = [w_1(t), w_2(t), \dots, w_n(t)]^T \in \mathbb{R}^n$ définie pour $t \in [0, T]$, on a :

$$\left(\int_0^t \|w(s)\| ds \right) e^{-\lambda t} \leq \frac{1}{\lambda} \|w(s)\|_{\lambda}, \quad t \in [0, T] \quad (\text{II.18})$$

Lemme II.2 (Gronwall-Bellman) [Yan 01]

Soient $f(t)$ et $g(t)$ deux fonctions scalaires réelles et localement intégrables dans $[a, b]$.

Si la fonction $f(t)$ satisfait

$$f(t) \leq L + \int_0^t g(\tau)f(\tau)d\tau, \quad t \in [0, T] \quad (\text{II.19})$$

Alors

$$f(t) \leq L \exp\left(\int_0^t g(\tau)d\tau\right) \quad (\text{II.20})$$

Avec L est une constante.

Lemme II.3 [Yan 01] :

Soient $\xi(t)$, $\mu_1(t)$ et $\mu_2(t)$ des fonctions continues dans $[0, T]$. Si

$$\|\xi(t)\| \leq \Gamma \int_0^t \|\xi(s)\| ds + \Gamma_1 \int_0^t \|\mu_1(s)\| ds + \Gamma_2 \|\mu_2(t)\| \quad (\text{II.21})$$

Alors

$$\|\xi(t)\|_\lambda \leq \left(\frac{1}{\lambda} \Gamma_1 \|\mu_1(t)\|_\lambda + \Gamma_2 \|\mu_2(t)\|_\lambda\right) \exp\left(\frac{\Gamma}{\lambda}\right) \quad (\text{II.22})$$

Avec $\Gamma \geq 0$, $\lambda > 0$, Γ_1 et Γ_2 sont des constantes.

II.4.2 Méthode de la norme- λ

La méthode de lambda norme a été développée dans le premier papier académique de la CAI [ARI 84], et plusieurs travaux ont utilisé cette méthode pour démontrer la stabilité des systèmes linéaires [BOU 101] et non linéaires [BOU 102]. Cette technique est très simple et elle facilite l'analyse de stabilité. L'idée principale de cette méthode est basée sur l'utilisation des propriétés des normes du système à commander.

La stabilité du système est vérifiée si

$$\|e_{k+1}(t)\|_\lambda \leq \alpha \|e_k(t)\|_\lambda \quad (\text{II.23})$$

Telle que e_k l'erreur à l'itération k et $\alpha \in]0,1[$.

Il est à noter que cette méthode est utilisée pour l'analyse de stabilité lors de la conception de la commande des systèmes satisfaisant la condition de continuité globale de Lipschitz. Cette condition est définie comme suit.

Lemme II.4 : La fonction $f(x(t), t)$ satisfait la condition de continuité globale de Lipschitz pour $t \in [0, T]$, s'il existe une constante ρ pour tout $x_1(t), x_2(t) \in R \times [0, T]$, alors on a

$$\|f(x_1, t) - f(x_2, t)\| \leq \rho \|x_1 - x_2\| \quad (\text{II.24})$$

II.4.3 Méthode de Lyapunov

La méthode de Lyapunov est basée sur la théorie qui a été développée par le mathématicien Russe Alexandre Mikhailovich Lyapunov. Cette méthode a été dédiée pour la commande des systèmes répétitifs pour remédier à limitations dans l'utilisation de la méthode de Lambda norme. En fait, cette technique donne l'importance pour la variation du temps et des itérations en même temps. Pour cela l'analyse de stabilité pour les systèmes répétitifs peut être établie pour toutes les lois de commande proposées.

Cette méthode a été appliquée pour un large type des classes des systèmes dynamiques répétitifs pour l'analyse de stabilité lors de la conception de la commande, que ce soit pour des systèmes linéaires [CHI 04], des systèmes non linéaires [LI 13], des systèmes avec des perturbations externes [BOU 11], des systèmes à base des observateurs non linéaires [BOU 13], des systèmes avec la présence des non-linéarités inconnues à l'entrée [WEI 17].

Le principe de cette méthode est divisé en deux parties : la première partie consiste à démontrer la bornitude des erreurs de poursuite à travers les itérations et la deuxième partie consiste à assurer que les erreurs de poursuite tendent vers zéro à travers les itérations.

En effet, la première partie consiste à proposer une fonction candidate de Lyapunov qui est définie positive $V_k(t)$ dans le même intervalle de temps du système, puis de prouver qu'elle est décroissante à travers les itérations, c.-à-d.

$$\Delta V_k(t) = V_k(t) - V_{k-1}(t) < 0 \quad (\text{II.25})$$

Pour la deuxième partie, afin de trouver la condition suffisante qui assure la convergence des erreurs de poursuite vers zéro, nous devons démontrer bornitude de la fonction de Lyapunov à l'itération initiale $V_0(t)$ sur le même intervalle de temps du système. La condition est vérifiée si

$$\dot{V}_0(t) < 0 \quad (\text{II.26})$$

II.4.4 Exemple illustratif 1

Soit le système non linéaire suivant [YAN 01] :

$$\begin{cases} \dot{x}_k(t) = f(x_k(t), t) + B(t)u_k(t) \\ y_k(t) = h(x_k(t), t) \end{cases} \quad (\text{II.27})$$

où t : est le temps , $k \in \mathbb{N}$ désigne l'indice des itérations, $x(t) \in \mathbb{R}^n$: est le vecteur d'état avec n l'ordre du système, $u(t) \in \mathbb{R}^m$: est le vecteur de commande, m le nombre d'entrées, $y(t) \in \mathbb{R}^p$: est le vecteur de sortie du système et p le nombre de sorties, $f \in \mathbb{R}^n$ et $h \in \mathbb{R}^p$ sont des vecteurs composés de fonctions continues. f peut-être structurellement inconnue, $B \in \mathbb{R}^{n \times m}$: est une matrice composée de fonctions continues.

Dans cet exemple les hypothèses suivantes sont nécessaires.

Hypothèse II.1 : Les conditions initiales sont les mêmes pour chaque cycle $x_k(0) = x^0$.

Hypothèse II.2 : La sortie désirée $y_d(t)$ est continue pour $t \in [0, T]$.

Théorème II.1 [YAN 01] : Etant donné le système non linéaire (II.27) vérifiant les hypothèses II.1 et II.2, en appliquant la loi de commande suivante

$$u_{k+1}(t) = u_k(t) + K(t)\dot{e}_k(t) \quad (\text{II.28})$$

Si

1. $h(x_k(t), t)$ est bornée dans $\mathbb{R}^n \times [0, T]$,
2. $\sup_{(x,t \in [0,T] \times \mathbb{R}^n)} \|I_m - h(x_k(t), t)B(t)K(t)\| < 1$

Alors

$$\lim_{k \rightarrow \infty} y_k(t) = y_d(t), t \in [0, T].$$

Preuve

La solution générale $x_{k+1}(t)$ peut-être écrite sous la forme suivante

$$x_{k+1}(t) = x_{k+1}(0) + \int_0^t \left(f(x_{k+1}(\tau)) + B(\tau)u_{k+1}(\tau) \right) d\tau \quad (\text{II.29})$$

En utilisant (II.28) et l'hypothèse (II.1), on obtient

$$x_{k+1}(t) = x_k(0) + \int_0^t f(x_{k+1}(\tau)) d\tau + \int_0^t B(\tau)(u_k(\tau) + K(\tau)\dot{e}_k(\tau)) d\tau \quad (\text{II.30})$$

$$\begin{aligned}
x_{k+1}(t) = x_k(0) + \int_0^t f(x_{k+1}(\tau)) d\tau + \int_0^t B(\tau)u_k(\tau) d\tau + B(t)K(t)e_k(t) \\
- \int_0^t \frac{d(B(\tau)K(\tau))}{d\tau} e_k(\tau) d\tau
\end{aligned} \tag{II.31}$$

En ajoutant et retranchant $\int_0^t f(x_k(\tau)) d\tau$, (II.31) devient

$$\begin{aligned}
x_{k+1}(t) = x_k(t) + \int_0^t \left(f(x_{k+1}(\tau)) - f(x_k(\tau)) \right) d\tau + B(t)K(t)e_k(t) \\
- \int_0^t \frac{d(B(\tau)K(\tau))}{d\tau} e_k(\tau) d\tau
\end{aligned} \tag{II.32}$$

En utilisant les propriétés des normes et Lemme (II.4), on obtient

$$\begin{aligned}
& \|x_{k+1}(t) - x_k(t)\| \\
& \leq \rho \int_0^t \|x_{k+1}(\tau) - x_k(\tau)\| d\tau + \max_{t \in [0, T]} \{ \|B(t)K(t)\| \} \|e_k(t)\| \\
& + \max_{t \in [0, T]} \left\{ \left\| \frac{d(B(t)K(t))}{dt} \right\| \right\} \int_0^t \|e_k(\tau)\| d\tau
\end{aligned} \tag{II.33}$$

En appliquant Lemme (II.3), il vient

$$\|x_{k+1}(t) - x_k(t)\|_\lambda \leq \gamma \exp\left(\frac{\rho}{\lambda}\right) \|e_k(t)\|_\lambda \tag{II.34}$$

Avec : $a = \{ \|B(t)K(t)\| \}$, $b = \max_{t \in [0, T]} \left\{ \left\| \frac{d(B(t)K(t))}{dt} \right\| \right\}$ et $\gamma = a + \frac{b}{\lambda}$

D'après le théorème de la valeur moyenne, il existe $\xi_k(t)$ telle que

$$e_{k+1}(t) - e_k(t) = y_k(t) - y_{k+1}(t) = h_x(\xi_k(t))(x_k(t) - x_{k+1}(t)) \tag{II.35}$$

De (II.32), on trouve

$$\begin{aligned}
e_{k+1}(t) = & \left(I_m - h_x(\xi_k(t))B(t)K(t) \right) e_k(t) \\
& - h_x(\xi_k(t)) \left[\int_0^t \left(f(x_{k+1}(\tau)) - f(x_k(\tau)) \right) d\tau \right. \\
& \left. - \int_0^t \frac{d(B(\tau)K(\tau))}{d\tau} e_k(\tau) d\tau \right]
\end{aligned} \tag{II.36}$$

En utilisant Lemme II.1 et (II.32), on obtient

$$\|e_{k+1}(t)\|_\lambda = \left\{ \|I_m - h_x(\xi_k(t))B(t)K(t)\| + \frac{\beta}{\lambda} \left[\rho \gamma \exp\left(\frac{\rho}{\lambda}\right) + b \right] \right\} \|e_k(t)\|_\lambda \quad (\text{II.37})$$

Avec $\beta = \sup_{(x,t) \in \mathbb{R}^n \times [0,T]} \{h(x_k(t))\}$, en vérifiant la deuxième condition du théorème (II.1), il s'en suit qu'il existe $\lambda (\lambda > 0)$ assez grand, tel que

$$\sup_{(x,t) \in \mathbb{R}^n \times [0,T]} \|I_m - h_x(\xi_k(t))B(t)K(t)\| + \frac{\beta}{\lambda} \left[\rho \gamma \exp\left(\frac{\rho}{\lambda}\right) + b \right] = \alpha < 1 \quad (\text{II.38})$$

D'où

$$\|e_{k+1}(t)\|_\lambda \leq \alpha \|e_k(t)\|_\lambda \quad (\text{II.39})$$

La condition de convergence est vérifiée, donc

$$\lim_{k \rightarrow \infty} \|e_k(t)\|_\lambda = 0 \quad t \in [0, T] \quad (\text{II.40})$$

II.4.5 Exemple illustratif 2

Soit le système non linéaire dynamique suivant [LI 16] :

$$\dot{x}_k(t) = f(x_k(t), t) + B(x_k(t), t)u_k(t) + w(x_d(t), t) \quad (\text{II.41})$$

où t : est le temps, $k \in \mathbb{N}$ désigne l'indice des itérations, $x(t) \in \mathbb{R}^n$: est le vecteur d'état avec n l'ordre du système, $u(t) \in \mathbb{R}^n$: est le vecteur de commande, $f \in \mathbb{R}^n$ est un vecteur composé des fonctions continues supposées inconnues avec une norme bornée, $B \in \mathbb{R}^{n \times n}$: est une matrice composée des fonctions connues.

La trajectoire désirée $x_d(t)$ est générée à partir de l'équation suivante

$$\dot{x}_d(t) = w(x_d(t), t), \quad t \in [0, T] \quad (\text{II.42})$$

L'erreur de poursuite est donnée par : $e_k(t) = x_k(t) - x_d(t)$. La dynamique des erreurs à l'itération k^{ieme} est définie comme suit

$$\dot{e}_k(t) = f(x_k(t), t) + B(x_k(t), t)u_k(t) \quad (\text{II.43})$$

L'hypothèse suivante est nécessaire dans la suite de cet exemple

Hypothèse II.3 : La norme de la fonction inconnue f est bornée par une fonction connue

$$\|f(x_k(t), t)\|_{L_2} \leq \mu(x_k(t), t) \quad (\text{II.44})$$

Théorème II.2 [LI 16] : Etant donné le système non linéaire (II.41) vérifiant les hypothèses II.1 et II.3, en appliquant la loi de commande suivante

$$\begin{cases} u_k(t) = B_k^{-1}[-\Gamma e_k(t) - \hat{\theta}_k(t)\mu(x_k(t), t)\text{sgn}(e_k(t))] \\ \hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + \gamma\mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)| \\ \hat{\theta}_{-1}(t) = 0 \end{cases} \quad t \in [0, T] \quad (\text{II.45})$$

On obtient

$$\lim_{k \rightarrow \infty} \|e_k(t)\|_\lambda = 0 \quad t \in [0, T] \quad (\text{II.46})$$

$\Gamma \in \mathbb{R}^{n \times n}$: la matrice des gains de commande, elle est symétrique et définie positive. γ est un gain positif.

Preuve : La démonstration est basée sur la méthode de Lyapunov

La fonction de Lyapunov est choisie comme suit

$$V_k(t) = \frac{1}{2} e_k^T(t) e_k(t) + \frac{1}{2\gamma} \int_0^t \varphi_k^2 d\tau \quad (\text{II.47})$$

avec $\varphi_k = \hat{\theta}_k - 1$. Dans la première partie on prouve que cette fonction est décroissante à travers les itérations.

Soit

$$\Delta V_k(t) = V_k(t) - V_{k-1}(t) \quad (\text{II.48})$$

A partir de (II.47), on trouve

$$\Delta V_k(t) = \frac{1}{2} e_k^T(t) e_k(t) - \frac{1}{2} e_{k-1}^T(t) e_{k-1}(t) + \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \quad (\text{II.49})$$

Cela implique que

$$\Delta V_k(t) \leq \frac{1}{2} e_k^T(t) e_k(t) + \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \quad (\text{II.50})$$

L'équation (II.50) peut être réécrite sous forme

$$\Delta V_k(t) \leq \int_0^t e_k^T(t) \dot{e}_k(t) d\tau + \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \quad (\text{II.51})$$

En utilisant (II.45), on obtient

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(e_k^T(t) f(x_k(t), t) \right. \\ & \left. - e_k^T(t) \Gamma e_k(t) - e_k^T(t) \hat{\theta}_k(t) \mu(x_k(t), t) \operatorname{sgn}(e_k(t)) \right) d\tau \\ & + \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \end{aligned} \quad (\text{II.52})$$

Comme $e_k^T(t) f(x_k(t), t) \leq \mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)|$ et $\varphi_k = \hat{\theta}_k - 1$, alors

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(-e_k^T(t) \Gamma e_k(t) - \varphi_k \mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)| \right) d\tau \\ & + \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \end{aligned} \quad (\text{II.53})$$

Le dernier terme de (II.53), peut être réécrit comme suit

$$\frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau = \frac{1}{2\gamma} \int_0^t [2(1 - \hat{\theta}_k) + (\hat{\theta}_k - \hat{\theta}_{k-1})](\hat{\theta}_{k-1} - \hat{\theta}_k) d\tau \quad (\text{II.54})$$

En remplaçant (II.45), il vient

$$\begin{aligned} & \frac{1}{2\gamma} \int_0^t (\varphi_k^2 - \varphi_{k-1}^2) d\tau \\ & \leq \int_0^t \left(\varphi_k \mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)| \right) d\tau - \frac{1}{2\gamma} \int_0^t (\hat{\theta}_k - \hat{\theta}_{k-1})^2 d\tau \end{aligned} \quad (\text{II.55})$$

Donc, l'équation (II.53) devient

$$\Delta V_k(t) \leq - \int_0^t e_k^T(t) \Gamma e_k(t) d\tau \leq -\lambda_{\min}(\Gamma) \int_0^t \|e_k(t)\|_{L_2}^2 d\tau \quad (\text{II.56})$$

Il est clair que $\Delta V_k(t)$ est décroissante à travers les itérations.

Dans la deuxième partie on va montrer que la fonction de Lyapunov est bornée à la première itération. En dérivant les deux côtés de (II.47) à l'itération « k=0 », on obtient

$$\dot{V}_0(t) = e_0^T(t) \dot{e}_0(t) + \frac{1}{2\gamma} \dot{\varphi}_0^2 \quad (\text{II.57})$$

En utilisant (II.43) et (II.45), il vient

$$\begin{aligned} \dot{V}_0(t) = & e_k^T(t) f(x_k(t), t) - e_k^T(t) \Gamma e_k(t) - e_k^T(t) \hat{\theta}_k(t) \mu(x_k(t), t) \operatorname{sgn}(e_k(t)) \\ & + \frac{1}{2\gamma} \varphi_0^2 \end{aligned} \quad (\text{II.58})$$

Cela implique que

$$\begin{aligned} \dot{V}_0(t) \leq & -e_k^T(t) \Gamma e_k(t) - \hat{\theta}_k(t) \mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)| + \mu(x_k(t), t) \sum_{i=1}^n |e_{i,k}(t)| \\ & + \frac{1}{2\gamma} \varphi_0^2 \end{aligned} \quad (\text{II.59})$$

De (II.45) et pour $k=0$, on a

$$\hat{\theta}_0(t) = \gamma \mu(x_0(t), t) \sum_{i=1}^n |e_{i,0}(t)| \quad (\text{II.60})$$

En remplaçant (II.60) dans (II.59), on obtient

$$\dot{V}_0(t) \leq -e_k^T(t) \Gamma e_k(t) - \frac{1}{\gamma} \hat{\theta}_0^2(t) + \frac{1}{\gamma} \hat{\theta}_0(t) + \frac{1}{2\gamma} \varphi_0^2 \quad (\text{II.61})$$

Ce qui nous donne

$$\dot{V}_0(t) \leq -e_k^T(t) \Gamma e_k(t) - \frac{1}{\gamma} \varphi_0 \hat{\theta}_0(t) + \frac{1}{2\gamma} \varphi_0^2 \quad (\text{II.62})$$

D'où

$$\dot{V}_0(t) \leq -e_k^T(t) \Gamma e_k(t) - \frac{1}{\gamma} \varphi_0 - \frac{1}{2\gamma} \varphi_0^2 \quad (\text{II.63})$$

D'après l'inégalité de Yong's $\varphi_0 \leq \|\varphi_0\| \leq c\varphi_0^2 + \frac{1}{4c} \forall c > 0$. Pour $0 < c < \frac{1}{2}$ on trouve

$$\dot{V}_0(t) \leq -\lambda_{\min} \|e_k(t)\|_2^2 - \left(\frac{1}{2} - c\right) \frac{1}{\gamma} \varphi_0^2 + \frac{1}{4c\gamma} \quad (\text{II.64})$$

Si l'inégalité $\lambda_{\min} \|e_k(t)\|_2^2 + \left(\frac{1}{2} - c\right) \frac{1}{\gamma} \varphi_0^2 \geq \frac{1}{4c\gamma}$ est vérifiée, alors $\dot{V}_0(t) \leq 0$ et donc $V_0(t)$ est bornée.

Dans la troisième partie on va démontrer que l'erreur tend vers zéro.

La fonction de Lyapunov peut être écrite comme

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_j(t) \quad (\text{II.65})$$

En utilisant (II.56), on obtient

$$V_k(t) = V_0(t) - \lambda_{\min}(\Gamma) \sum_{j=1}^k \int_0^t \|e_j(\tau)\|_{L_2}^2 d\tau \quad (\text{II.66})$$

Comme $V_k(t)$ est positive et bornée, on trouve

$$\sum_{j=1}^k \int_0^t \|e_j(\tau)\|_{L_2}^2 d\tau \leq \frac{V_0(t)}{\lambda_{\min}(\Gamma)} < M \quad (\text{II.67})$$

Ça veut dire que

$$\lim_{k \rightarrow \infty} \|e_k(t)\|_{\lambda} = 0 \quad t \in [0, T] \quad (\text{II.68})$$

II.5 Réseaux de neurones

Un réseau de neurones artificiel est un système informatique dont sa conception s'inspire du fonctionnement des neurones biologiques. Le but de son utilisation c'est de remplacer l'être humain dans la prise des décisions sur la base de connaissances imprécises. Actuellement, les réseaux de neurones artificiels sont utilisés dans tous les domaines tel que l'automatique, électronique, médecine, informatique, traitement des images... L'élément principal dans l'architecture des réseaux de neurones artificiels s'appelle le neurone formel. Le neurone formel est une fonction algébrique, non linéaire et bornée qui présente la modélisation mathématique du fonctionnement du neurone biologique.

II.5.1 Neurone biologique

Dans le cerveau humain il y a environ de 10^{12} neurones, avec 1000 à 10000 synapses (connexions) par neurone. Les cellules nerveuses appelées " neurones ", sont les éléments de base du système nerveux central. Elles sont constituées de trois parties essentielles : le corps cellulaire, les dendrites et l'axone.

II.5.1.1 Corps cellulaire

Il contient le noyau du neurone et effectue les transformations biochimiques nécessaires à la synthèse des enzymes et des autres molécules qui assurent la vie de neurone. Sa forme est pyramidale ou sphérique dans la plupart des cas, contient aussi les autres molécules essentielles à la vie de la cellule. Elle dépend souvent de sa position dans le cerveau. Ce corps cellulaire fait quelques microns de diamètre.

II.5.1.2 Dendrites

Chaque neurone possède une chevelure de dendrites. Celles-ci sont de fines extensions tubulaires, de quelques dixièmes de microns de diamètre et d'une longueur de quelques dizaines de microns. Elles sont les récepteurs principaux du neurone qui servent à capter les signaux qui lui parviennent.

II.5.1.3 Axone

L'axone, qui est à proprement parler la fibre nerveuse, sert de moyen de transport pour les signaux émis par le neurone. Il se distingue des dendrites par sa forme et par les propriétés de sa membrane externe. En effet, il est généralement plus long que les dendrites, et se ramifie à son extrémité, là où il communique avec les autres neurones, alors que les ramifications des dendrites se produisent plutôt près du corps cellulaire.

II.5.2 Neurone formel (artificiel)

La figure ci-dessous montre la structure d'un neurone artificiel. Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones amont. A chacune de ces entrées est associé un poids w abréviation de weight (poids en anglais) représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones aval. A chaque connexion est associé un poids.

La structure générale des réseaux de neurones artificiels est formée de combinaison de plusieurs neurones formels, qui sont associés en succession de couches placées en parallèle.

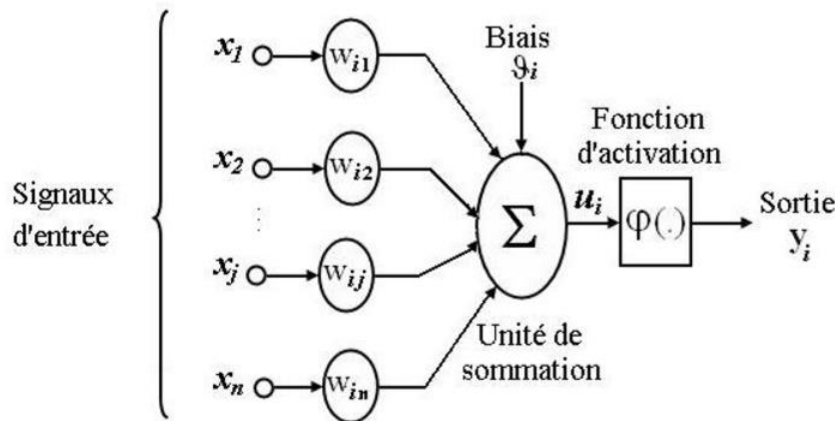


Figure II.2 Model d'un neurone formel.

La nature des entrées et la sortie pour le neurone formel peuvent être de types booléens ou réels. u_i est définie par la somme des produits des entrées avec leurs poids associés w . Par la suite, la sortie de neurone formel est calculée suivant la fonction d'activation qui détermine l'état interne de neurone suivant la fonction d'entrée totale.

La sortie u_k du neurone formel est donnée par la relation

$$u_i = \sum_{j=1}^n w_{ij} x_j \quad (\text{II.69})$$

A partir de cette valeur, une fonction d'activation calcule la valeur de sortie y du neurone. C'est cette valeur qui sera transmise aux neurones avals :

$$y_i = \varphi(u_j) \quad (\text{II.70})$$

II.5.3 Fonction d'activation

La fonction d'activation sert à introduire une non linéarité dans le fonctionnement du neurone. Elle calcule la valeur de l'état du neurone. C'est cette valeur qui sera transmise aux neurones avals. Il existe de nombreuses formes possibles pour la fonction d'activation. Les plus utilisées sont les fonctions seuil, linéaire et sigmoïde.

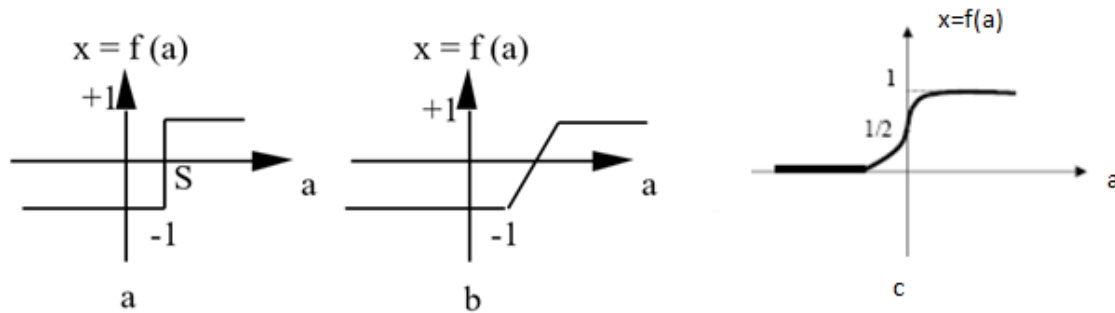


Figure II.3 Différents types de fonctions d'activation, a) fonction à seuil, b) linéaire, c) sigmoïde.

II.5.4 Apprentissage des réseaux de neurones

Pour définir et modifier les valeurs des poids de chaque connexion dans le réseau de neurones, on distingue deux types d'apprentissage :

- **Apprentissage supervisé** : l'apprentissage est dit supervisé lorsque le réseau de neurones est alimenté avec la connaissance des sorties désirées. Ce type consiste à trouver les valeurs des poids du réseau de telle sorte que pour les exemples utilisés lors d'apprentissage, les sorties du réseau de neurones soient proches le plus fort possible des sorties désirées.
- **Apprentissage non supervisé** : ce type d'apprentissage est plus complexe que le premier, il consiste à détecter automatiquement des réalités qui figurent dans les exemples présentés et à modifier des connexions pour que les exemples ayant les mêmes caractéristiques de régularité qui provoquent la même sortie. Les réseaux de neurones non supervisés les plus connus sont les réseaux de Kohonen.

II.5.5 Réseaux de neurones à fonctions de base radiales (RBF)

Les réseaux de neurones à fonctions de base radiales (RBF) sont une classe particulière des réseaux de neurones artificiels. La particularité de ce type des réseaux réside dans ses conceptions. Son architecture est fixée, elle se compose d'une couche d'entrée, une couche de sortie et une seule couche cachée, comme il est indiqué dans la figure suivante.

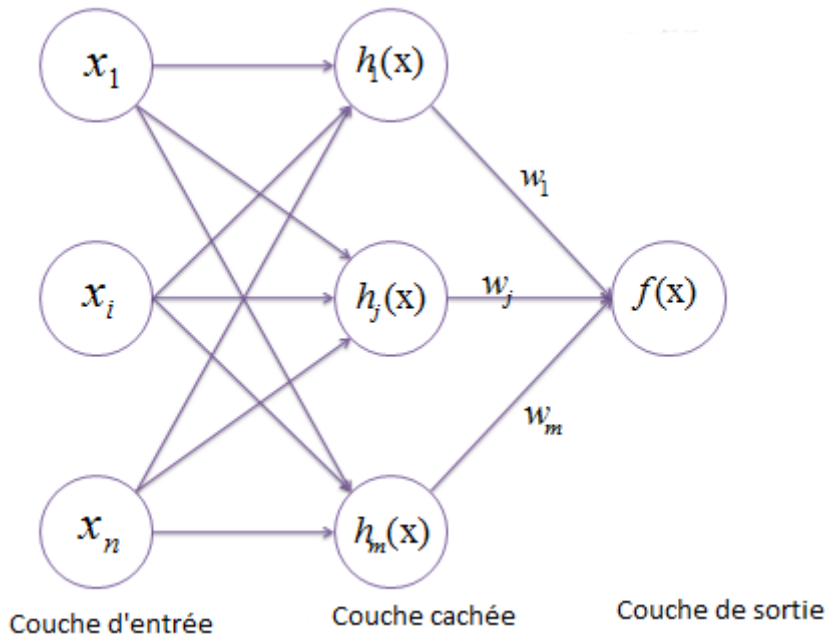


Figure II.4 Architecture du réseau RBF.

Ce qui le rend type le plus simple à utiliser et le plus rapide. Dans la littérature, on trouve que la fonction d'activation la plus utilisée dans les réseaux de neurones à fonctions de base radiales est la fonction gaussienne. Sa forme générale est définie par

$$h_i = \exp\left(-\frac{\|x_i - \mu_i\|^2}{2\sigma_i}\right) \quad (\text{II.71})$$

Où : $\|v\|$ est la norme euclidienne du vecteur v .

x_i représente le vecteur d'entrée, μ_i et σ_i sont le centre et le rayon d'influence de neurone i de la couche cachée, respectivement.

La sortie du réseau RBF est donnée comme suit

$$f(x) = \sum_{i=1}^N w_i \exp\left(-\frac{\|x_i - \mu_i\|^2}{2\sigma_i}\right) \quad (\text{II.72})$$

D'où : N est le nombre des neurones de la couche d'entrée, w_i les poids des neurones de sortie.

L'objectif d'approximation d'une fonction inconnue $f(x)$ est de trouver une fonction estimée $\hat{f}(x)$, qui peut représenter le plus fort possible la fonction $f(x)$. Les réseaux de neurones à fonctions de base radiales sont utilisés dans la littérature pour résoudre le problème d'approximation des fonctions complexes. En raison de son architecture feedforward puissante,

le RBF est devenu un bon approximateur universel, qui est capable d'approximer n'importe quelle fonction complexe non linéaire dans des espaces de grandes dimensions. Les résultats obtenus avec le RBF sont les plus rapides et avec une grande précision par rapport à d'autres types des réseaux de neurones.

C'est dans ce sens que les RBFs semblent très intéressants en la commande des systèmes non linéaire pour approximer les fonctions inconnues des systèmes à commander ou bien les fonctions et les gains inconnus dans la commande proposée.

II.6 Systèmes flous

Les systèmes flous sont basés sur la technique de la logique floue qui a été introduite en 1965 par le professeur Lotfi Zadeh [ZAD 65]. Elle permet de traiter des variables exprimées en langage humain, dont les valeurs sont non exactes, ces valeurs peuvent varier entre le 0 et 1. L'objectif de cette technique est de remplacer l'être humain dans la prise de décision dans les divers domaines. L'intégration de cette technique dans le domaine de la commande adaptative, permet d'ouvrir des nouvelles perspectives et résoudre beaucoup des contraintes qui existent sur la commande des systèmes et par la suite, d'améliorer les performances des systèmes de commande. La figure II.3, illustre la structure générale d'un système flou à trois entrées et une seule sortie. Les composants de base de cette technique sont les suivants :

- **Fuzzification** : la fuzzification est la traduction des grandeurs physiques réelles des entrées à des grandeurs floues à l'aide des ensembles flous.
- **Mécanisme d'inférence floue** : l'inférence floue permet de traduire, à l'aide des bases de règles, la partie floue issue de la fuzzification en une nouvelle partie floue.
- **Base de règles** : cette base présente l'architecture générale du système. Elle contient les règles floues décrivant le comportement du système. La forme générale d'une règle floue est : Si permisse Alors conclusion.
- **Défuzzification** : la défuzzification est la traduction des grandeurs floues de la partie d'inférence en des grandeurs physiques réelles.

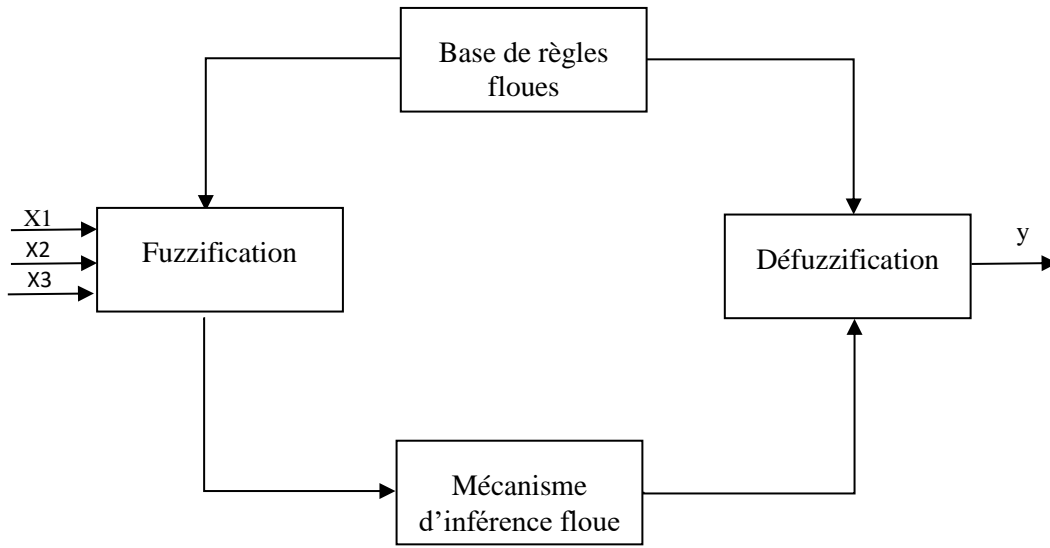


Figure II.5 Structure interne d'un système flou.

Les systèmes flous de Takagi-Sugeno (TS) sont des approximateurs universels, sont capables d'approximer n'importe quelle fonction réelle non linéaire $f(x)$ définie et continue sur un ensemble compact $\Omega_x \subseteq \mathcal{R}$.

Soit $x = [x_1, \dots, x_n] \in X \subseteq \mathcal{R}^n$ l'ensemble des entrées du système flou et $y \in \mathcal{R}$ sa sortie. Pour chaque entrée x_i nous définissons m_i ensembles flous F_j^i dans X_i , avec $i = 1, \dots, n$ et $j = 1, \dots, m_i$. Les règles floues R_k ($k = 1, 2, \dots, N$) sont données sous la forme :

$$R_k: \text{SI } x_1 \text{ est } G_k^1 \text{ et } \dots \text{ et } x_n \text{ est } G_k^n \text{ ALORS } y = f_k(x)$$

Où $G_i^k \in \{F_i^1, \dots, F_i^{m_i}\}$, et $f_k(x)$, $j = 1, \dots, m_i$ sont des ensembles flous.

Généralement, on note $f_k(x)$ un polynôme en fonction des variables d'entrées. Donc le polynôme linéaire $f_k(x)$ peut réécrire sous la forme :

$$f_k(x) = a_0^k + a_1^k x_1 + \dots + a_n^k x_n = a_0^k + \sum_{i=1}^n a_i^k x_i \quad (\text{II.73})$$

Par l'utilisation de la fuzzification singleton, la méthode de produit algébrique pour l'implication, et la défuzzification de somme moyenne, la sortie du système flou est donnée par:

$$y(x) = \frac{\sum_{k=1}^N \mu_k f_k(x)}{\sum_{k=1}^N \mu_k} = \frac{\sum_{k=1}^N \mu_k [a_0^k + a_1^k x_1 + \dots + a_n^k x_n]}{\sum_{k=1}^N \mu_k} \quad (\text{II.74})$$

avec : $\mu_k = \prod_{i=1}^n \mu_{G_i^k}(x_i)$, $\mu_{G_i^k} \in \{\mu_{G_i^1}, \dots, \mu_{G_i^{m_i}}\}$

où $\mu_{G_i^j}$ représente la fonction d'appartenance associée à l'ensemble flou G_i^j .

Donc la sortie peut être réécrite comme suit :

$$y(x) = w(x)\theta^T \quad (\text{II.75})$$

avec : $\theta = [a^1 \dots \dots a^k]$ est le vecteur des paramètres de la partie conclusion du système flou, $W(x) = [w_1(x) \dots \dots w_1(x)]$ est le vecteur des fonctions de base floue où chaque élément w_k est défini par :

$$w_k(x) = \frac{\mu_k}{\sum_{k=1}^N \mu_k} \quad (\text{II.76})$$

Le théorème suivant présente la capacité de l'approximation universelle par cette technique.

Théorème II.3. Pour toute fonction réelle non linéaire $f(x)$ définie et continue sur un ensemble compact $\Omega_x \subseteq \mathcal{R}$, il existe un système flou $y(x)$ de la forme (1.74) tel que :

$$\sup_x |f(x) - y(x)| < \varepsilon \quad (\text{II.77})$$

avec : ε est une constante positive.

II.7 Conclusion

Dans ce chapitre, nous avons donné le principe de base de la commande par apprentissage itératif de systèmes répétitifs. Nous avons défini, également, la forme générale de cette approche en donnant les différentes structures qui existent dans la littérature. Dans la deuxième partie de ce chapitre, nous avons présenté les deux méthodes utilisées pour l'analyse la stabilité lors de la conception de la CAI, à savoir, la méthode de norme-Lambda et l'approche de Lyapunov. Ces deux méthodes sont utilisées dans deux exemples illustratifs. La dernière partie de ce chapitre est dédiée à la présentation d'un résumé sur les réseaux de neurones à fonctions de base radiales et les systèmes flous qui vont être utilisés dans cette thèse, pour la conception et le développement de nouveaux schémas de commande des systèmes répétitifs.

Chapitre. III.

Commande par apprentissage itératif d'une classe des systèmes non linéaires multivariables

III.1 Introduction

Au cours des dernières décennies, l'analyse et la synthèse des lois de commande par apprentissage itératif ont connu un développement important. Cette commande a été le sujet de plusieurs travaux de recherche publiés sur les systèmes répétitifs [JI 16], [BOU 16] [ZHA 16], [KIM 17] et [BOU 20]. Les classes des systèmes considérées dans ces articles doivent vérifier la condition de continuité globale de Lipschitz. Sans utiliser cette condition, il est difficile de garantir la stabilité du système en boucle fermée. À travers notre recherche nous avons trouvé quelques travaux qui ont étudié ce problème, par exemple dans [CHE 11] et [JIN 17], les auteurs ont proposé une loi de commande sans utiliser la supposition de Lipschitz, pour une classe des systèmes dont les non-linéarités sont supposées bornées par des constantes. Dans [LI 16], une loi de commande par apprentissage itératif robuste a été développée sans utiliser cette condition, dont les non-linéarités du système sont supposées bornées par une fonction connue. Ceci nous a motivé pour améliorer les travaux cités précédemment en proposant des lois de commande par apprentissage itératif des systèmes non linéaires répétitifs sans la nécessité de satisfaire la condition de continuité globale de Lipschitz.

D'autre part, l'un des problèmes existant dans la synthèse de la commande par apprentissage itératif est la nécessité de connaître à priori la direction de la commande (le signe de la matrice de commande) [DIN 14], [JIN 16], [LI 17] et [YOU 18]. Il est très important et plus pratique de proposer une loi de commande sans utiliser cette hypothèse. Dans [YUM 13], une loi de commande par apprentissage itératif robuste a été proposée pour un système non linéaire SISO

avec le signe de commande est inconnu. Dans [WEI 17], sur le même système précédant, une loi de commande par apprentissage itératif adaptative a été présentée pour résoudre ce problème. Bien qu'il y ait quelques travaux qui traitent ce problème pour les systèmes SISO, il y a peu de travaux sur les systèmes non linéaires multivariables.

Dans ce chapitre, nous proposons une loi de commande par apprentissage itératif à base des réseaux de neurones à fonctions de base radiales (RBF) pour les systèmes non linéaires multivariables sans la nécessité de satisfaire la condition de continuité globale de Lypschitz et sans connaître le signe de matrice de commande [BEN 191]. En utilisant la théorie de Lyapunove, la stabilité asymptotique du système en boucle fermée est démontrée. A la fin de ce chapitre, une comparaison a été faite entre les résultats de simulation obtenus et d'un autre schéma de commande existant dans la littérature pour montrer l'efficacité de la méthode proposée.

III.2 Position du Problème

Considérons une classe des systèmes non linéaires MIMO incertains donnée par le modèle d'état suivant à la $k^{\text{ième}}$ itération :

$$\dot{x}_k = f(x_k(t), t) + B(x_k(t), t)u_k(t) \quad (\text{III.1})$$

où $x_k \in R^n, u_k \in R^n$ représentent le vecteur d'état et la commande, respectivement.

$f(*) = [f_1(*), \dots, f_n(*)] \in R^n$ est une fonction non linéaire inconnue, $B \in R^{n \times n}$ est une matrice de commande inconnue et son signe est inconnu. $f(x_k(t))$ et $B(x_k(t))$ sont bornées par des fonctions inconnues :

$$\|\dot{x}_d(t) - f(x_k(t))\| \leq \varphi(x_k(t)) \quad (\text{III.2})$$

et

$$\|B(x_k(t))\| \leq \xi(x_k(t)) \quad (\text{III.3})$$

avec $\varphi(x_k(t))$ et $\xi(x_k(t))$ sont des fonctions non linéaires inconnues, $\dot{x}_d(t) \in R^n$ est la dérivée du vecteur des trajectoires désirées $x_d(t)$.

En utilisant les réseaux de neurones à fonctions de base radiales (RBF), la fonction non linéaire $\varphi(x_k(t))$ peut-être approximée comme suit

$$\hat{\phi}(x_k(t)) = \hat{\theta}_k^T(t) \phi(x_k(t)) \quad (\text{III.4})$$

où $\phi(x_k)$ est un vecteur de RBF. Dans ce travail, on utilise la fonction gaussienne

$$\phi_i(x_k(t)) = \exp\left(-\frac{\|x_k(t) - C_i\|^2}{\sigma_i^2}\right) \quad (\text{III.5})$$

où $C_i \in R^l$ et σ_i sont le centre et le rayon d'influence de neurone, respectivement.

$\hat{\theta}_k(t) \in R^l$ est le vecteur d'estimation de $\theta_k(t)$ à travers les itérations. La loi d'adaptation proposée est comme suit :

$$\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + \beta e_k^T(t) \text{sgn}(e_k(t)) \phi(x_k(t)) \quad (\text{III.6})$$

avec $\hat{\theta}_{-1}(t) = 0$ et β est une constante positive. L'erreur d'estimation est donnée par

$$\tilde{\theta}_k(t) = \theta_k(t) - \hat{\theta}_k(t) \quad (\text{III.7})$$

L'erreur de poursuite à la $k^{\text{ième}}$ itération est définie par :

$$e_k(t) = x_d(t) - x_k(t) \quad (\text{III.8})$$

Par dérivation, on obtient

$$\dot{e}_k(t) = \dot{x}_d(t) - \dot{x}_k(t) \quad (\text{III.9})$$

En remplaçant (III.1) dans (III.9), il vient

$$\dot{e}_k(t) = \dot{x}_d(t) - f(x_k(t)) - B(x_k(t))u_k(t) \quad (\text{III.10})$$

Objectif : Notre objectif est de développer une loi de commande par apprentissage itératif à base des réseaux de neurones $u_k(t)$ pour forcer la sortie réelle $x_k(t)$ à suivre une trajectoire désirée $x_d(t)$ sur un intervalle de temps fixe $t \in [0, T]$, lorsque k tend vers l'infini.

L'hypothèse suivante est nécessaire.

Hypothèse III.1 : A chaque itération, les conditions initiales sont les mêmes, i.e., $x_k(0) = x_d(0)$.

Afin de traiter le problème de signe de la matrice de commande, nous utilisons la fonction de Nussbaum.

Définition III.1 : On peut dire que la fonction $N(*)$ est de type Nussbaum si elle vérifie les deux propriétés suivantes :

- 1) $\lim_{s \rightarrow +\infty} \sup \frac{1}{s} \int_0^s N(h) dh = +\infty$
- 2) $\lim_{s \rightarrow +\infty} \inf \frac{1}{s} \int_0^s N(h) dh = -\infty$

Lemme III.1 [GE 04] : Soient $V(*)$ et $h(*)$ deux fonctions continues définies sur un intervalle de temps $[T_0, T_f]$ avec $V(t) \geq 0 \forall t \in [T_0, T_f]$, $N(*)$ est une fonction continue de type Nussbaum. Si l'inégalité suivante est vérifiée :

$$V(t) \leq c_0 + e^{-c_1 t} \int_0^t (g(x(\tau))N(h) + 1) \dot{h} e^{c_1 \tau} d\tau, \quad \forall t \in [T_0, T_f]$$

Alors

$$V(t), h(t) \text{ et } \int_0^t g(x(\tau))N(h) \dot{h} d\tau \text{ sont bornés sur } [T_0, T_f].$$

où c_0 est une constante, c_1 est une constante positive quelconque, et $g(x(\tau))$ est une fonction continue dont ses valeurs appartiennent à l'intervalle fermé $t \in [T_0, T_f] - \{0\}$.

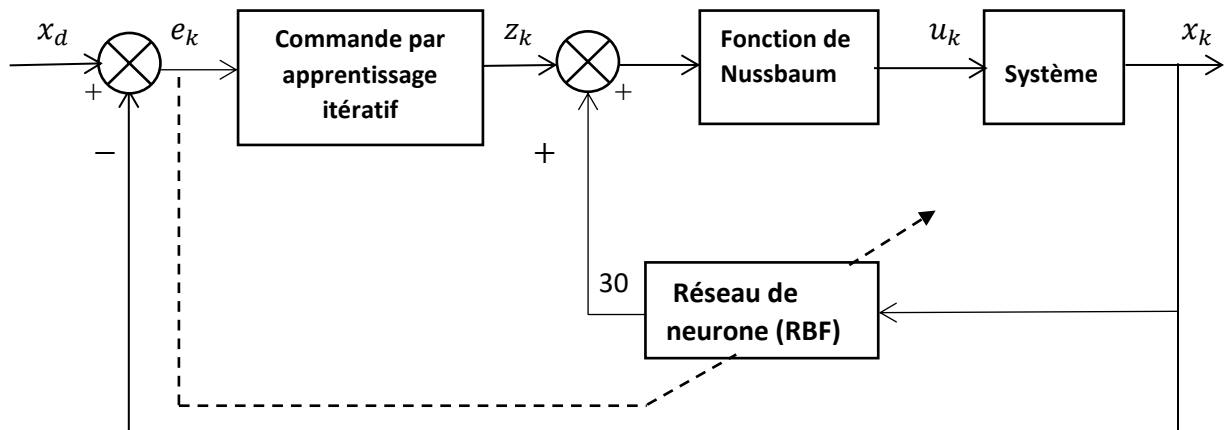
Dans ce travail, on choisit la fonction de Nussbaum comme suit : $N(h) = h^2 \cos(h)$.

III.3 Synthèse d'une loi de commande par apprentissage itératif

Dans ce travail, nous proposons une loi de commande par apprentissage itératif combinée à base des réseaux de neurones à fonctions de base radiales (RBF)

$$\begin{cases} u_k(t) = N(h_k(t))z_k(t) \\ \dot{h}_k(t) = |e_k^T(t)||z_k(t)| / h_{k+1}(0) = h_k(T), h(0) = 0 \\ z_k(t) = P e_k(t) + \hat{\varphi}_k(t) \text{sgn}(e_k(t)) \end{cases} \quad (\text{III.11})$$

Avec $N(*)$ est une fonction continue de type Nussbaum, $\hat{\varphi}_k$ est définie dans (III.4) et $P \in R^{n \times n}$ matrice symétrique positive, Le schéma bloc de la commande proposée est présenté par la Figure III.1.



Théorème III [BEN 191] : considérons le système non linéaire MIMO incertain (III.1), vérifiant l'hypothèse (III.1), en appliquant la loi de commande proposée (III.11) avec la loi d'adaptation (III.6) avec (III.4) et (III.5), alors

- i) $e_k(t) \in L_{\infty e}[0, T]$ et $\hat{\theta}_k(t) \in L_{\infty e}[0, T], \forall k \in N$.
- ii) $\lim_{k \rightarrow \infty} e_k(t) = 0, \forall t \in [0, T]$.

Preuve : La preuve de ce théorème est divisée en trois parties. Dans la première partie, on choisit une fonction de Lyapunov $V_k(t)$ qui est définie positive et montrer que cette fonction reste décroissante à travers les itérations. En seconde partie, on vérifie que $V_0(t)$ est bornée pour tout $t \in [0, T]$. À la fin, on montre que $\lim_{k \rightarrow \infty} e_k(t) = 0$ pour tout $t \in [0, T]$.

Partie 1 : La fonction candidate de Lyapunov est choisie comme suit

$$V_k(t) = \frac{1}{2} e_k^T(t) e_k(t) + \frac{1}{2\beta} \int_0^t \tilde{\theta}_k^T(\tau) \tilde{\theta}_k(\tau) d\tau \quad (\text{III.12})$$

On note que

$$\Delta V_k = V_k - V_{k-1} \quad (\text{III.13})$$

À partir de (III.12), il vient

$$\Delta V_k = \frac{1}{2} e_k^T(t) e_k(t) - \frac{1}{2} e_{k-1}^T(t) e_{k-1}(t) + \frac{1}{2\beta} \int_0^t \tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1} d\tau \quad (\text{III.14})$$

Cela implique que

$$\Delta V_k \leq \frac{1}{2} e_k^T(t) e_k(t) + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \quad (\text{III.15})$$

En utilisant (III.1), (III.15) peut être réécrite

$$\Delta V_k \leq \int_0^t e_k^T(\tau) \dot{e}_k(\tau) d\tau + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \quad (\text{III.16})$$

En remplaçant (III.10) dans (III.16), on trouve

$$\begin{aligned} \Delta V_k \leq & \int_0^t e_k^T(\tau) (x_d(\tau) - f(x_k) - B(x_k)u_k(\tau)) d\tau \\ & + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{III.17})$$

À partir de (III.11), on obtient

$$\begin{aligned} \Delta V_k \leq & \int_0^t e_k^T(\tau) (x_d(\tau) - f(x_k) - N(h_k(\tau))B(x_k)z_k(\tau) - Pe_k(\tau) \\ & - \hat{\varphi}_k(\tau) \text{sgn}(e_k(\tau)) + z_k(\tau)) d\tau + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{III.18})$$

En utilisant (III.2), (III.18) devient

$$\begin{aligned} \Delta V_k \leq & \int_0^t (\varphi_k(\tau) e_k^T(\tau) \text{sgn}(e_k(\tau)) - e_k^T(\tau) P e_k(\tau) - N(h_k(\tau)) e_k^T(\tau) B(x_k) z_k(\tau) \\ & - \hat{\varphi}_k(\tau) e_k^T(\tau) \text{sgn}(e_k(\tau)) + e_k^T(\tau) z_k(\tau)) d\tau \\ & + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{III.19})$$

En utilisant (III.3) et (III.11) et tant que $\tilde{\varphi}_k(t) = \varphi_k(t) - \hat{\varphi}_k(t)$, alors

$$\begin{aligned} \Delta V_k \leq & \int_0^t (\tilde{\varphi}_k(\tau) e_k^T(\tau) \text{sgn}(e_k(\tau)) - e_k^T(\tau) P e_k(\tau) + \xi(x_k(\tau)) |N(h_k(\tau))| \dot{h}_k(\tau) \\ & + \dot{h}_k(\tau)) d\tau + \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{III.20})$$

À partir de (III.6), on a

$$\tilde{\theta}_k(t) = \tilde{\theta}_{k-1}(t) - \beta e_k^T(t) \text{sgn}(e_k(t)) \phi(x_k) \quad (\text{III.21})$$

D'autre part, en utilisant (III.21), le dernier terme de l'équation (III.20) devient

$$\begin{aligned}
& \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\
&= \frac{1}{2\beta} \int_0^t \left((\tilde{\theta}_{k-1} - \beta e_k^T(t) \operatorname{sgn}(e_k(t)) \phi(x_k))^T \tilde{\theta}_k \right. \\
&\quad \left. - (\tilde{\theta}_k + \beta e_k^T(t) \operatorname{sgn}(e_k(t)) \phi(x_k))^T \tilde{\theta}_{k-1} \right) d\tau
\end{aligned} \tag{III.22}$$

Sachant que $\tilde{\theta}_{k-1}^T \tilde{\theta}_k = \tilde{\theta}_k^T \tilde{\theta}_{k-1}$, alors

$$\begin{aligned}
& \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\
&= \frac{1}{2\beta} \int_0^t \left(-(\beta e_k^T(t) \operatorname{sgn}(e_k(t)) \phi(x_k))^T \tilde{\theta}_k \right. \\
&\quad \left. - (\beta e_k^T(t) \operatorname{sgn}(e_k(t)) \phi(x_k))^T \tilde{\theta}_{k-1} \right) d\tau
\end{aligned} \tag{III.23}$$

Donc

$$\begin{aligned}
& \frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\
&= \frac{1}{2\beta} \int_0^t \left(-(\beta e_k^T \operatorname{sgn}(e_k) \phi(x_k))^T \tilde{\theta}_k \right. \\
&\quad \left. - (\beta e_k^T \operatorname{sgn}(e_k) \phi(x_k))^T \tilde{\theta}_{k-1} - (\beta e_k^T \operatorname{sgn}(e_k) \phi(x_k))^T \beta e_k^T \operatorname{sgn}(e_k) \phi(x_k) \right) d\tau
\end{aligned} \tag{III.24}$$

Ce qui implique que

$$\frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq \frac{1}{2\beta} \int_0^t \left(-2\beta e_k^T(t) \operatorname{sgn}(e_k(t)) \phi(x_k) \right)^T \tilde{\theta}_k d\tau \tag{III.25}$$

Sachant que $\phi(x_k)^T \tilde{\theta}_k = \tilde{\theta}_k^T \phi(x_k) = \tilde{\varphi}_k(t)$, alors

$$\frac{1}{2\beta} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq - \int_0^t \tilde{\varphi}_k(t) e_k^T(t) \operatorname{sgn}(e_k(t)) d\tau \tag{III.26}$$

En remplaçant (III.26) dans (III.20), on obtient

$$\Delta V_k \leq \int_0^t \left(-e_k^T(t) P e_k(t) + (\xi(x_k(t)) |N(h_k(t))| + 1) \dot{h}_k(t) \right) d\tau \tag{III.27}$$

En appliquant le théorème de Rayleigh-Ritz, l'équation (III.27) devient

$$\Delta V_k \leq \int_0^t \left(-\lambda_{\min}(P) \|e_k(t)\|^2 + (\xi(x_k(t)) |N(h_k(t))| + 1) \dot{h}_k(t) \right) d\tau \quad (\text{III.28})$$

D'où

$$\Delta V_k \leq \int_0^t (\xi(x_k(t)) |N(h_k(t))| + 1) \dot{h}_k(t) dt \quad (\text{III.29})$$

A partir de Lemme III.1 avec $c_0 = 0$ et $c_1 \cong 0$, alors ΔV_k est bornée et décroissante. D'autre part, $V_k(t)$ est bornée si et seulement si $V_0(t)$ est bornée.

Partie 2 : Nous démontrons que $V_0(t)$ est bornée $\forall t \in [0, T]$

À partir de (III.12), avec $k=0$, on a

$$V_0(t) = \frac{1}{2} e_0^T(t) e_0(t) + \frac{1}{2\beta} \int_0^t \tilde{\theta}_0^T \tilde{\theta}_0 dt \quad (\text{III.30})$$

Par dérivation, il vient

$$\dot{V}_0(t) = e_0^T(t) \dot{e}_0(t) + \frac{1}{2\beta} \tilde{\theta}_0^T \tilde{\theta}_0 \quad (\text{III.31})$$

En remplaçant (III.10) et (III.11) dans (III.31), on obtient

$$\begin{aligned} \dot{V}_0(t) = & e_0^T(t) \left(x_d(t) - f(x_0(t), t) - P e_0(t) \right. \\ & \left. - \hat{\varphi}_0(t) \text{sgn}(e_0(t)) - N(h_0(t)) B(x_0(t), t) z_0(t) + z_0(t) \right) \\ & + \frac{1}{2\beta} \tilde{\theta}_0^T \tilde{\theta}_0 \end{aligned} \quad (\text{III.32})$$

En utilisant (III.2) et (III.11), (III.32) devient

$$\begin{aligned} \dot{V}_0(t) = & \varphi_0(t) e_0^T(t) \text{sgn}(e_0(t)) - e_0^T(t) P e_0(t) - \hat{\varphi}_0(t) e_0^T(t) \text{sgn}(e_0(t)) \\ & + \xi(x_0(t)) |N(h_0(t))| \dot{h}_0(t) + \dot{h}_0(t) + \frac{1}{\beta} \tilde{\theta}_0^T \tilde{\theta}_0 \end{aligned} \quad (\text{III.33})$$

Sachant que $\tilde{\varphi}_0 = \varphi_0 - \hat{\varphi}_0$ et $\frac{1}{2\beta} \tilde{\theta}_0^T \tilde{\theta}_0 \leq \frac{1}{\beta} \tilde{\theta}_0^T \tilde{\theta}_0$, alors

$$\begin{aligned} \dot{V}_0(t) = & \tilde{\varphi}_0(t)e_0^T(t)sgn(e_0(t)) - e_0^T(t)Pe_0(t) + \xi(x_0(t))|N(h_0(t))|\dot{h}_0(t) \\ & + \dot{h}_0(t) + \frac{1}{\beta}\tilde{\theta}_0^T\tilde{\theta}_0 \end{aligned} \quad (\text{III.34})$$

À partir de (III.21) avec $k=0$, on a

$$\tilde{\theta}_0(t) = -\beta e_0^T(t)sgn(e_0(t))\phi(x_0) \quad (\text{III.35})$$

En remplaçant (III.35) dans (III.34), on trouve

$$\begin{aligned} \dot{V}_0(t) = & \tilde{\varphi}_0(t)e_0^T(t)sgn(e_0(t)) - e_0^T(t)Pe_0(t) + \xi(x_0(t))|N(h_0(t))|\dot{h}_0(t) \\ & + \dot{h}_0(t) - \frac{1}{\beta}\left(\beta e_0^T(t)sgn(e_0(t))\phi(x_0)\right)^T\tilde{\theta}_0 \end{aligned} \quad (\text{III.36})$$

Tant que $\tilde{\varphi}_0 = \tilde{\theta}_0^T\phi(x_0)$, alors

$$\dot{V}_0(t) \leq \int_0^t \left(-e_0^T(t)Pe_0(t) + (\xi(x_0(t))|N(h_0(t))| + 1)\dot{h}_0(t)\right) dt \quad (\text{III.37})$$

En appliquant le théorème de Rayleigh-Ritz, (III.37) devient

$$\dot{V}_0(t) \leq \int_0^t \left(-\lambda_{\min}(P)\|e_0(t)\|^2 + (\xi(x_0(t))|N(h_0(t))| + 1)\dot{h}_0(t)\right) dt \quad (\text{III.38})$$

Donc

$$\dot{V}_0(t) \leq \int_0^t (\xi(x_0(t))|N(h_0(t))| + 1)\dot{h}_0(t) dt \quad (\text{III.39})$$

A partir de Lemme III.1 avec, $c_0 = 0$ et $c_1 \cong 0$, on a $\dot{V}_0(t)$ est bornée. Par conséquent, $V_0(t)$ est bornée $\forall t \in [0, T]$.

Partie 3 : La fonction $V_k(t)$ peut être réécrite comme suit

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_j(t) \quad (\text{III.40})$$

En utilisant (III.28), (III.40) devient

$$\begin{aligned}
V_k(t) \leq V_0(t) + \sum_{j=1}^k \int_0^t \left(-\lambda_{\min}(P) \|e_k(t)\|^2 \right. \\
\left. + (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) \right) d\tau
\end{aligned} \tag{III.41}$$

Ce qui nous donne

$$\begin{aligned}
\sum_{j=1}^k \int_0^T \|e_k(t)\|^2 d\tau \\
\leq \frac{1}{\lambda_{\min}(P)} \left(V_0(t) - V_k(t) \right. \\
\left. + \sum_{j=1}^k \int_0^T (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \right)
\end{aligned} \tag{III.42}$$

D'autre part, le dernier terme de (III.42) peut être réécrit comme suit

$$\begin{aligned}
\sum_{j=1}^k \int_0^T (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \\
= \int_{h_1(0)}^{h_1(T)} (\xi(x_k) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \\
+ \int_{h_2(0)}^{h_2(T)} (\xi(x_k) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau + \dots \\
+ \int_{h_k(0)}^{h_k(T)} (\xi(x_k) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau
\end{aligned} \tag{III.43}$$

A partir de (III.11), on obtient

$$\begin{aligned}
& \sum_{j=1}^k \int_0^T (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \\
&= \int_{h_1(0)}^{h_1(T)} (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \\
&+ \int_{h_1(T)}^{h_2(T)} (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau + \dots \\
&+ \int_{h_{k-1}(T)}^{h_k(T)} (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau
\end{aligned} \tag{III.44}$$

Alors

$$\begin{aligned}
& \sum_{j=1}^k \int_0^T (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau \\
&= \int_{H_1(0)}^{H_k(T)} (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau
\end{aligned} \tag{III.45}$$

A partir de lemme III.1 avec $c_0 = 0$ et $c_1 \cong 0$, on a $\sum_{j=1}^k \int_0^T (\xi(x_0(t)) |N(h_k(t))| + 1) \dot{h}_k(t) d\tau$ est bornée.

Sachant que, $V_0(t)$ et $V_k(t)$ sont bornées, de (III.42) il est clair que

$$\sum_{j=1}^k \int_0^T \|e_k(t)\|^2 d\tau \leq M \tag{III.46}$$

Où M est une constante finie. Ainsi,

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \forall t \in [0, T] \tag{III.47}$$

III.4 Résultats de simulation

III.4.1 Exemple d'un robot manipulateur

Considérons un robot manipulateur à deux degrés de liberté. Son modèle dynamique peut être écrit par [TAY 04] :

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} C_{11}\dot{q}_1 + C_{12}\dot{q}_2 + G_1 - d_1 \\ C_{21}\dot{q}_1 + C_{22}\dot{q}_2 + G_2 - d_2 \end{bmatrix} + \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \right\} \quad (\text{III.48})$$

Avec :

$$\begin{aligned} M_{11} &= m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_2)) + I_1 + I_2, \\ M_{12} &= M_{21} = m_2 (l_{c2}^2 + l_1 l_{c2} \cos(q_2)) + I_2, \\ M_{22} &= m_2 l_{c2}^2 + I_2, C_{11} = h\dot{q}_2, C_{12} = h\dot{q}_1 + h\dot{q}_2, C_{21} = -h\dot{q}_1, \\ C_{22} &= 0, h = -m_2 l_1 l_{c2} \sin(q_2), \\ G_1 &= (m_1 l_{c1} + m_2 l_1) g \cos(q_1) + m_2 l_{c2} g \cos(q_1 + q_2), \\ G_2 &= m_2 l_{c2} g \cos(q_1 + q_2). \end{aligned}$$

Les paramètres du robot sont donnés par

$$m_1 = m_2 = 1 [\text{Kg}], l_{c1} = l_{c2} = 0.25 [\text{m}], \quad l_1 = l_2 = 0.5 [\text{m}], \quad I_1 = I_2 = 0.1 [\text{Kg} \cdot \text{m}^3], \quad g = 9.81 \left[\frac{\text{m}}{\text{s}^2} \right]$$

Le vecteur de perturbations est donné par

$$\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} 2 + 5\pi t \\ 40 + \frac{t}{5} \end{bmatrix} \quad (\text{III.49})$$

Le vecteur des trajectoires désirées est donné par

$$\begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} = \begin{bmatrix} \sin(2\pi t) \\ \cos(2\pi t) \end{bmatrix}; t \in [0, 2] \quad (\text{III.50})$$

En appliquant la loi de commande proposée (III.11) avec la loi d'adaptation (III.6) avec (III.4) et (III.5), où $P = 6I_{2 \times 2}$, $\beta = 4$. Les paramètres du réseau neuronal RBF sont choisis comme $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 0.25$, $C_1 = 0.5$, $C_2 = 1.5$, $C_3 = -0.5$ et $C_4 = -1.5$. Pour l'architecture de RBF on a utilisé deux neurones dans la couche d'entrée, quatre neurones dans la couche cachée et un neurone dans la couche de sortie. Les paramètres sont choisis par essai et erreur. En utilisant MATLAB, les résultats de simulation à travers les itérations sont présentés dans les figures III.2, III.4, III.6 et III.8 pour la première articulation, et dans les figures III.3, III.5, III.7 et III.9 pour la deuxième articulation. Les figures III.2 et III.3 présentent les trajectoires réelles et désirées de la 1^{ère}, 2^{ème}, 7^{ème} et 20^{ème} itération pour la première et la deuxième articulation, respectivement. On peut voir que les trajectoires réelles suivent les trajectoires désirées à travers les itérations, pour les deux articulations. Les erreurs de la première et de la deuxième articulation à travers les itérations sont présentées dans les figures III.4 et III.5, respectivement. Il est clair que les erreurs diminuent à travers les itérations. Les figures III.6 et III.7 présentent respectivement la norme-Sup des erreurs à travers les itérations. On constate qu'après 20 itérations la norme-Sup de la première et la deuxième articulation se dégradent jusqu'à 0.035

et 0.088, respectivement. Les figures III.8 et III.9 présentent les signaux de commande pour les deux articulations. Dans laquelle, on a remplacé la fonction signe par la fonction tangente hyperbolique pour réduire le problème de chattering. A partir de ces résultats, nous pouvons conclure que la méthode proposée fonctionne bien.

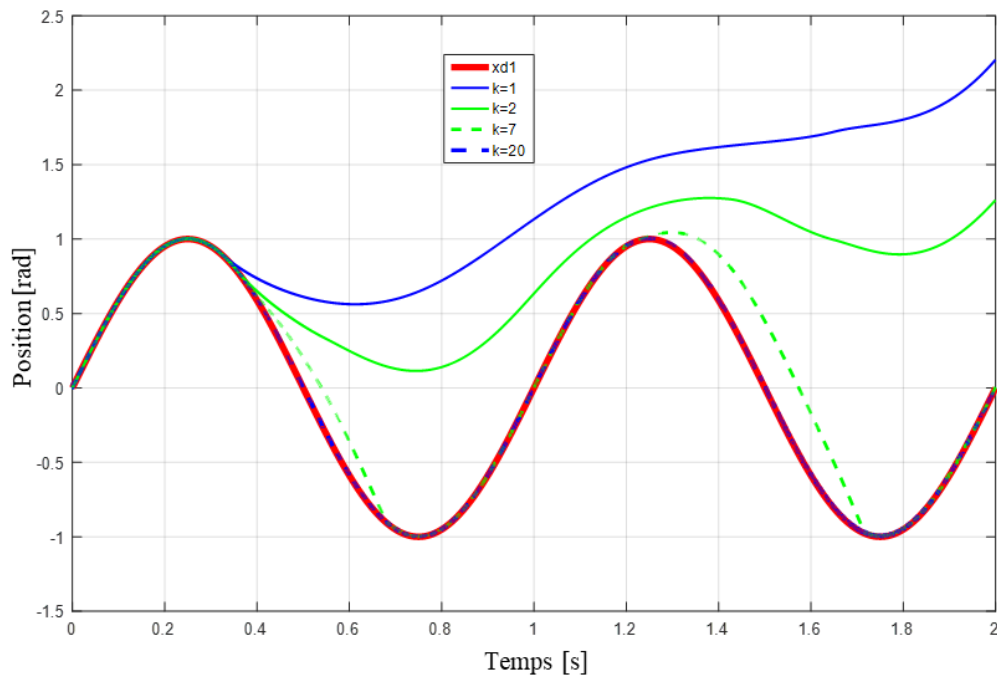


Figure III.2 Trajectoires réelles et désirée de la première articulation.

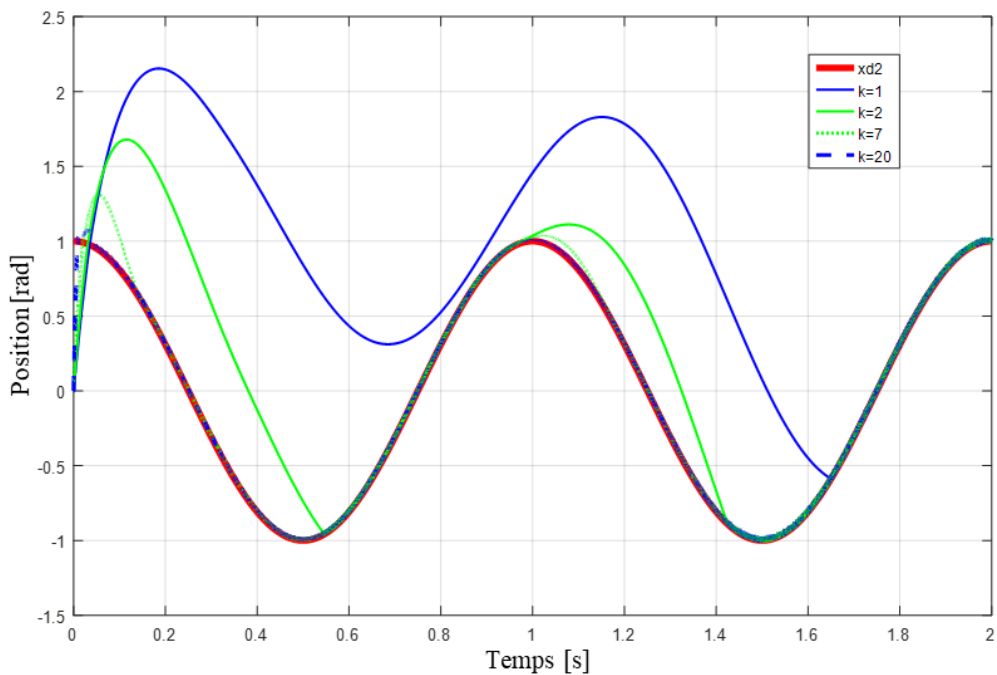


Figure III.3 Trajectoires réelles et désirée de la deuxième articulation.

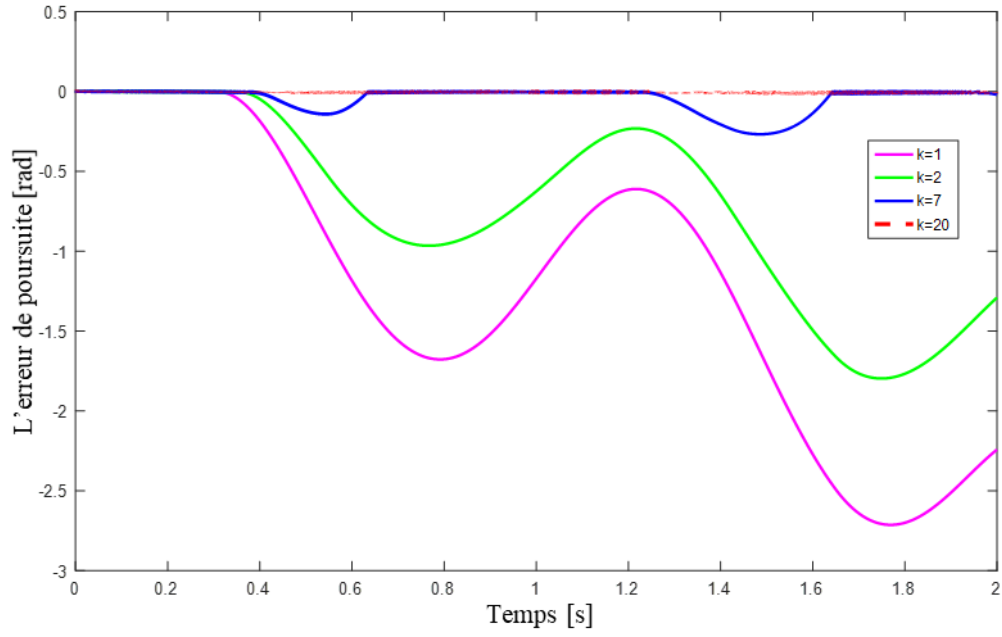


Figure III.4 Erreurs de poursuite de la première articulation.

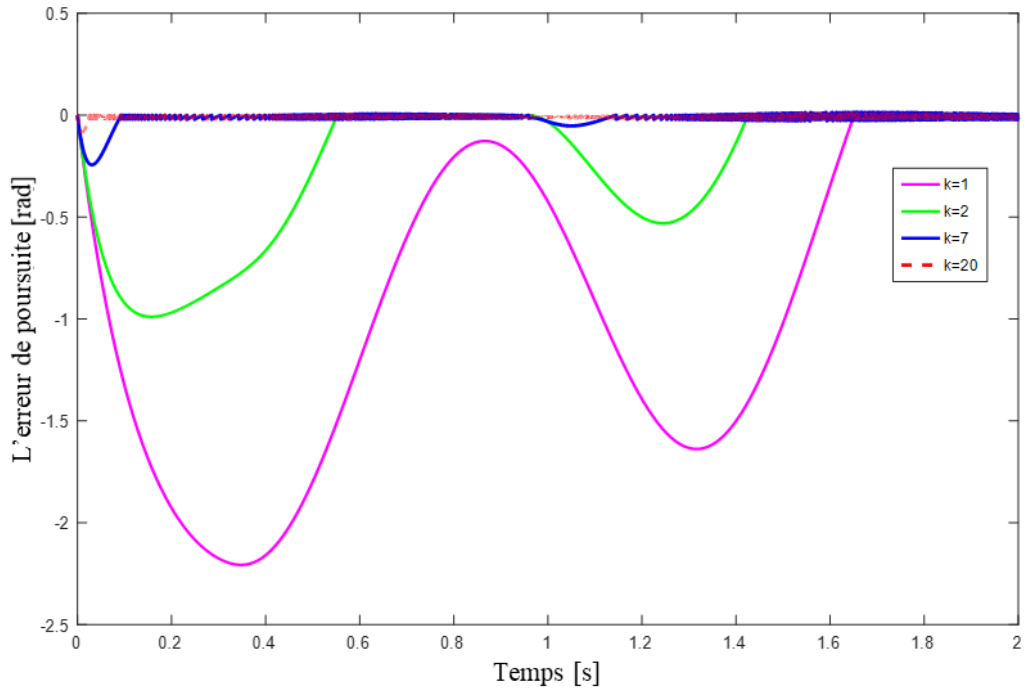


Figure III.5 Erreurs de poursuite de la deuxième articulation.

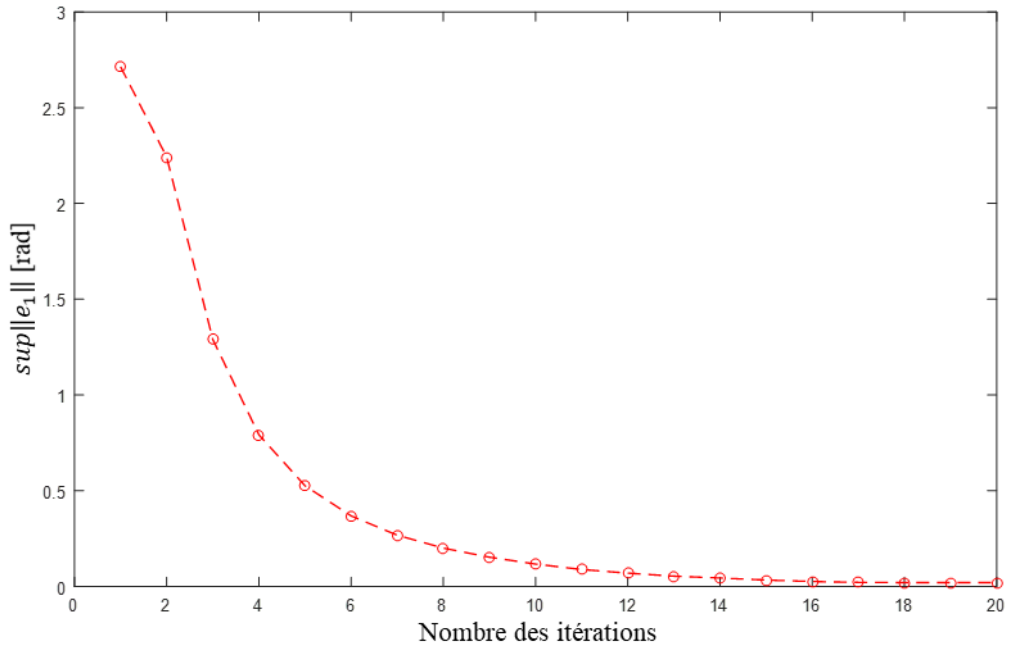


Figure III.6 Norme -Sup de l'erreur de la première articulation au cours des itérations.

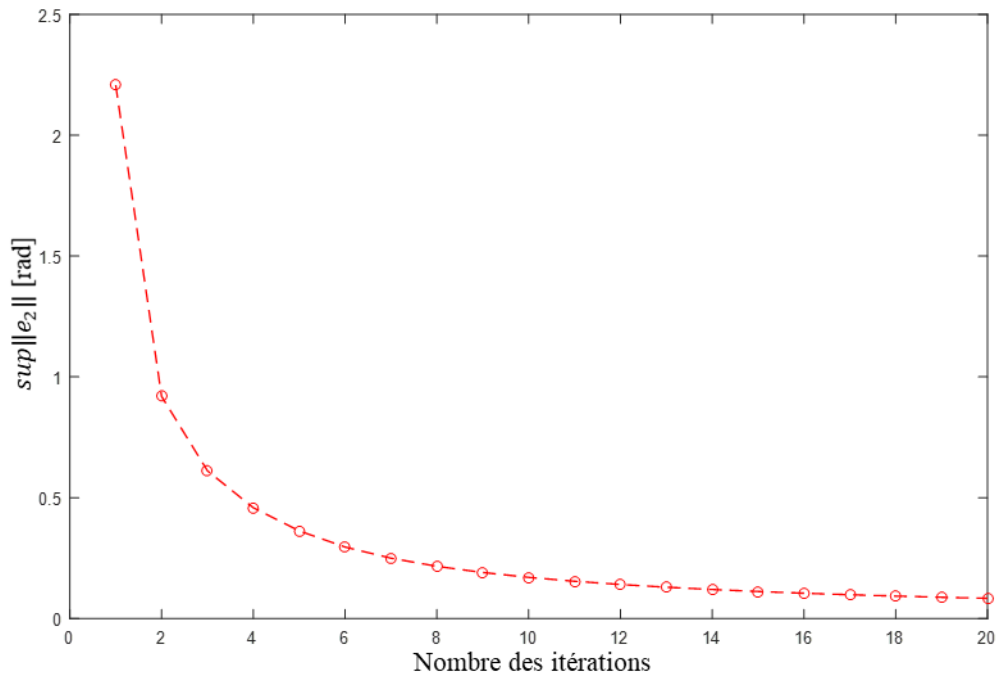


Figure III.7 Norme sup de l'erreur de la deuxième articulation au cours des itérations.

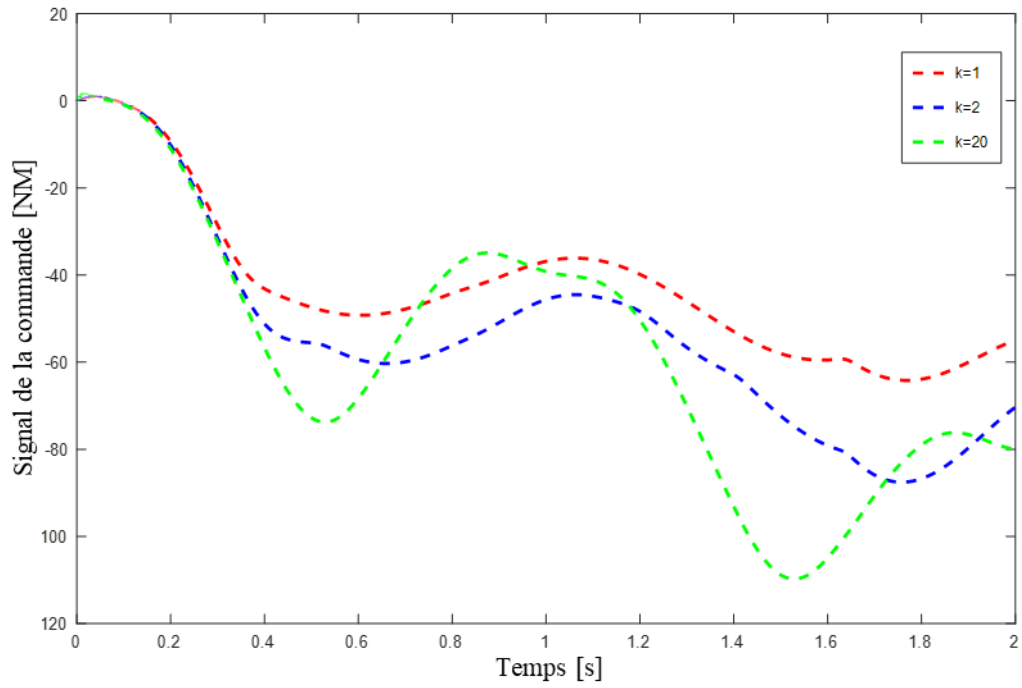


Figure III.8 Signal de commande pour la première articulation.

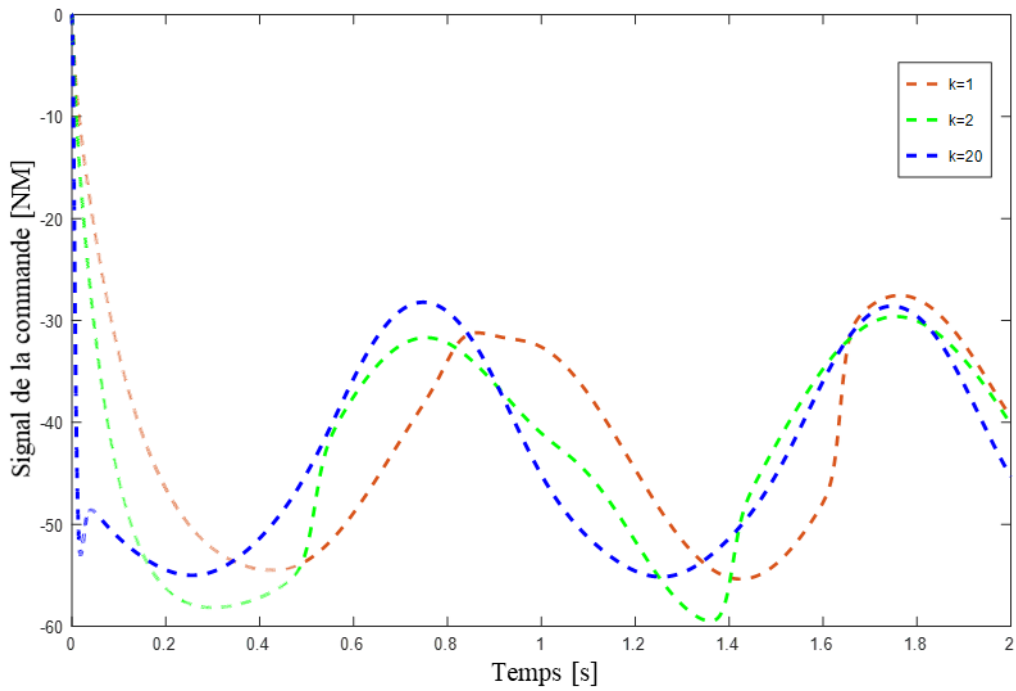


Figure III.9 Signal de commande pour la deuxième articulation.

III.4.2 Comparaison avec la commande robuste par apprentissage itératif

Dans cette section, l'exemple de simulation utilisé dans [LI 16] est adopté pour faire une comparaison avec notre méthode proposée. Le modèle d'état du système non linéaire est donné par

$$\begin{aligned} & \begin{bmatrix} \dot{x}_{1k} \\ \dot{x}_{2k} \end{bmatrix} \\ &= \begin{bmatrix} x_{1k} + \sin(x_{2k}) \\ 2 + \sin(x_{1k}) + x_{2k} \end{bmatrix} \\ &+ \begin{bmatrix} 2 + \sin(x_{1k}) + 0.1 \sin(2\pi t) & 0 \\ 0 & 1 + \sin(x_{2k}) + 0.1 \sin(4\pi t) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \end{aligned} \quad (\text{III.51})$$

Le vecteur des trajectoires désirées est défini comme

$$\begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \sin(8\pi t) \\ \frac{1}{4} \cos(16\pi t) \end{bmatrix} \quad t \in [0, 0.5] \quad (\text{III.52})$$

En appliquant la loi de commande proposée (III.11), la loi d'adaptation (III.6) avec (III.4) et (III.5). Les paramètres de la commande et l'architecture de RBF sont choisis comme dans la section précédente.

Les résultats de simulation à travers les itérations sont présentés dans les figures III.10 et III.12 pour la première articulation, et dans les figures III.11 et III.13 pour la deuxième articulation. Les figures III.10 et III.11 présentent respectivement les trajectoires réelles et désirées de la 1^{ère}, 2^{ème} et 7^{ème} itération pour la première et la deuxième articulation, respectivement. Nous pouvons observer que les trajectoires réelles suivent les trajectoires désirées à travers les itérations, pour les deux articulations. Les figures III.12 et III.13 présentent les signaux de commande pour les deux articulations à la 1^{ème}, 4^{ème} et 7^{ème} itération.

D'autre part, la loi de commande robuste par apprentissage itératif qui a été proposée dans [LI 16] est donnée par :

$$\begin{cases} u_k(t) = B_k^{-1} [-\Gamma e_k(t) - \hat{\theta}_k(t) \rho(x_k, t) \text{sgn}(e_k(t))] \\ \hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + \gamma \rho(x_k, t) \sum_{i=1}^n |e_{i,k}(t)| \end{cases} \quad (\text{III.53})$$

où $(x_k, t) = \sqrt{(f_1^2 + f_2^2)}$, $\Gamma = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$ et $\hat{\theta}_0(t) = 0$

Maintenant, pour faire la comparaison avec cette méthode, nous avons utilisé la norme-Sup de l'erreur de poursuite du système en boucle fermée ($\sup |e(t)|$) comme un indice de comparaison d'évolution des performances du système (III.51). Dans laquelle la valeur de P est choisi comme celle de $\Gamma = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$. Les figures III.14 et III.15 présentent la norme-Sup de l'erreur de poursuite à travers les itérations pour la première et la deuxième articulation, respectivement. Pour bien illustrer la comparaison, les valeurs de la norme-Sup de la première et la deuxième erreur de poursuite pour les deux méthodes à travers les itérations sont bien détaillées dans les tableaux III.1 et III.2, respectivement.

A partir de ces résultats, on peut voir que le taux de convergence de l'erreur de poursuite de notre loi de commande proposée est plus rapide par rapport à la commande robuste par apprentissage itératif, on constate qu'on a besoin seulement à 10 itérations pour obtenir une bonne performance, telles que la norme-Sup de l'erreur se dégrade jusqu'à 0,0076 pour la première articulation, et jusqu'à 0,0008 pour la deuxième articulation. Par conséquent, on peut conclure que notre commande est plus puissante en termes de rapidité de convergence.

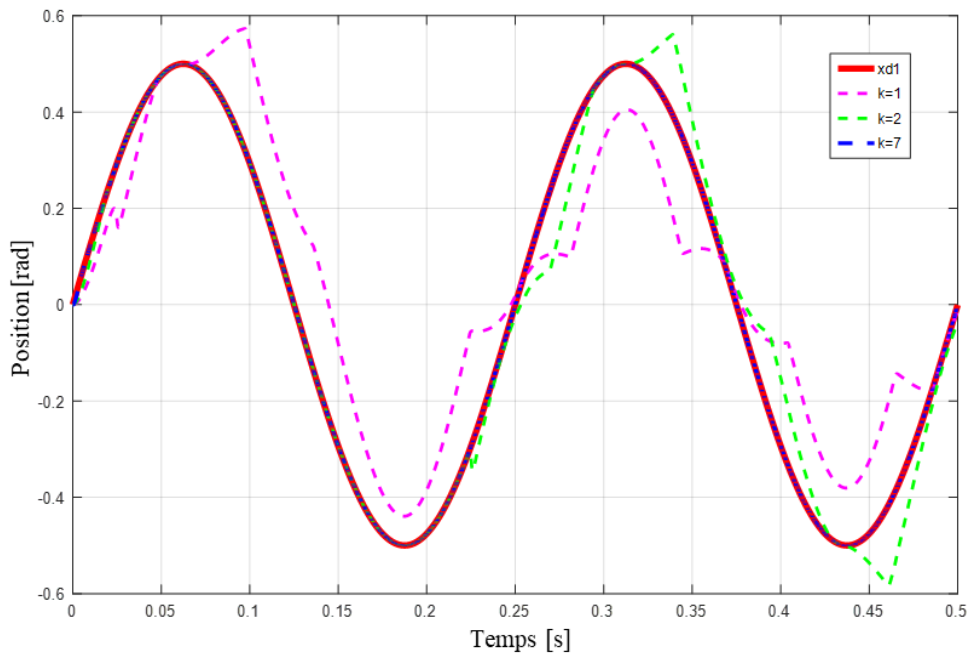


Figure III.10 Trajectoires réelles et désirée de la première articulation.

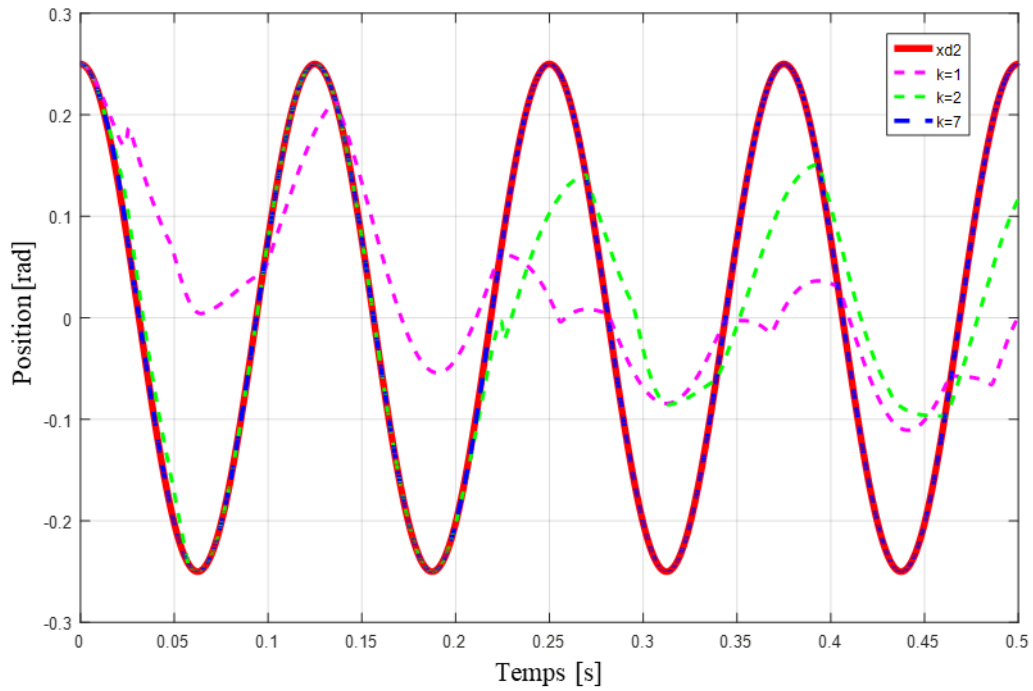


Figure III.11 Trajectoires réelles et désirée de la deuxième articulation.

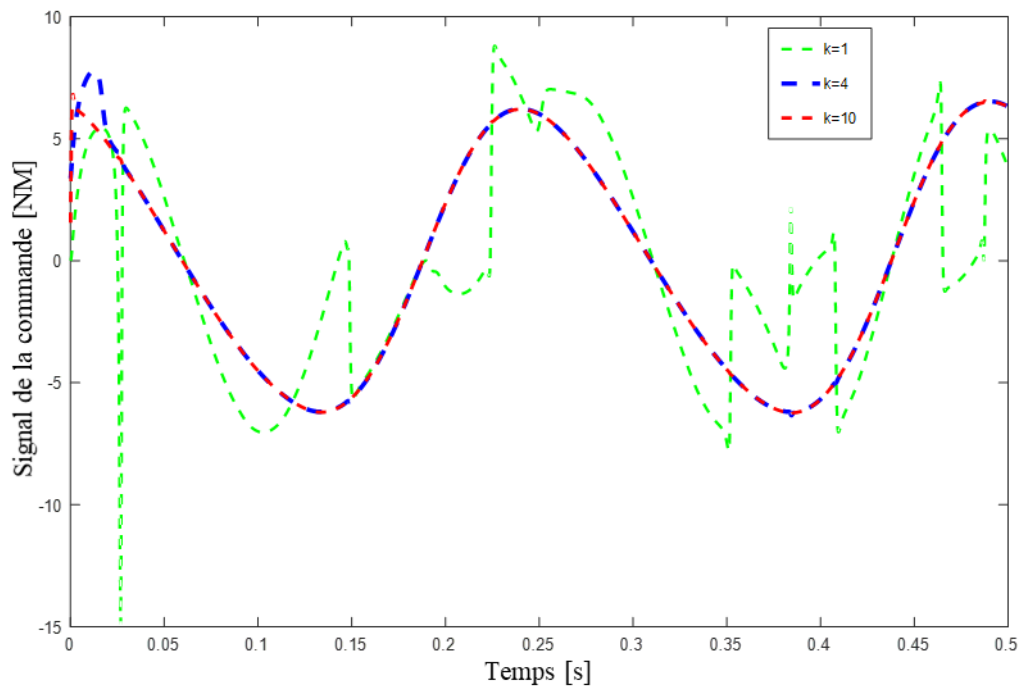


Figure III.12 Signal de commande pour la première articulation.

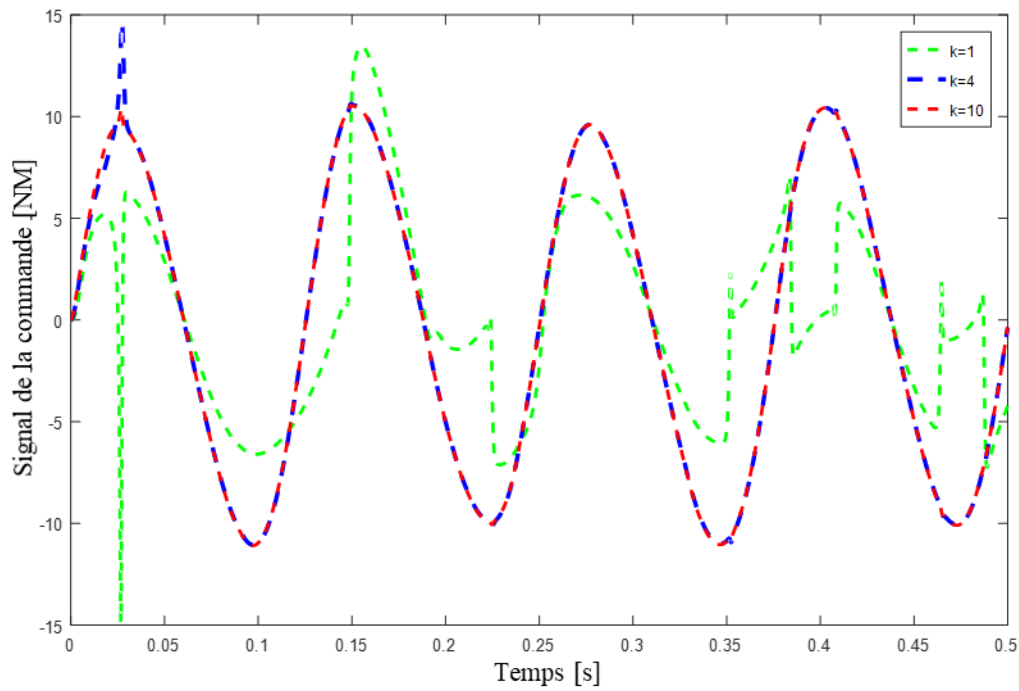


Figure III.13 Signal de commande pour la deuxième articulation.

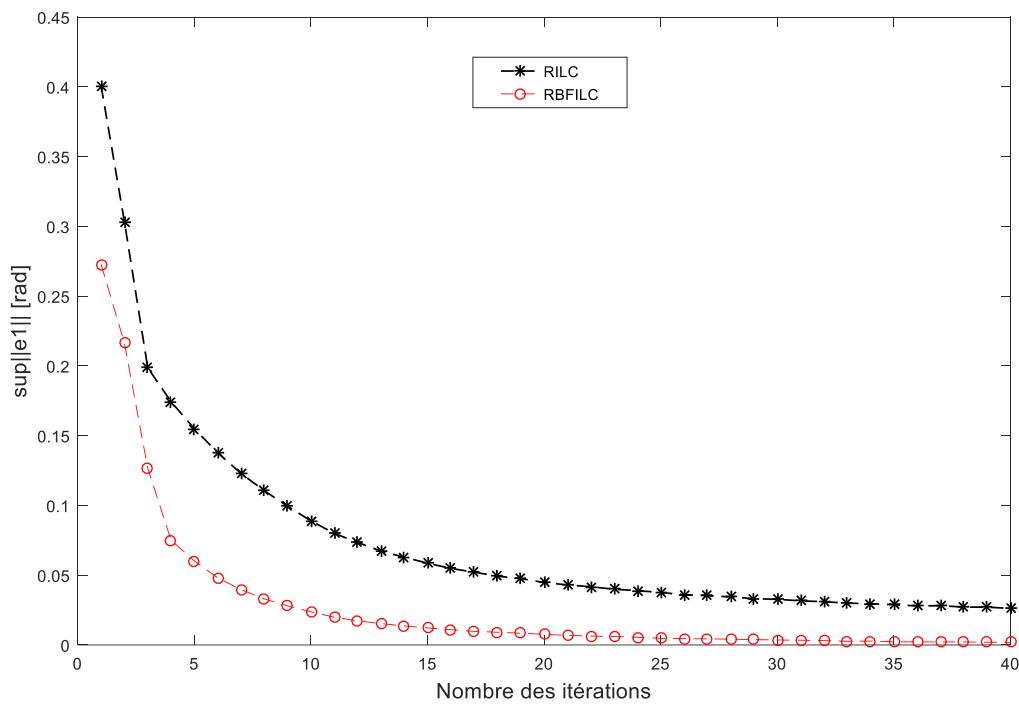


Figure III.14 Norme Sup de l'erreur de la première articulation au cours des itérations pour les deux méthodes.

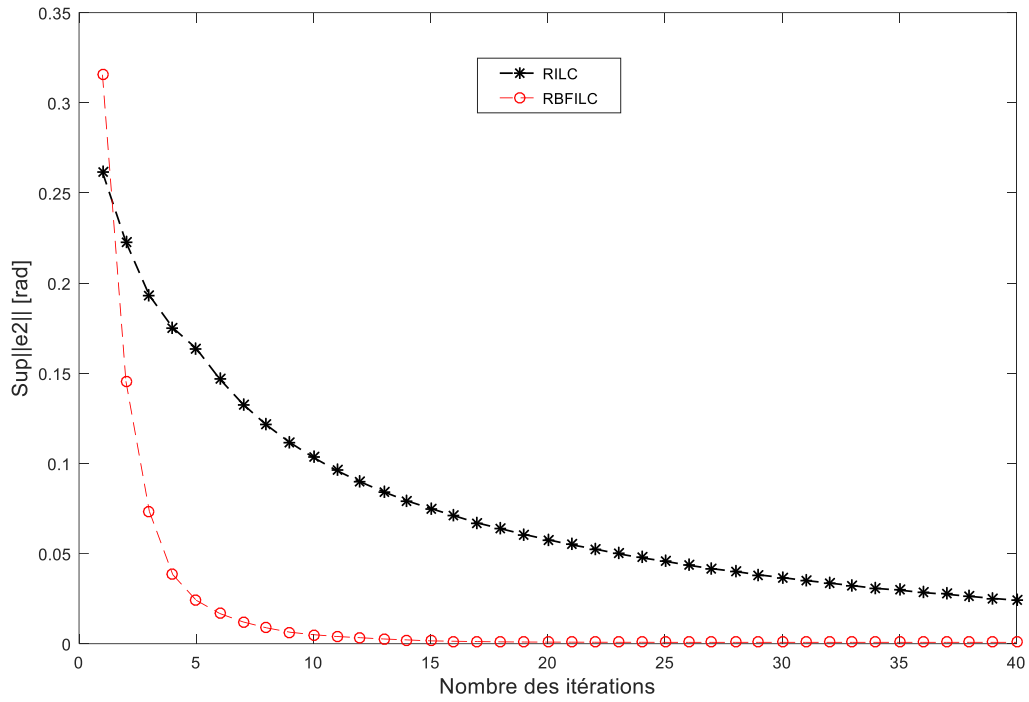


Figure III.15 Norme Sup de l'erreur de la deuxième articulation au cours des itérations pour les deux méthodes.

Table III.1 Valeurs de $\sup \| e1 \|$ pour les deux méthodes (1^{ère} articulation).

	1 ^{ère} itération	2 ^{ème} itérations	7 ^{ème} itérations	10 ^{ème} itérations	20 ^{ème} itérations
Erreurs (RILC)	0.4007	0.3034	0.1299	0.0889	0.043
Erreurs (RBFILC)	0.2727	0.2164	0.0396	0.0238	0.0076

Table III.2 Valeurs de $\sup \| e2 \|$ pour les deux méthodes (2^{ème} articulation).

	1 ^{ère} itération	2 ^{ème} itérations	7 ^{ème} itérations	10 ^{ème} itérations	20 ^{ème} itérations
Erreurs (RILC)	0.2616	0.2228	0.1328	0.1033	0.0577
Erreurs (RBFILC)	0.3155	0.1456	0.0121	0.005	0.0009

III.5 Conclusion

Dans ce chapitre, le problème de poursuite de trajectoires des systèmes non linéaire répétitifs a été traité, d'où, nous avons proposé une loi de commande par apprentissage itératif à base des réseaux de neurones à fonctions de base radiales (RBF). La particularité de ce schéma de commande c'est qu'il est applicable aux systèmes non linéaires multivariables sans la nécessité de satisfaire la condition de continuité globale de Lypschitz et sans connaître à priori la direction de la commande (le signe de la fonction de commande). En utilisant la méthode de Lyapunov, la stabilité du système en boucle fermée est démontrée. Des résultats de simulations sur un robot manipulateur à deux degrés de liberté sont effectués. La comparaison de ces résultats avec d'autre méthode de commande a permis de montrer clairement l'efficacité de notre commande proposée.

Chapitre. IV.

Commande par apprentissage itératif des systèmes non linéaires en présence des non linéarités à leurs entrées

IV.1 Introduction

Dans le monde industriel, la majorité des systèmes dynamiques sont soumis à des incertitudes et des limitations sur leurs fonctionnement délivrées par les actionneurs comme la zone-morte, saturation et hystérésis. En réalité, ces contraintes peuvent exercer une forte dégradation des performances des systèmes et en même temps peuvent détruire et déstabiliser le système en boucle fermée. Pour cela, la prise en compte de ces non linéarités lors de la conception d'une commande devient une autre direction avec une grande importance dans le développement des schémas de commande.

Récemment, A l'heure actuelle, la commande des robots manipulateurs constitue l'un des axes majeurs et prépondérants dans le domaine de recherche en robotique. En effet, ces types des systèmes fonctionnent de manière répétitive et la même erreur sera automatiquement générée à chaque opération. Pour cela, la commande par apprentissage itératif est appropriée pour commande ce genre de systèmes à cause de sa capacité d'améliorer les performances du système à travers les itérations. En fait, en utilisant cette approche itérative, plusieurs applications industrielles sur les robots manipulateurs ont été faites, par exemple, les travaux de [HAK 08], [PAS 13], [TAY 06] et [WAN 18].

Dans la littérature, plusieurs travaux sur la CAI des systèmes dynamique SISO sont réalisés pour traiter le problème de présence du non linéarité zone-morte à l'entrée [TAN 08], [TAN 12] et [WEI 14]. À travers notre recherche nous avons constaté qu'il y a peu de travaux sur la CAI qui ont traité ce problème pour les systèmes non linéaires multivariables, tels que les robots

manipulateurs. Ceci nous a motivé d'orienter notre recherche vers cet axe et de proposer une loi de commande CAI pour résoudre le problème de poursuite des robots manipulateurs sous la contrainte de la zone morte à l'entrée.

D'autre part, à travers notre recherche nous avons constaté que, il n'y pas beaucoup des travaux sur la CAI qui prennent en compte la présence de la contrainte de la saturation à l'entrée, en raison de la difficulté rencontrée pour démontrer la stabilité du système en boucle fermée. D'où, des lois de CAI sont présentées pour traiter ce problème sur des systèmes SISO [YUQ 16], [ZHA 18] et [YAN 19], et pour des systèmes linéaires [SEB 19]. Dans ce chapitre, nous proposons une loi de CAI pour résoudre le problème de saturation à l'entrée pour des systèmes non linéaires multivariables (MIMO).

Il est à noter que, la commande par apprentissage itératif peut être classée en deux types : CAI d'ordre un et CAI d'ordre élevé, suivant le nombre des informations des itérations utilisée. Il, il est a noté que la CAI d'ordre élevé est meilleure en termes de performances obtenues [BOU 14] et [CHI 181]. En outre, suivant le type des erreurs utilisées dans la commande, on peut distinguer quatre types de CAI : -P-Type [BEN 20], -D-Type [BOU 111], -PD-Type [TAY 04] et -PID-Type [MAD 08]. Dans les applications pratiques, l'accumulation de l'action dérivée à travers les itérations peut générer des bruits, pour cela, l'action proportionnelle est préférée dans la CAI.

Dans ce chapitre, deux approches de commandes par apprentissage itératif, sont proposées pour étudier et surmonter le problème d'existence des non linéarités à l'entrée des systèmes répétitifs. Dans le premier schéma, nous avons présenté une loi de commande adaptative par apprentissage itératif de type -P- à base des réseaux de neurones à fonction de base radiales (RBF) [BEN 20] pour résoudre le problème de poursuite des trajectoires des robot manipulateurs possédants des non-linéarités de type zone-morte à leurs entrées et soumis à des perturbations externes non répétitives. Dans la deuxième approche, une loi de commande par apprentissage itératif de Type -P- d'ordre élevée est proposée pour traiter le problème de poursuite des systèmes non linéaires multivariables MIMO en présence des saturations à leurs entrées.

IV.2 Commande adaptative par apprentissage itératif type-P des robots manipulateurs avec zone-morte à l'entrée

IV.2.1 Formulation du problème

IV.2.1.1 Dynamique des robots manipulateurs

Basant sur les équations de mouvement de Lagrange, le modèle dynamique d'un robot manipulateur à n degrés de liberté avec zone-morte à l'entrée peut s'écrire

$$\begin{cases} u_k(t) = M(q_k)\ddot{q}_k + C(q_k, \dot{q}_k)\dot{q}_k + G(q_k) + d_k(t) \\ u_k(t) = DZ(\tau_k(t)) \end{cases} \quad (\text{IV.1})$$

où t est le temps, k : indice indique le nombre des itérations.

$q_k(t) \in R^n$, $\dot{q}_k(t) \in R^n$ et $\ddot{q}_k(t) \in R^n$ sont des vecteurs de la position, la vitesse et l'accélération des articulations, respectivement. $M(q_k) \in R^{n \times n}$ est la matrice d'inertie, $C(q_k, \dot{q}_k) \in R^{n \times n}$ représente la matrice de forces centrifuges et de Coriolis, $G(q_k) \in R^n$ est un vecteur de forces gravitationnelles. $\tau_k(t) \in R^n$ est le vecteur des couples de l'entrée.

$DZ(\tau_k(t)) \in R^n$ est le vecteur des non-linéarités aux entrées de type zone morte supposé inconnu. $d_k(t) \in R^n$ est le vecteur des perturbations externes supposées inconnues.

Le système (IV.1) a les propriétés suivantes.

Propriété IV.1. La matrice d'inertie $M(q_k)$ est symétrique, définie positive et bornée comme suit

$$0 < \beta_1 < \|M(q_k)\| < \beta_2 \quad (\text{IV.2})$$

Où $q_k \in R^n$ et $\beta_2 > \beta_1 > 0$.

Propriété IV.2. La norme de $G(q_k)$ est bornée comme suit

$$\|G(q_k)\| < l_g \quad (\text{IV.3})$$

où l_g est une constant positive.

Propriété IV.3. La norme de la matrice des forces de centrifuges et Coriolis est bornée comme suit

$$\|C(q_k, \dot{q}_k)\| \leq C_m \|\dot{q}_k\| \quad (\text{IV.4})$$

où C_m est une constante positive.

IV.2.1.2 Analyse de la zone-morte

La zone-morte est l'un des types des non-linéarités les plus courantes qui existent dans les articulations des robots manipulateurs. La présence de ce type peut générer des oscillations non contrôlables et peut déstabiliser le système et détruire ses performances. La figure suivante présente le phénomène de la zone-morte.

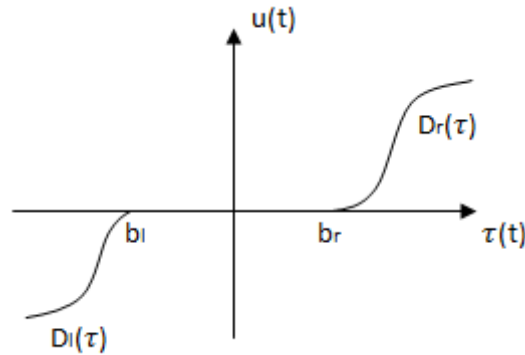


Figure IV.1 Fonction de la zone-morte.

Le modèle dynamique de ce type des non linéarités est d'écrire la relation entre l'entrée de la zone-morte $\tau_k(t)$ et sa sortie $u_k(t)$ qui est donnée comme suite :

$$u_k(t) = DZ(\tau_k(t)) = \begin{cases} D_r(\tau_k(t)) & \tau_k \in]b_r, \infty[\\ 0 & \tau_k \in [b_l, b_r] \\ D_l(\tau_k(t)) & \tau_k \in]-\infty, b_l[\end{cases} \quad (\text{IV.5})$$

où $D_r(\tau_k(t))$ et $D_l(\tau_k(t))$ sont des fonctions non linéaires inconnues.

Pour simplifier la conception du système de commande, le modèle de la zone-morte peut être représenté sous forme de combinaison d'un terme linéaire (avec gain variant) et d'un terme de perturbation. Sachant que, les fonctions $D_r(\tau_k(t))$ et $D_l(\tau_k(t))$ sont continues sur $]b_r, \infty[$ et sur $]-\infty, b_l[$, il existe des constantes positives inconnues d_{l0}, d_{l1}, d_{r0} et d_{r1} tels que

$$0 < d_{l0} \leq D'_l(\tau_k) \leq d_{l1} \text{ for } \tau_k \in]-\infty, b_l[\quad (\text{IV.6})$$

$$0 < d_{r0} \leq D'_r(\tau_k) \leq d_{r1} \text{ for } \tau_k \in]b_r, \infty[\quad (\text{IV.7})$$

où $D'_l(\tau_k) = \left. \frac{d(D_l(z))}{dz} \right|_{z=\tau} = l, r$ est la dérivée de $D_i(\tau)$, $i = l, r$

Sachant que $D_r(b_r) = D_l(b_l) = 0$ et en appliquant le théorème différentiel des valeurs moyennes sur $D_r(\tau_k(t))$ et $D_l(\tau_k(t))$, il vient

$$D_l(\tau) = D_l(\tau) - D_l(b_l) = D'_l(\xi_l)(\tau - b_l), \forall \tau_k \in]-\infty, b_l[\text{ et } \xi_l \in]\tau, b_l[\quad (\text{IV.8})$$

$$D_r(\tau) = D_r(\tau) - D_r(b_r) = D'_r(\xi_r)(\tau - b_r), \forall \tau_k \in]b_r, +\infty[\text{ et } \xi_r \in]b_r, \tau[\quad (\text{IV.9})$$

Supposons que

$$m_r(t) = D'_r(\xi_r) \quad (\text{IV.10})$$

$$m_l(t) = D'_l(\xi_l) \quad (\text{IV.11})$$

Par conséquent, la zone-morte DZ (*) peut être représentée comme suit

$$DZ(\tau_k(t)) = \begin{cases} m_{r,k}(t)(\tau_k(t) - b_{r,k}(t)) & \tau_k \in]b_r, \infty[\\ 0 & \tau_k \in [b_l, b_r] \\ m_{l,k}(t)(\tau_k(t) - b_{l,k}(t)) & \tau_k \in]-\infty, b_l[\end{cases} \quad (\text{IV.12})$$

avec $m_{r,k}(t)$ et $m_{l,k}(t)$ sont des fonctions inconnues, avec $m_{r,k}(t) > 0$, $m_{l,k}(t) > 0$, $b_{r,k}(t) > 0$ et $b_{l,k}(t) < 0$.

Dans ce chapitre, l'hypothèse suivante sur la zone-morte est considérée.

Hypothèse IV.1 : $m_{r,k}(t)$, $m_{l,k}(t)$, $b_r(t)$ et $b_l(t)$ sont bornées par des fonctions inconnues. En plus, $\frac{d}{dt}(m_{r,k}(t))$ et $\frac{d}{dt}(m_{l,k}(t))$ sont bornées.

Remarque IV.1 : l'hypothèse IV.1 est largement utilisée dans la littérature, par exemple [ZHA 07]

L'équation (IV.12), peut être réécrite comme suit

$$u_k(t) = DZ(\tau_k(t)) = \begin{cases} m_r(t)\tau_k(t) - b_1(\tau_k(t)) & \tau_k \in [0, \infty[\\ m_l(t)\tau_k(t) - b_1(\tau_k(t)) & \tau_k \in]-\infty, 0] \end{cases} \quad (\text{IV.13})$$

Avec

$$b_1(\tau_k(t)) = \begin{cases} m_r(t)b_r & \text{pour } \tau_k \in]b_r, \infty[\\ m_r(t)\tau_k(t) & \text{pour } \tau_k \in [b_r, 0] \\ m_l(t)\tau_k(t) & \text{pour } \tau_k \in [0, b_l] \\ m_l(t)b_l & \text{pour } \tau_k \in]-\infty, b_l[\end{cases} \quad (\text{IV.14})$$

À partir de l'hypothèse IV.1 Il est claire que $b_1(\tau_k(t))$ est bornée.

Dans ce travail, les hypothèses suivantes sur le système sont nécessaires.

Hypothèse IV.2 : La trajectoire désirée $y_d(t)$ et sa première dérivée $\dot{y}_d(t)$ sont bornées.

Hypothèse IV.3 : A chaque itération, les conditions initiales sont les mêmes, c.à.d., $q_k(0) = q_a(0)$.

Remarque IV.2 : Dans la littérature, l'hypothèse 3 est largement utilisée dans la plupart des travaux (voir par exemple [BOU 18] et [GU 19]).

Hypothèse IV.4 : Les perturbations externes $d_k(t)$ sont inconnues et leurs normes sont bornées par des fonctions inconnues.

Hypothèse IV.5 : La vitesse du robot est bornée par une constante connue V_m

$$\|\dot{q}_k(t)\| \leq V_m \quad (\text{IV.15})$$

A partir de la propriété IV.1, le système (IV.1) peut être réécrit comme suit

$$\ddot{q}_k = -M^{-1}(q_k)[C(q_k, \dot{q}_k)\dot{q}_k + G(q_k) + d_k(t)] + M^{-1}(q_k)D(\tau_k(t)) \quad (\text{IV.16})$$

En utilisant (IV.13), (IV.16) peut être transformée à la forme suivante

$$\begin{cases} \dot{x}_k(t) = \phi(x_k(t), t) + m_*(t)\psi(x_k(t), t)\tau_k(t) \\ y_k(t) = x_k(t) \end{cases} \quad (\text{IV.17})$$

$$\text{avec } m_*(t) = \begin{cases} m_r(t) & \text{pour } \tau_k \in]-\infty, 0] \\ m_l(t) & \text{pour } \tau_k \in [0, \infty[\end{cases}$$

$$\text{où } x_k(t) = \begin{bmatrix} q_k(t) \\ \dot{q}_k(t) \end{bmatrix}, \psi(x_k(t), t) = \begin{bmatrix} 0 \\ M^{-1}(q_k) \end{bmatrix}, \phi(x_k(t), t) = \begin{bmatrix} \dot{q}_k(t) \\ -M^{-1}(q_k)[C(q_k, \dot{q}_k)\dot{q}_k + G(q_k) + d_k(t) + b_1(\tau_k(t))] \end{bmatrix}, x_k \in R^{2n}, \psi(x_k) \in R^{2n \times n}, \phi(x_k) \in R^{2n} \text{ et } y_k \in R^{2n}. b_1(\tau_k(t)) \text{ est définie dans l'équation (IV.14)}$$

À partir des propriétés IV.1-IV.3 et les hypothèses IV.1, IV.2, IV.4 et IV.5, il est clair que $\phi(x_k(t), t)$ est bornée par une fonction inconnue. Par conséquent

$$\left\| \text{+}K_p \text{+}\psi(x_k(t), t) \frac{1}{m_*} (\dot{y}_d(t) - \phi(x_k(t), t)) \right\| \leq \alpha(x_k(t), t) \quad (\text{IV.18})$$

avec $\text{+}K_p \in R^{2n \times n}$ est une matrice des gains, elle est définie comme suit

$$\text{+}\psi(x_k) = (\psi^T(x_k)\psi(x_k))^{-1}\psi^T(x_k) \quad (\text{IV.19})$$

$$\text{+}K_p = (K_p^T K_p)^{-1} K_p^T \quad (\text{IV.20})$$

Il est clair que $\mathcal{K}_p \mathcal{K}_p = I^{2n \times 2n}$ et $\psi(x_k) \mathcal{K}_p \psi(x_k) = I^{2n \times 2n}$.

Pour estimer la fonction non linéaire inconnue $\alpha(x_k(t), t)$, la fonction de base radiale (RBF) est utilisée, d'où

$$\alpha(x_k) = \theta_k^T \mu(x_k) + \epsilon \quad (\text{IV.21})$$

avec θ_k est le vecteur des paramètres optimaux et ϵ est l'erreur d'approximation supposée bornée et très petite. $\mu(x_k, t)$ est le vecteur connu de la fonction de base, ses éléments sont choisis comme suit

$$\mu_i(x_k(t)) = \exp\left(-\frac{\|x_k - C_i\|^2}{\sigma_i^2}\right) \quad (\text{IV.22})$$

Où $C_i \in R^l$ et σ_i sont le centre et le rayon d'influence de neurone, respectivement. $\hat{\theta}_k(t) \in R^l$ est le vecteur d'estimation de $\theta_k(t)$ à travers les itérations. La loi d'adaptation proposée est donnée par :

$$\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + e_k^T(t) \text{sgn}(e_k(t)) \mu(x_k, t) \quad (\text{IV.23})$$

avec $\hat{\theta}_{-1}(t) = 0$. L'erreur d'estimation est présentée comme suit :

$$\tilde{\theta}_k(t) = \theta - \hat{\theta}_k(t) \quad (\text{IV.24})$$

L'erreur de poursuite à la $k^{\text{ième}}$ itération est définie par :

$$e_k(t) = y_d(t) - y_k(t) \quad (\text{IV.25})$$

Par dérivation, on obtient

$$\dot{e}_k(t) = \dot{y}_d(t) - \dot{x}_k(t) \quad (\text{IV.26})$$

En remplaçant (IV.17) dans (IV.26), il vient

$$\dot{e}_k(t) = \dot{y}_d(t) - \phi(x_k(t), t) - m_*(t) \psi(x_k(t), t) \tau_k(t) \quad (\text{IV.27})$$

Objectif : Notre objectif est de développer une loi de commande adaptative par apprentissage itératif à base des réseaux de neurones $\tau_k(t)$ pour forcer la sortie réelle $y_k(t)$ à suivre une trajectoire désirée $y_d(t)$ sur un intervalle de temps fixe $t \in [0, T]$, lorsque k tend vers l'infini.

IV.2.2 Synthèse d'une loi de commande adaptative par apprentissage itératif

Dans cette section, nous proposons une loi de commande adaptative par apprentissage itératif de type -P- à base des réseaux de neurones à fonction de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des robots manipulateurs (IV.1). Elle est donnée comme suit :

$$\tau_k(t) = K_p [e_k(t) + \hat{\alpha}_k(x) \text{sgn}(e_k(t))] \quad (\text{IV.28})$$

où $K_p \in R^{n \times 2n}$ est une matrice symétrique des gains proportionnels, $\hat{\alpha}_k(x) \in R$ est l'estimation de la fonction $\alpha_k(x)$. Elle est donnée par

$$\hat{\alpha}(x_k) = \hat{\theta}_k^T(t) \mu(x_k) \quad (\text{IV.29})$$

avec $\mu(x_k, t)$ et $\hat{\theta}_k(t)$ sont définies dans (IV.22) et (IV.23), respectivement.

Le schéma bloc complet de la commande proposée est présenté par la figure suivante.

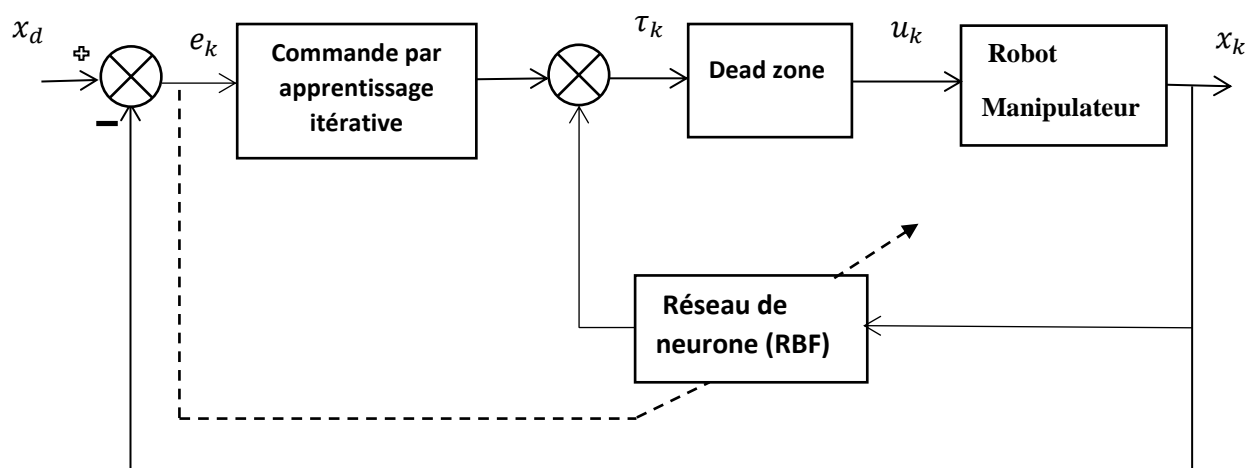


Figure IV.2 Diagramme fonctionnel de la commande proposée.

Théorème IV.1 [BEN 20] : Considérons le robot manipulateur (IV.1), vérifiant les hypothèses (IV.1-IV.5), en appliquant la loi de commande proposée (IV.28) avec la loi d'adaptation (IV.23-IV.29), Si

$$\lambda_{\min}(K_p) > \frac{C_m V_m}{\beta_1^2} \quad (\text{IV.30})$$

Alors

$$\text{iii) } e_k(t) \in L_{\infty e}[0, T], \hat{\theta}_k(x) \in L_{\infty e}[0, T]$$

$$\text{iv)} \quad \lim_{k \rightarrow \infty} e_k(t) = 0, \forall t \in [0, T].$$

Où $\lambda_{\min}(K_p)$ désigne la valeur propre minimale de K_p .

Preuve

On va montrer la stabilité de système en boucle fermée pour les deux cas suivants, le premier cas pour $\tau_k \in [0, \infty[$ et le deuxième cas pour $\tau_k \in]-\infty, 0]$.

1^{er} cas: $\tau_k \in [0, \infty[$; La preuve est divisée en trois parties, dans la première partie, on choisit une fonction de Lyapunov $V_k(t)$ définie positive et montrer que cette fonction reste décroissante à travers le nombre des itérations. En seconde partie, on démontre que $V_0(t)$ est bornée pour tout $t \in [0, T]$. À la fin, on montre que $\lim_{k \rightarrow \infty} e_k(t) = 0$ pour tout $t \in [0, T]$.

Dans le reste, les notations suivantes sont utilisées : $\psi(x_k(t), t) = \psi(x_k)$, $\phi(x_k(t), t) = \phi(x_k)$, $\theta_k(t) = \theta_k$, $\alpha(x_k, t) = \alpha_k$ et $e_k(t) = e_k$.

Partie 1 : Soit la fonction de Lyapunov suivante

$$V_k(t) = \frac{1}{2} \left(\frac{1}{m_r} e_k^T \right)^{\leftarrow K_p \leftarrow \psi(x_k)} \left(\frac{1}{m_r} e_k \right) + \frac{1}{2} \int_0^t \frac{1}{m_r} \tilde{\theta}_k^T \tilde{\theta}_k d\tau \quad (\text{IV.31})$$

Notons que

$$\Delta V_k = V_k - V_{k-1} \quad (\text{IV.32})$$

De (IV.31) et (IV.32), il vient

$$\begin{aligned} \Delta V_k(t) &= \frac{1}{2m_r} e_k^T \leftarrow K_p \leftarrow \psi(x_k) \frac{1}{m_r} e_k - \frac{1}{2m_r} e_{k-1}^T \leftarrow K_p \leftarrow \psi(x_{k-1}) \frac{1}{m_r} e_{k-1} \\ &\quad + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{IV.33})$$

Cela implique que

$$\Delta V_k(t) \leq \frac{1}{2m_r} e_k^T \leftarrow K_p \leftarrow \psi(x_k) \frac{1}{m_r} e_k + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \quad (\text{IV.34})$$

En utilisant l'hypothèse IV.3, (IV.34) devient

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\frac{1}{m_r} e_k^T + K_p + \psi(x_k) \left(\frac{m_r \dot{e}_k - \dot{m}_r e_k}{m_r^2} \right) + \frac{1}{2m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} e_k \right) d\tau \\ & + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (IV.35)$$

D'où

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\frac{1}{m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} \dot{e}_k - \frac{1}{m_r} e_k^T + K_p + \psi(x_k) \frac{\dot{m}_r}{m_r^2} e_k \right. \\ & \left. + \frac{1}{2m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} e_k \right) d\tau + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (IV.36)$$

Il est clair que

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\frac{1}{m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} \dot{e}_k + \frac{1}{2m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} e_k \right) d\tau \\ & + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (IV.37)$$

Sachant que $+ \psi(x_k) \psi(x_k) = I^{2n \times 2n}$, et en remplaçant (IV.27) et (IV.28) dans (IV.37), on obtient

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\frac{1}{m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} (\dot{y}_d(t) - \phi(x_k)) - e_k^T + K_p K_p \frac{1}{m_r} e_k \right. \\ & \left. - \frac{1}{m_r} \hat{\alpha}_k e_k^T (+K_p) K_p \operatorname{sgn}(e_k) + \frac{1}{2m_r} e_k^T + K_p + \psi(x_k) \frac{1}{m_r} e_k \right) d\tau \\ & + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (IV.38)$$

Tant que $+K_p K_p = I^{2n \times 2n}$, et en utilisant (IV.18), (IV.38) peut être réécrite comme suit

$$\Delta V_k(t) \leq \int_0^t \left(\frac{1}{m_r} \alpha e_k^T \operatorname{sgn}(e_k) - \frac{1}{m_r} e_k^T e_k - \frac{1}{m_r} \hat{\alpha}_k e_k^T \operatorname{sgn}(e_k) \right) d\tau$$

$$+ \frac{1}{2m_r} e_k^T (-K_p) \dot{\psi}(x_k) \frac{1}{m_r} e_k d\tau + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \quad (\text{IV.39})$$

Sachant que $\tilde{\alpha}_k = \alpha - \hat{\alpha}_k$, alors (IV.39) devient

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\frac{1}{m_r} \tilde{\alpha}_k e_k^T \text{sgn}(e_k) - \frac{1}{m_r} e_k^T e_k + \frac{1}{2m_r} e_k^T (-K_p) \dot{\psi}(x_k) \frac{1}{m_r} e_k \right) d\tau \\ & + \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \end{aligned} \quad (\text{IV.40})$$

De (IV.23) et (IV.24), on trouve

$$\tilde{\theta}_k = \tilde{\theta}_{k-1} - e_k^T \text{sgn}(e_k) \mu_k \quad (\text{IV.41})$$

Maintenant, nous examinons le dernier terme de (IV.40). En utilisant (IV.41), on obtient

$$\begin{aligned} & \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ & = \frac{1}{2} \int_0^t \frac{1}{m_r} \left((\tilde{\theta}_{k-1} - e_k^T \text{sgn}(e_k) \mu_k)^T \tilde{\theta}_k \right. \\ & \quad \left. - (\tilde{\theta}_k + e_k^T \text{sgn}(e_k) \mu_k)^T \tilde{\theta}_{k-1} \right) d\tau \end{aligned} \quad (\text{IV.42})$$

Comme $\tilde{\theta}_{k-1}^T \tilde{\theta}_k = \tilde{\theta}_k^T \tilde{\theta}_{k-1}$, alors (41) devient

$$\begin{aligned} & \frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ & = \frac{1}{2} \int_0^t \frac{1}{m_r} \left(-(e_k^T \text{sgn}(e_k) \mu_k)^T \tilde{\theta}_k - (e_k^T \text{sgn}(e_k) \mu_k)^T \tilde{\theta}_{k-1} \right) d\tau \end{aligned} \quad (\text{IV.43})$$

En remplaçant (IV.41) dans (IV.43), il vient

$$\frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq \frac{1}{2} \int_0^t \frac{1}{m_r} (-2e_k^T \text{sgn}(e_k) \mu_k)^T \tilde{\theta}_k d\tau \quad (\text{IV.44})$$

Tan que $\mu_k^T \tilde{\theta}_k = \tilde{\theta}_k^T \mu_k = \tilde{\alpha}_k$, alors

$$\frac{1}{2} \int_0^t \frac{1}{m_r} (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq - \int_0^t \frac{1}{m_r} \tilde{\alpha}_k e_k^T \text{sgn}(e_k) d\tau \quad (\text{IV.45})$$

En remplaçant (IV.45) dans (IV.40), on trouve

$$\Delta V_k \leq \int_0^t -\frac{1}{m_r} e_k^T e_k + \frac{1}{2m_r} e_k^T + K_p + \dot{\psi}(x_k) \frac{1}{m_r} e_k d\tau \quad (\text{IV.46})$$

Maintenant, on distingue deux cas, a) $m_r \geq 1$ et b) $m_r < 1$. On va les traiter

a) Si $m_r \geq 1$ alors

$$\frac{1}{m_r} e_k^T + K_p + \dot{\psi}(x_k) e_k \leq e_k^T + K_p + \dot{\psi}(x_k) e_k \quad (\text{IV.47})$$

De (IV.46) et (IV.47), il est clair que

$$\Delta V_k \leq \int_0^t \frac{1}{m_r} (-e_k^T e_k + \frac{1}{2} e_k^T + K_p + \dot{\psi}(x_k) e_k) d\tau \quad (\text{IV.48})$$

Donc

$$\Delta V_k \leq -\int_0^t \frac{1}{m_r} (1 - \frac{1}{2} \| +K_p + \dot{\psi}(x_k) \|) \|e_k\|^2 d\tau \quad (\text{IV.49})$$

Si on choisit

$$\| +K_p + \dot{\psi}(x_k) \| < 2 \quad (\text{IV.50})$$

Il vient

$$\Delta V_k \leq \int_0^t -\frac{1}{m_r} \|e_k\|^2 d\tau \quad (\text{IV.51})$$

Il est clair que

$$\Delta V_k \leq 0 \quad (\text{IV.52})$$

Par conséquent, V_k est décroissante.

b) Si $m_r < 1$, à partir de (IV.46), on trouve

$$\Delta V_k \leq \int_0^t \frac{1}{m_r} (-e_k^T e_k + \frac{1}{2m_r} e_k^T + K_p + \dot{\psi}(x_k) e_k) d\tau \quad (\text{IV.53})$$

Si $m_r < 1$ alors, on choisit $\|{}^+K_p + \dot{\psi}(x_k)\| < 2\|m_r\|$. Sachant que $\|m_r\| < 1$, et $\|{}^+K_p + \dot{\psi}(x_k)\| < 2\|m_r\| < 2$, alors nous arrivons à la même équation de (51). Il est clair que $\Delta V_k \leq 0$. Par conséquent, V_k est décroissante.

Dans les deux cas ($m_r \geq 1$ or $m_r < 1$), si on choisit $\|{}^+K_p + \dot{\psi}(x_k)\| < 2$, alors, $\Delta V_k \leq 0$ et V_k est décroissante.

Partie 2 : Maintenant, nous Montrons que $V_0(t)$ est bornée $\forall t \in [0, T]$

À partir de (31), avec $k=0$, on obtient

$$V_0(t) = \frac{1}{2} \left(\frac{1}{m_r} e_0^T \right) {}^+K_p + \psi(x_0) \left(\frac{1}{m_r} e_0 \right) + \frac{1}{2} \int_0^t \frac{1}{m_r} \tilde{\theta}_0^T \tilde{\theta}_0 d\tau \quad (\text{IV.54})$$

Par dérivation, il vient

$$\begin{aligned} \dot{V}_0(t) &= \frac{1}{m_r} e_0^T {}^+K_p + \psi(x_0) \left(\frac{m_r \dot{e}_0 - \dot{m}_r e_0}{m_r^2} \right) + \frac{1}{2m_r} e_0^T {}^+K_p + \dot{\psi}(x_k) \frac{1}{m_r} e_0 \\ &\quad + \frac{1}{2m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \end{aligned} \quad (\text{IV.55})$$

Donc

$$\begin{aligned} \dot{V}_0(t) &= \frac{1}{m_r} e_0^T {}^+K_p + \psi(x_0) \frac{1}{m_r} \dot{e}_0 - \frac{1}{m_r} e_0^T {}^+K_p + \psi(x_0) \frac{\dot{m}_r}{m_r^2} e_0 \\ &\quad + \frac{1}{2m_r} e_0^T {}^+K_p + \dot{\psi}(x_0) \frac{1}{m_r} e_0 + \frac{1}{2m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \end{aligned} \quad (\text{IV.56})$$

Tan que $m_r > 0$ et $\dot{m}_r > 0$, alors

$$\dot{V}_0(t) \leq \frac{1}{m_r} e_0^T {}^+K_p + \psi(x_0) \frac{1}{m_r} \dot{e}_0 + \frac{1}{2m_r} e_0^T {}^+K_p + \dot{\psi}(x_0) \frac{1}{m_r} e_0 + \frac{1}{2m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \quad (\text{IV.57})$$

En remplaçant (IV.27) et (IV.28) (avec $k=0$) dans (IV.57), on trouve

$$\begin{aligned} \dot{V}_0(t) &\leq \frac{1}{m_r} e_0^T {}^+K_p + \psi(x_0) \frac{1}{m_r} (\dot{y}_d(t) - \phi(x_0)) - e_0^T {}^+K_p K_p \frac{1}{m_r} e_0 \\ &\quad - \frac{1}{m_r} \hat{\alpha}_0 e_0^T {}^+K_p K_p \text{sgn}(e_0) + \frac{1}{2m_r} e_0^T {}^+K_p + \dot{\psi}(x_0) \frac{1}{m_r} e_0 + \frac{1}{2m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \end{aligned} \quad (\text{IV.58})$$

Sachant que ${}^+K_p K_p = I^{2n \times 2n}$ et $\tilde{\alpha}_k = \alpha - \hat{\alpha}_k$, et en utilisant (IV.18), on obtient

$$\dot{V}_0(t) \leq \frac{1}{m_r} \tilde{\alpha}_0 e_0^T \text{sgn}(e_0) - \frac{1}{m_r} e_0^T e_0 + \frac{1}{2m_r} e_0^{T+K_p+\psi(x_0)} \frac{1}{m_r} e_0 + \frac{1}{2m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \quad (\text{IV.59})$$

À partir de (IV.41) avec $k=0$, on trouve

$$\tilde{\theta}_0 = -e_0^T \text{sgn}(e_0) \mu_0 \quad (\text{IV.60})$$

Tant que $\frac{1}{2} \frac{1}{m_r} \tilde{\theta}_0^T \tilde{\theta}_0 \leq \frac{1}{m_r} \tilde{\theta}_0^T \tilde{\theta}_0$ et $\tilde{\alpha}_0 = \tilde{\theta}_0^T \mu_0 = \mu_0^T \tilde{\theta}_0$, il vient

$$\dot{V}_0(t) \leq \frac{1}{m_r} (-e_0^T e_0 + \frac{1}{2m_r} e_0^{T+K_p+\psi(x_0)} e_0) \quad (\text{IV.61})$$

a) Si $m_r \geq 1$ alors

$$\frac{1}{m_r} e_0^{T+K_p+\psi(x_0)} e_0 \leq e_0^{T+K_p+\psi(x_0)} e_0 \quad (\text{IV.62})$$

Donc

$$\dot{V}_0(t) \leq \frac{1}{m_r} (-e_0^T e_0 + \frac{1}{2} e_0^{T+K_p+\psi(x_0)} e_0) \quad (\text{IV.63})$$

Si on choisit $\|+K_p+\psi(x_0)\| < 2$, on obtient

$$\dot{V}_0(t) \leq -\frac{1}{2m_r} \|e_0\|^2 \quad (\text{IV.64})$$

Il est clair que $\dot{V}_0(t) \leq 0$.

b) Si $m_r < 1$, dans ce cas, on choisit $\|+K_p+\psi(x_0)\| < 2\|m_r\|$. Sachant que $\|m_r\| < 1$ et $\|+K_p+\psi(x_0)\| < 2\|m_r\| < 2$, alors nous arrivons à la même équation de (IV.64). Il est clair que $\dot{V}_0(t) \leq 0$.

Dans les deux cas ($m_r \geq 1$ et $m_r < 1$), si on choisit $\|+K_p+\psi(x_0)\| < 2$ nous obtenons $\dot{V}_0(t) \leq 0$. Il est clair que \dot{V}_0 est décroissante.

À partir de (IV.54) et (IV.64), nous pouvons conclure que $\tilde{\theta}_0, e_0$ et V_0 sont bornées.

Partie 3 : La fonction $V_k(t)$ peut être réécrite sous forme

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_j(t) \quad (\text{IV.65})$$

En utilisant (IV.51), on obtient

$$V_k(t) \leq V_0(t) - \sum_{j=1}^k \int_0^t -\frac{1}{2m_r} \|e_k\|^2 d\tau \quad (\text{IV.66})$$

Donc

$$\sum_{j=1}^k -\frac{1}{2m_r} \int_0^t \|e_k\|^2 d\tau \leq V_0(t) - V_k(t) \leq V_0(t) \quad (\text{IV.67})$$

Comme $V_0(t)$ est bornée, alors

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \forall t \in [0, T] \quad (\text{IV.68})$$

Cas 2 : $\tau_k \in]-\infty, 0]$; on utilise le même raisonnement du premier cas, en remplaçant m_r par m_l .

A partir du premier cas on peut trouver

$$\Delta V_k \leq \int_0^t -\frac{1}{m_l} e_k^T e_k + \frac{1}{2m_l} e_k^T {}^+K_p {}^+\dot{\psi}(x_k) \frac{1}{m_l} e_k d\tau \quad (\text{IV.69})$$

En utilisant les mêmes étapes du premier cas pour m_l , et suivant la valeur de m_l (si sa norme est inférieure à 1 ou supérieure à 1) nous arrivons au résultat suivant : si on choisit $\|{}^+K_p {}^+\dot{\psi}(x_k)\| < 2$, alors le système en boucle fermée est asymptotiquement stable.

Maintenant on va examiner l'inégalité (IV.50) afin de trouver une condition simple. Par conséquent, à partir de l'équation (IV.19), il est facile de constater que

$$\frac{d}{dt} ({}^+\psi(x_k)) = (\psi^T(x_k) \psi(x_k))^{-1} \dot{\psi}^T(x_k) \quad (\text{IV.70})$$

$$\text{Où } \dot{\psi}(x_k(t), t) = \begin{bmatrix} 0 \\ \frac{d}{dt} (M^{-1}(q_k)) \end{bmatrix}.$$

De la littérature robotique [KEL 05, p 98], il est bien connu que

$$\frac{d}{dt} (M^{-1}(q_k)) = C(q_k, \dot{q}_k) + C(q_k, \dot{q}_k)^T \quad (\text{IV.71})$$

À partir de (IV.70), (IV.71), les propriétés IV.1 et IV.3, l'hypothèse IV.5, (IV.50) devient

$$\lambda_{\max}(+K_p) < \frac{\beta_1^2}{C_m V_m} \quad (\text{IV.72})$$

Où $\lambda_{\max}(+K_p)$ est la valeur propre maximale de $+K_p$.

À partir de (20), (72) devient

$$\lambda_{\min}(K_p) > \frac{C_m V_m}{\beta_1^2} \quad (\text{IV.73})$$

Et donc nous avons arrivé à une condition plus simple.

IV.2.3 Résultats de simulation

IV.2.3.1 Résultats de simulation sur le robot manipulateur Puma 560

Afin de valider la loi de commande adaptative par apprentissage itératif proposée, on considère le problème de suivi des trajectoires d'un robot industriel PUMA 560 [ALA 08]. Il est présenté dans la Figure IV.3. Nous considérons juste les trois premières articulations.

Les éléments de $M(q_k)$ sont donnés par :

$$m_{11} = 2.57 + 1.38c_2^2 + 0.3s_2s_3 + 7.44 \times 10^{-1}c_2s_{23}$$

$$m_{12} = m_{21} \approx 6.90 \times 10^{-1}s_2 - 1.34 \times 10^{-1}c_{23} + 2.38 \times 10^{-2}c_2$$

$$m_{13} = m_{31} \approx -1.34 \times 10^{-1}c_{23} - 3.97 \times 10^{-3}s_{23}$$

$$m_{22} \approx 6.79 + 7.44 \times 10^{-3}s_3$$

$$m_{23} = m_{32} \approx 0.333 + 3.72 \times 10^{-1}s_3 - 1.10 \times 10^{-2}c_3$$

$$m_{33} \approx 1.16$$

Les éléments de $C(q_k, \dot{q}_k)$ sont donnés par :

$$C_{11} \approx (-2.76s_2c_2 + 7.44 \times 10^{-1}c_{23} + 0.60s_2c_3 - 2.13 \times 10^{-2}(1 - 2s_2s_3))\dot{q}_{2,k}$$

$$C_{12} \approx (6.90 \times 10^{-1}c_2 + 1.34 \times 10^{-1}s_{23} - 2.38 \times 10^{-2}s_2)\dot{q}_{2,k} \\ + (2.67 \times 10^{-1}s_{23} - 7.58 \times 10^{-3}c_{23})\dot{q}_{3,k}$$

$$C_{13} \approx (7.44 \times 10^{-1}c_2c_{23} + 0.60s_2c_3 + 2.20 \times 10^{-2}c_2s_{23} - 2.13 \times 10^{-2}(1 - 2s_2s_3))\dot{q}_{1,k}$$

$$C_{21} \approx -0.5(-2.76s_2c_2 + 7.44 \times 10^{-1}c_{23} + 0.60s_2c_3 - 2.13 \times 10^{-2}(1 - 2s_2s_3))\dot{q}_{1,k}$$

$$C_{22} \approx (2.20 \times 10^{-2} s_3 + 7.44 \times 10^{-1} c_3) \dot{q}_{3,k}$$

$$C_{23} \approx 0.5(2.20 \times 10^{-2} s_3 + 7.44 \times 10^{-1} c_3) \dot{q}_{3,k}$$

$$C_{31} \approx -0.5(7.44 \times 10^{-1} c_3 c_{23} + 0.60 s_2 c_3 + 2.20 \times 10^{-2} c_2 s_{23} - 2.13 \times 10^{-2} (1 - 2s_2 s_3)) \dot{q}_{1,k}$$

$$C_{32} \approx -0.5(2.20 \times 10^{-2} s_3 + 7.44 \times 10^{-1} c_3) \dot{q}_{3,k}$$

$$C_{33} \approx 0$$

Les éléments de $G(q_k)$ sont donnés par :

$$g_1 = 0; ; .g_2 \approx -37.2c_2 - 8.4s_{23} + 1.02s_2g_3 \approx -8.4s_{23} - 0.25c_{23}$$

C_m dans l'équation (IV.4) est définie comme suit [KEL 05]:

$$C_m \geq n^2 \left(\sup_{i,j,k,q} |c_{ijk}(q)| \right) \quad (IV.74)$$

Pour notre robot $n = 3$. Par conséquent, nous trouvons $C_m = 12.42[kgm^2]$, $V_m = 3[rad/s]$, $\beta_1 = 1.348[kgm^2]$ et $\lambda_{min}(k_p) > 20.51$. Dans cette simulation, on s'intéresse seulement à la poursuite des trajectoires de positions. On choisit $K_p = diag\{25,35,21\}$.

Les perturbations externes $d_k(t)$ sont données par :

$$\begin{cases} d_1 = rank(k) \sin(2t) \\ d_2 = -rank(k) \sin(2t) \\ d_3 = rank(k) \sin(2t) \end{cases} \quad (IV.76)$$

Les trajectoires désirées sont définies :

$$\begin{cases} q_{d1}(t) = \frac{\pi}{2} \sin(2\pi t) + \pi \\ q_{d2}(t) = \sin(2\pi t) + 1 \\ q_{d3}(t) = \frac{\pi}{2} \sin(2\pi t) - \pi \end{cases} \quad (IV.77)$$

Les paramètres du réseau neuronal RBF sont choisis comme $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 0.2$, $C_1 = 0.5$, $C_2 = 0.3$, $C_3 = -0.3$ et $C_4 = -0.5$. Pour l'architecture de RBF on a utilisé deux neurones dans la couche d'entrée, quatre neurones dans la couche cachée et un neurone dans la couche de sortie. Les paramètres sont choisis par essai et erreur. Les paramètres de la zone-morte sont choisis comme : $b_r = 2$, $b_l = -2$, $m_r(t) = 1 + t + 0.2\sin(t)$ et $m_l(t) = 1 + t - 0.2\sin(t)$.

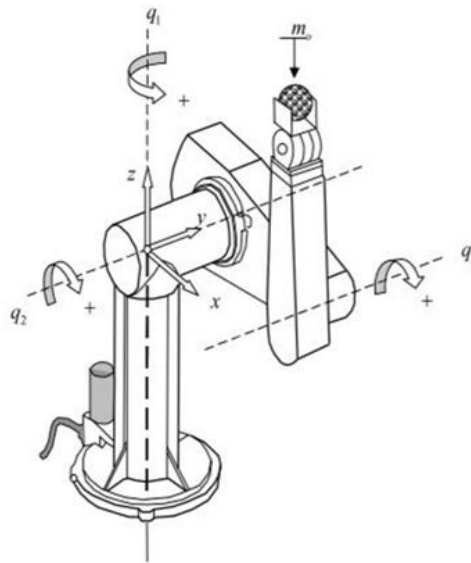


Figure IV.3 Robot Manipulateur PUMA 560.

En appliquant la loi de commande proposée (IV.28) avec les lois d'adaptation (IV.23-IV.29), les résultats de simulation à travers les itérations sont présentés dans les figures IV.4-IV.9. Les trajectoires réelles et désirées à la 2^{ème}, 10^{ème} et 30^{ème} itérations pour la première, deuxième et troisième articulations sont présentées dans les figures IV.4, IV.6 et IV.8, respectivement. On peut voir que les trajectoires réelles suivent exactement les trajectoires désirées à travers les itérations, pour les trois articulations. Les erreurs de poursuite pour la première, la deuxième et la troisième articulation sont présentées dans les figures IV.5, IV.7 et IV.9, respectivement. Il est très clair que les erreurs diminuent à travers les itérations pour toutes les articulations. Dans la première itération les valeurs des erreurs maximales sont 13, 8.4 et 19.1 pour la première, deuxième et la troisième articulation, respectivement. Après 30 itérations, les erreurs maximales deviennent inférieures de 2.5×10^{-2} , 1.8×10^{-2} et 3.7×10^{-2} pour la première, la deuxième et la troisième articulation, respectivement. Donc les erreurs de poursuite sont réduites plus de 520 fois, 466 et 516 fois pour la première, la deuxième et troisième articulation, respectivement. Par conséquent, nous pouvons conclure que la méthode proposée fonctionne très bien.

IV.2.3.2 Comparaison entre la commande proposée et la CAI de type-D

L'exemple de simulation utilisé dans [GU 19] est bien adopté pour faire une comparaison entre la CAI de type-D [GU 19] et la méthode proposée dans ce chapitre.

Le modèle dynamique du système non linéaire dans [GU 19] est décrit par :

$$\begin{cases} \dot{x}_k(t) = Ax_k(t) + f(x_k(t)) + Bu_k(t) \\ y_k(t) = Cx_k(t) \end{cases} \quad (\text{IV.78})$$

$$\text{où } x_k(t) = \begin{bmatrix} x_{1k}(t) \\ x_{2k}(t) \end{bmatrix}, A = \begin{bmatrix} 1 & -2 \\ -1 & 1 \end{bmatrix}, f(x_k(t)) = \begin{bmatrix} -x_{1k}(t)(x_{1k}^2(t) + x_{2k}^2(t)) \\ -x_{2k}(t)(x_{1k}^2(t) + x_{2k}^2(t)) \end{bmatrix},$$

$$B = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \text{ et } C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Les trajectoires désirées sont choisies comme suit

$$y_d(t) = \begin{bmatrix} \sin(2\pi t) \\ t^2(5t^2 - 1) \end{bmatrix} \quad (\text{IV.79})$$

En appliquant la loi de commande proposée (IV.28) avec les lois d'adaptations (IV.23-IV.29), avec $K_p = \text{diag}\{25,25\}$, et en gardons les mêmes paramètres de RBF utilisés dans la section précédente, les résultats de la simulation sont présentés dans les figures IV.10 et IV.11. On constate qu'après seulement 10 itérations on obtient des très bonnes performances pour les deux sorties. Maintenant, la commande par apprentissage itératif de type-D- est appliquée sur le même système (IV.78). La loi de commande est définie comme suit [Gu 19]

$$u_{k+1}(t) = u_k(t) + \Lambda \dot{e}_{k+1}(t) \quad (\text{IV.80})$$

$$\text{Avec } \Lambda = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ et } u_0(t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Maintenant, pour faire la comparaison avec cette méthode, nous avons utilisé la norme sup de l'erreur de poursuite du système ($\sup |e(t)|$) comme un indice de comparaison d'évolution des performances de système (IV.78). Les figures IV.12 et IV.13 présentent la norme sup de l'erreur de poursuite à travers les itérations pour la première et la deuxième sortie, respectivement. À partir de ces résultats, il est clair que les performances obtenues à partir de notre méthode sont meilleures que l'autre méthode.

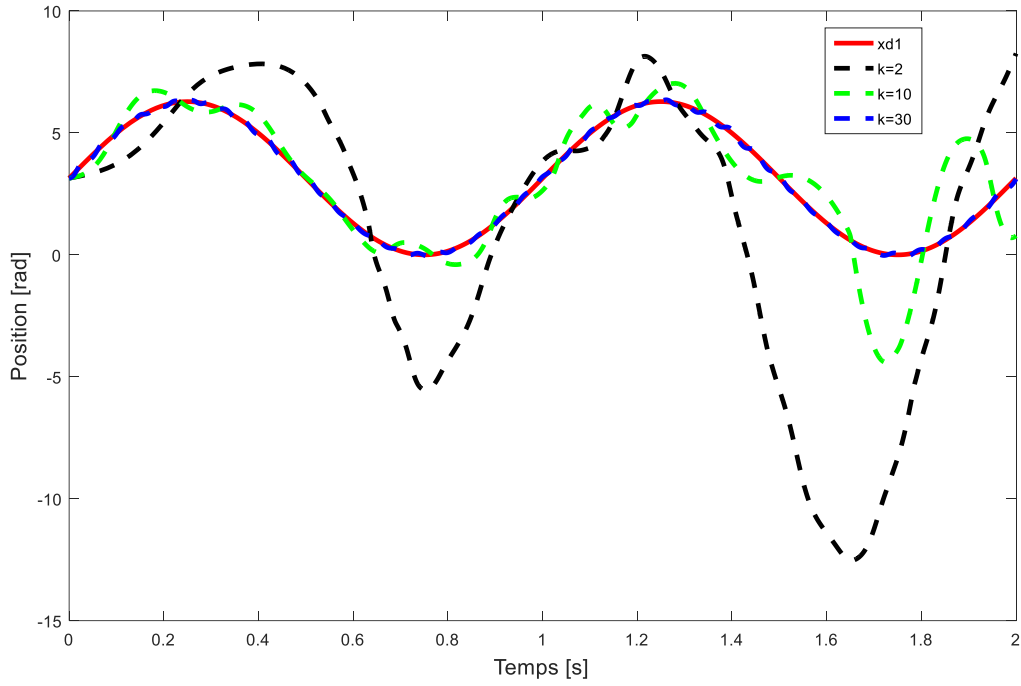


Figure IV.4 Trajectoires réelles et désirée de la première articulation.

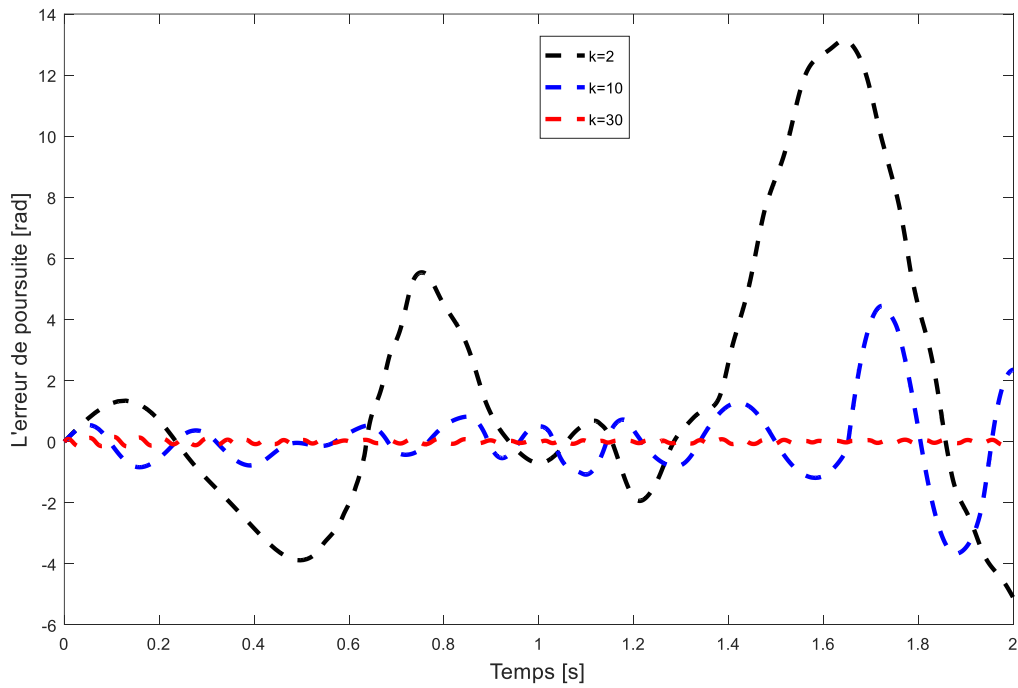


Figure IV.5 Erreurs de poursuite de la première articulation.

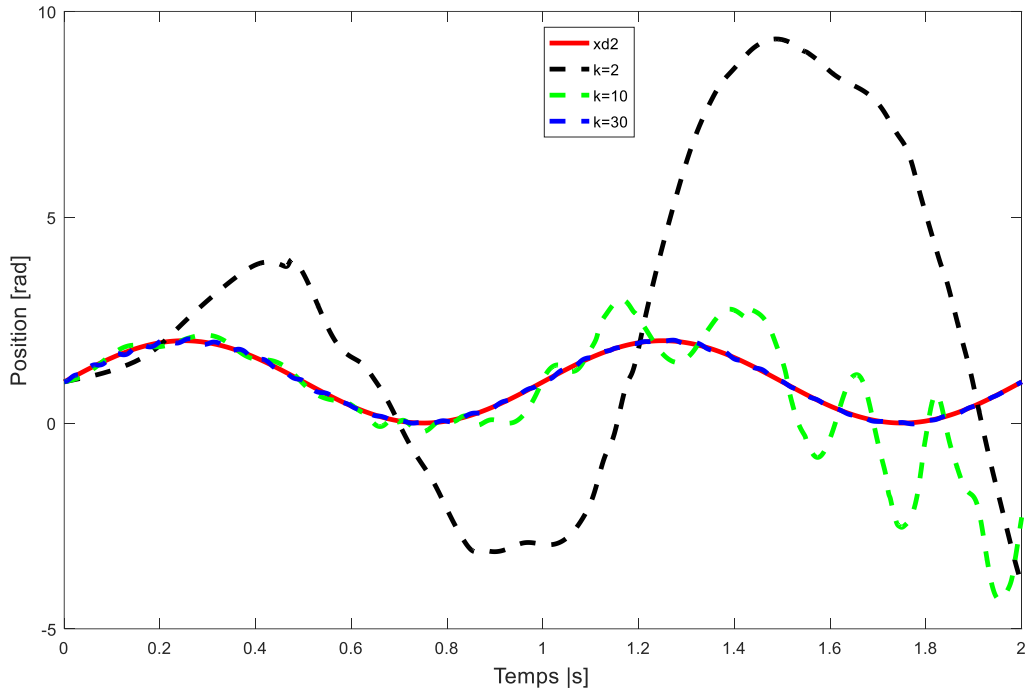


Figure IV.6 Trajectoires réelles et désirée de la deuxième articulation.

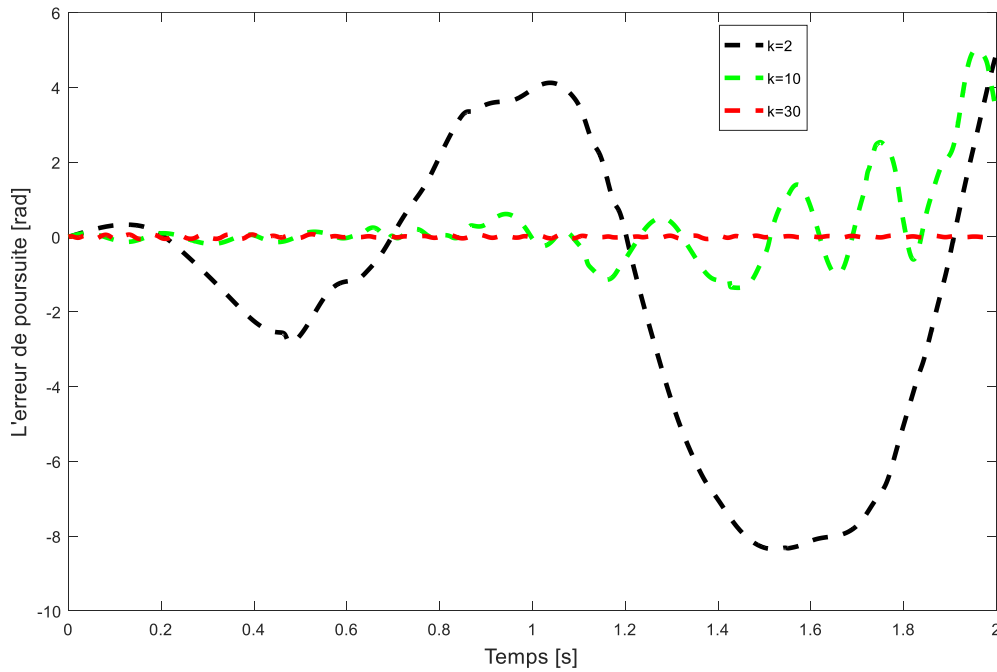


Figure IV.7 Erreurs de poursuite de la deuxième articulation.

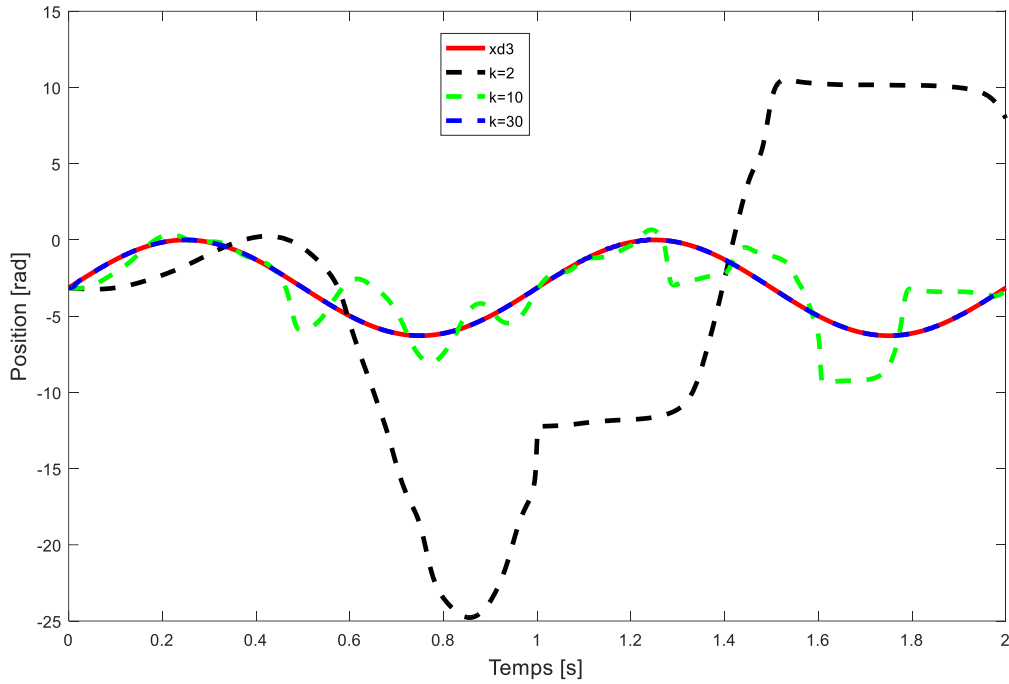


Figure IV.8 Trajectoires réelles et désirée de la troisième articulation.

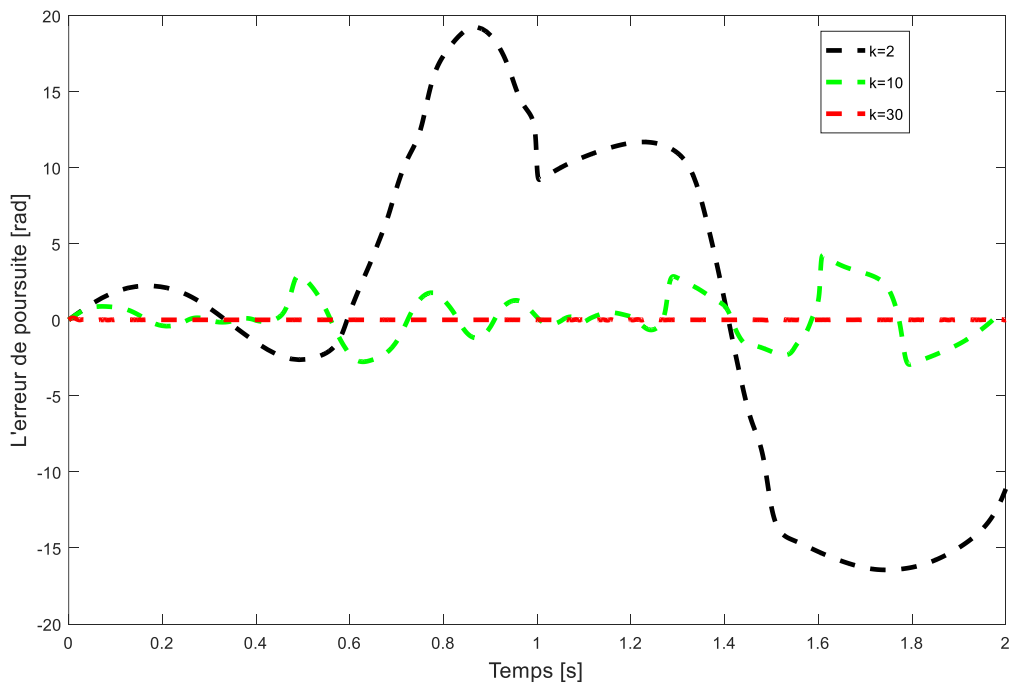


Figure IV.9 Erreurs de poursuite de la troisième articulation.

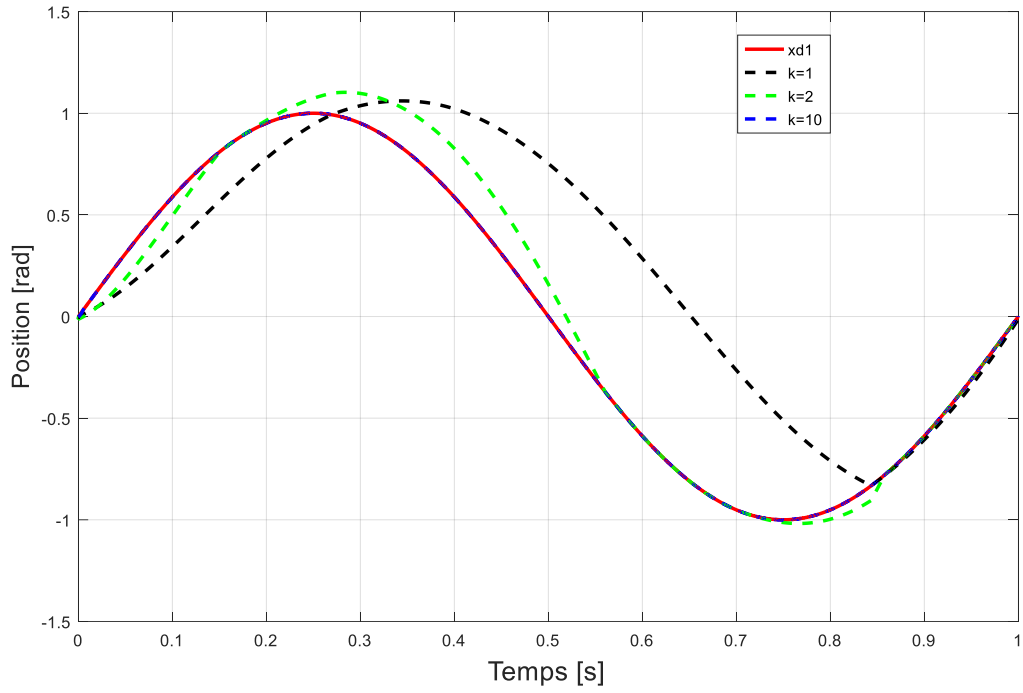


Figure IV.10 Trajectoires réelles et désirée de la première sortie.

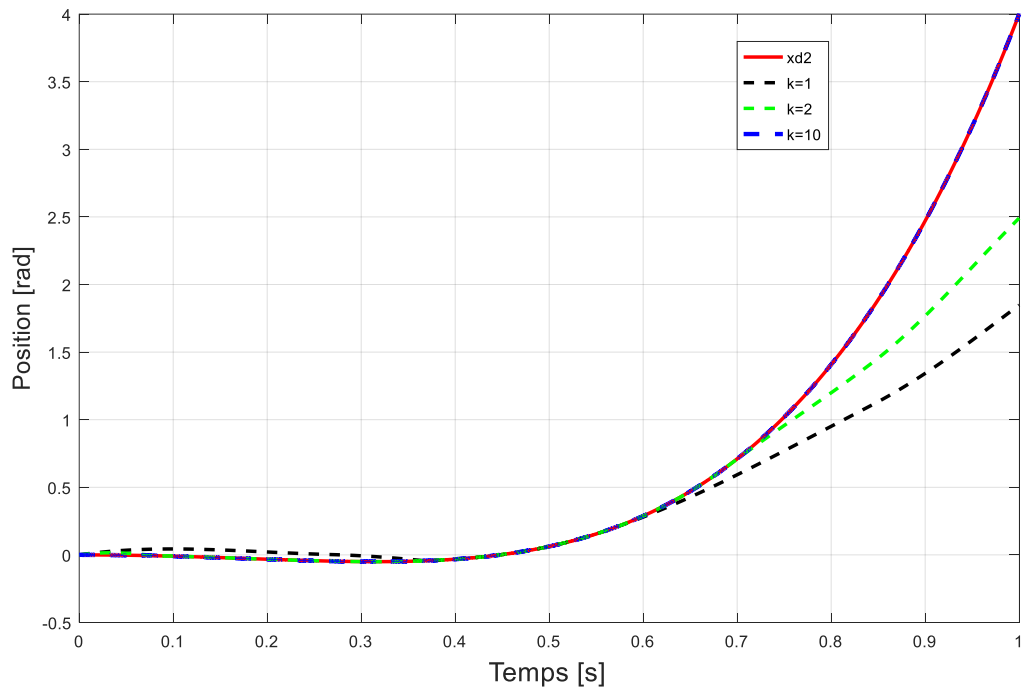


Figure IV.11 Trajectoires réelles et désirée de la deuxième sortie.

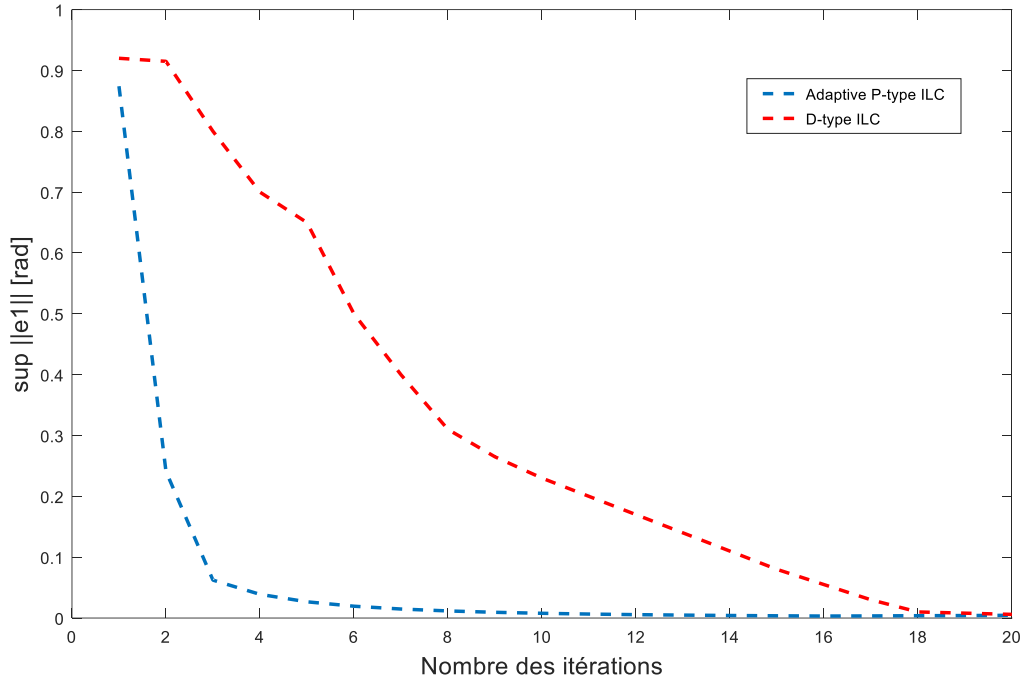


Figure IV.12 Norme sup de l'erreur du premier joint au cours des itérations pour les deux méthodes.

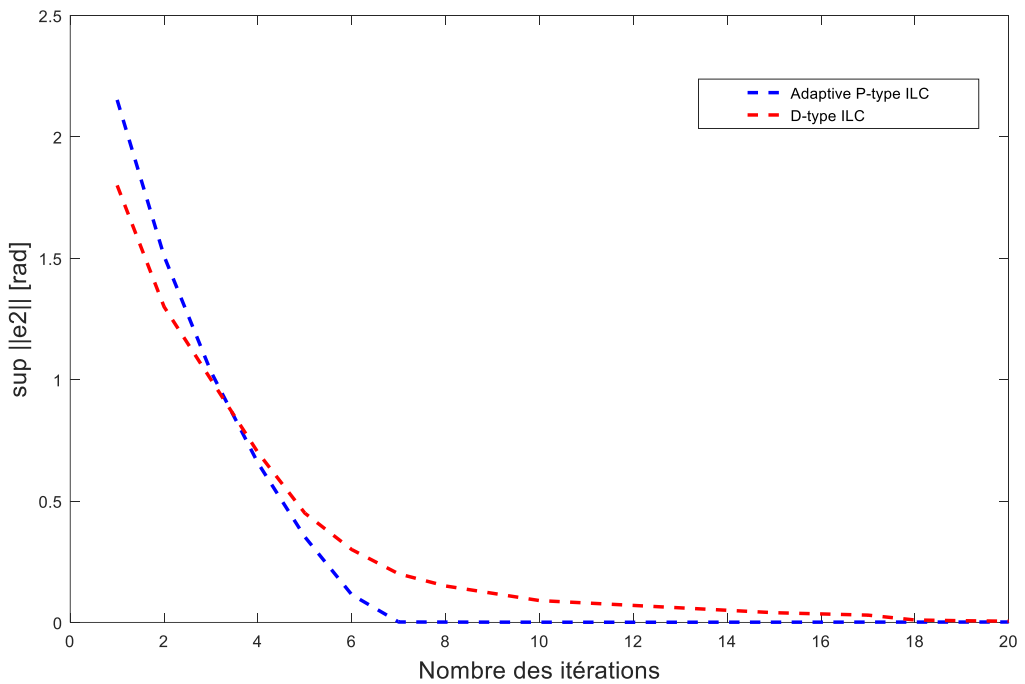


Figure IV.13 Norme sup de l'erreur du deuxième joint au cours des itérations pour les deux méthodes.

IV.3 Commande par apprentissage itératif d'ordre élevé des systèmes non linéaires multivariables avec saturation à l'entrée

IV.3.1 Formulation du problème

Considérons le système non linéaire avec la saturation dans son entrée

$$\dot{x}_k(t) = f(x_k(t), t) + B(t) \text{sat}(u_k(t), u^*) \quad (\text{IV.81})$$

où :

$x_k \in R^n, u_k \in R^n$ représentent respectivement, le vecteur d'état et la commande.

$f(*) = [f_1(*), \dots, f_n(*)] \in R^n$ est un vecteur des fonctions non linéaires inconnues, $B \in R^{n \times n}$ est une matrice de commande inconnue.

La fonction de saturation est définie comme suit

$$\text{sat}(u_k(t), u^*) = \begin{cases} u_k(t) & |u_k(t)| \leq u^* \\ \text{sgn}(u_k(t))u^* & \text{sinon} \end{cases} \quad (\text{IV.82})$$

Les lemmes suivants sont utilisés dans le reste de chapitre

Lemme IV.1 [JIA 04] : Pour une fonction de saturation w définie comme $w = \text{sat}(u, l) + z$, l'inégalité suivante est vraie :

$$(\text{sat}(w, l) - w)^T (\text{sat}(w, l) - w) \leq z^T z \quad (\text{IV.83})$$

avec w, u, l et $z \in R^l$

Lemme IV.2 [HOR 93] : Soit $x(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T \in R^n$ est définie pour $t \in [0, T]$, alors on obtient

$$\left(\int_0^t \|x(\tau)\| d\tau \right) e^{-\lambda t} \leq \frac{1}{\lambda} \|x(t)\|_\lambda, \quad \lambda > 0 \quad (\text{IV.84})$$

Lemme IV.3 [Gronwall-Bellman] [YAN 01] : Soient $f(t)$ et $g(t)$ deux fonctions scalaires réelles et localement intégrables dans $[a, b]$. Si la fonction $f(t)$ satisfait l'intégralité suivante

$$f(t) \leq L + \int_0^t g(\tau) f(\tau) d\tau, \quad t \in [a, b] \quad (\text{IV.85})$$

alors

$$f(t) \leq L \exp\left(\int_0^t g(\tau) d\tau\right) \quad (\text{IV.86})$$

avec L est une constante.

Hypothèse VI.1 : La fonction $f(x(t), t)$ satisfait la condition de continuité globale de Lipchitz pour $t \in [0, T]$, c.à.d. qu'il existe une constante α pour tout $x_1(t), x_2(t) \in R \times [0, T]$, on a

$$\|f(x_1(t), t) - f(x_2(t), t)\| \leq \alpha \|x_1(t) - x_2(t)\| \quad (\text{IV.87})$$

avec α est le gain de Lipschitz.

Hypothèse VI.2 : A chaque itération, les conditions initiales sont les mêmes, i.e., $x_k(0) = x_d(0)$.

Notre objectif est de développer une loi de commande par apprentissage de Type -P- et d'ordre élevé $u_k(t)$ en présence des saturations, pour forcer la sortie réelle $x_k(t)$ à suivre une trajectoire désirée $x_d(t)$, sur un intervalle de temps fixe $t \in [0, T]$, lorsque k tend vers l'infini.

IV.3.2 Synthèse de la loi de commande

Théorème VI : Etant donné le système non linéaire (IV.81), vérifiant les hypothèses IV.1 et IV.2. En appliquant la loi de commande par apprentissage itératif d'ordre élevé

$$u_{k+1}(t) = \text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t) \quad (\text{IV.88})$$

Alors

$$\lim_{k \rightarrow \infty} x_k(t) = x_d(t) \quad \forall t \in [0, T] \quad (\text{IV.89})$$

Avec P_1 et P_2 sont des gains positifs, $e_k(t) = x_d(t) - x_k(t)$ et $e_{k+1}(t) = x_d(t) - x_{k+1}(t)$.

Preuve

La solution générale $x_{k+1}(t)$ de (IV.81) peut-être écrite sous la forme suivante :

$$x_{k+1}(t) = x_{k+1}(0) + \int_0^t \left(f(x_{k+1}(t)) + B(t) \text{sat}(u_k(t), u^*) \right) dt \quad (\text{IV.90})$$

En remplaçant (IV.88) dans (IV.90), il vient

$$\begin{aligned}
x_{k+1}(t) &= x_{k+1}(0) \\
&+ \int_0^t \left(f(x_{k+1}(t)) \right. \\
&\left. + B(t) \text{sat}(\text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t), u^*) \right) d\tau
\end{aligned} \tag{IV.91}$$

En ajoutant et en soustrayant $f(x_k(t)) + B(t)(\text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t), u^*)$, on obtient

$$\begin{aligned}
x_{k+1}(t) &= x_{k+1}(0) + \int_0^t \left(f(x_{k+1}(t)) - f(x_k(t)) \right) d\tau \\
&+ \int_0^t \left(f(x_k(t)) \right. \\
&\left. + B(t)(\text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t), u^*) \right) \\
&+ \int_0^t B(t) \left(\text{sat}(\text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t), u^*) \right) \\
&- \left(\text{sat}(u_k(t), u^*) + P_1 e_k(t) + P_2 e_{k+1}(t) \right) d\tau
\end{aligned} \tag{IV.92}$$

En utilisant le lemme IV.1, (VI.92) peut être réécrite sous forme

$$\begin{aligned}
x_{k+1}(t) &= x_{k+1}(0) + \int_0^t \left(f(x_{k+1}(t)) - f(x_k(t)) \right) d\tau \\
&+ \int_0^t \left(f(x_k(t)) + B(t) \text{sat}(u_k(t), u^*) \right) \\
&+ \int_0^t (P_1 e_k(t) + P_2 e_{k+1}(t)) d\tau \\
&+ \int_0^t \mu (P_1 \|e_k(t)\| + P_2 \|e_{k+1}(t)\|) d\tau
\end{aligned} \tag{IV.93}$$

avec $\mu = \max(\|B(t)\|)$.

De (IV.81) et l'hypothèse IV.2, on trouve

$$\begin{aligned}
x_{k+1}(t) &\leq x_k(t) + \int_0^t (f(x_{k+1}(t)) - f(x_k(t))) d\tau \\
&\quad + \int_0^t (P_1 e_k(t) + P_2 e_{k+1}(t)) d\tau \\
&\quad + \int_0^t (\mu(P_1 \|e_k(t)\| + P_2 \|e_{k+1}(t)\|)) d\tau
\end{aligned} \tag{IV.94}$$

En appliquant les propriétés des normes sur les deux coté de (IV.94) et en utilisant l'hypothèse IV.1, on obtient

$$\begin{aligned}
&\|x_{k+1}(t) - x_k(t)\| \\
&\leq \alpha \int_0^t \|x_{k+1}(t) - x_k(t)\| d\tau + \int_0^t (P_1 \|e_k(t)\| + P_2 \|e_{k+1}(t)\|) d\tau \\
&\quad + \int_0^t \mu(P_1 \|e_k(t)\| + P_2 \|e_{k+1}(t)\|) d\tau
\end{aligned} \tag{IV.95}$$

En utilisant le lemme IV.3, il vient

$$\begin{aligned}
&\|x_{k+1}(t) - x_k(t)\| \\
&\leq \int_0^t ((P_1 + \mu P_1) \|e_k(t)\| + (P_2 + \mu P_2) \|e_{k+1}(t)\|) d\tau \exp(\alpha)
\end{aligned} \tag{IV.96}$$

En multipliant les deux côtés de (IV.96) par $e^{-\lambda t}$, on trouve

$$\begin{aligned}
&\|x_{k+1}(t) - x_k(t)\|_\lambda \\
&\leq \left(\left(\frac{P_1 + \mu P_1}{\lambda} \right) \|x_k(t)\|_\lambda + \left(\frac{P_2 + \mu P_2}{\lambda} \right) \|x_{k+1}(t)\|_\lambda \right) \exp\left(\frac{\alpha}{\lambda}\right)
\end{aligned} \tag{IV.97}$$

Sachant que $e_{k+1}(t) - e_k(t) = x_k(t) - x_{k+1}(t)$, alors (IV.94) peut être réécrite comme suit

$$\begin{aligned}
&-e_{k+1}(t) - \int_0^t (P_1 e_k(t) + P_2 e_{k+1}(t)) d\tau - \int_0^t \mu(P_1 \|e_k(t)\| + P_2 \|e_{k+1}(t)\|) d\tau \\
&\leq -e_k(t) - \int_0^t (f(x_{k+1}(t)) - f(x_k(t))) d\tau
\end{aligned} \tag{IV.98}$$

En utilisant l'hypothèse IV.1 et le lemme IV.2, il vient

$$\begin{aligned} & \left(\frac{P_1 + \mu P_1}{\lambda} \right) \|e_k(t)\|_\lambda + \left(1 + \frac{P_2 + \mu P_2}{\lambda} \right) \|e_{k+1}(t)\|_\lambda \\ & \leq \|e_k(t)\|_\lambda + \frac{\alpha}{\lambda} \|x_{k+1}(t) - x_k(t)\|_\lambda \end{aligned} \quad (\text{IV.99})$$

Soustrayant (IV.96) de (IV.99), on obtient

$$\begin{aligned} & \left(1 + \frac{P_2 + \mu P_2}{\lambda} \right) \|e_{k+1}(t)\|_\lambda \\ & \leq \left(1 - \frac{P_1 + \mu P_1}{\lambda} \right) \|e_k(t)\|_\lambda \\ & \quad + \frac{\alpha}{\lambda} \left(\left(\frac{P_1 + \mu P_1}{\lambda} \right) \|e_k(t)\|_\lambda + \left(\frac{P_2 + \mu P_2}{\lambda} \right) \|e_{k+1}(t)\|_\lambda \right) \exp\left(\frac{\alpha}{\lambda}\right) \end{aligned} \quad (\text{IV.100})$$

D'où

$$\begin{aligned} \|e_{k+1}(t)\|_\lambda & \leq \left(\frac{1 - \frac{P_1 + \mu P_1}{\lambda}}{1 + \frac{P_2 + \mu P_2}{\lambda}} \right) \|e_k(t)\|_\lambda \\ & \quad + \frac{\alpha \exp\left(\frac{\alpha}{\lambda}\right)}{\lambda \left(1 + \frac{P_2 + \mu P_2}{\lambda} \right)} \left(\left(\frac{P_1 + \mu P_1}{\lambda} \right) \|e_k(t)\|_\lambda \right. \\ & \quad \left. + \left(\frac{P_2 + \mu P_2}{\lambda} \right) \|e_{k+1}(t)\|_\lambda \right) \end{aligned} \quad (\text{IV.101})$$

Si on choisit $\lambda > 0$ assez grand, alors

$$\|e_{k+1}(t)\|_\lambda \leq \left(\frac{1 - \frac{P_1 + \mu P_1}{\lambda}}{1 + \frac{P_2 + \mu P_2}{\lambda}} \right) \|e_k(t)\|_\lambda \quad (\text{IV.102})$$

Comme $\left(\frac{1 - \frac{P_1 + \mu P_1}{\lambda}}{1 + \frac{P_2 + \mu P_2}{\lambda}} \right) < 1$, il est clair que

$$\|e_{k+1}(t)\|_\lambda \leq \rho \|e_k(t)\|_\lambda, \quad \rho < 1 \quad (\text{IV.103})$$

Donc

$$\lim_{k \rightarrow \infty} \|e_k(t)\|_\lambda = 0 \quad \forall t \in [0, T] \quad (\text{IV.104})$$

Par la suite

$$\lim_{k \rightarrow \infty} x_k(t) = x_d(t) \quad \forall t \in [0, T] \quad (\text{IV.105})$$

IV.3.3 Résultats de Simulation

IV.3.3.1 Exemple 1 : Bras manipulateur

Considérons un bras manipulateur avec le modèle dynamique suivant [BOU 16] :

$$\begin{bmatrix} \dot{x}_{1,k}(t) \\ \dot{x}_{2,k}(t) \end{bmatrix} = \begin{bmatrix} x_{2,k} \\ -J_m^{-1} S \sin(x_{1,k}) \end{bmatrix} + \begin{bmatrix} 0 \\ -J_m^{-1} \end{bmatrix} \text{sat}(u_k(t), u^*) \quad \forall t \in [0, 2] \quad (\text{IV.106})$$

Avec $J_m = 14 \text{ kg.m}^2$, $S = 6 \text{ kg.m}$ and $g = 9.8 \text{ m/s}^2$.

$$x_d(t) = \sin(2\pi t) \text{ et } y(t) = x_2(t)$$

Les paramètres de la commande sont choisis comme suit : $P_1 = 100$, $P_2 = 25$ et $u^* = 9$.

En appliquant la loi de commande proposée (IV.88) sur le système (IV.106), les résultats de simulations sont illustrés dans les figures IV.14, IV.15 et IV.16. A partir de la figure IV.14, il est clair que la trajectoire réelle de système suit exactement la trajectoire désirée à travers les itérations. La figure IV.15 présente la norme-sup de l'erreur à travers les itérations, on peut voir qu'après seulement quatre itérations, la norme-sup tend vers zéro. Pour vérifier l'influence de la fonction de saturation sur le système, la figure IV.16 présente le signal de commande pour différentes itérations.

IV.3.3.2 Exemple 2 : Système MIMO non linéaire

Considérons le système MIMO non linéaire, dont son modèle dynamique est présenté par [Li 13] :

$$\begin{bmatrix} \dot{x}_{1,k}(t) \\ \dot{x}_{2,k}(t) \end{bmatrix} = \begin{bmatrix} f_1(x_k) \\ f_2(x_k) \end{bmatrix} + \begin{bmatrix} 1.5 + t & 0 \\ 0 & 2 - 0.3t \end{bmatrix} \text{sat}(u_k(t), u^*) \quad \forall t \in [0, 2] \quad (\text{IV.107})$$

$$f_1(x_k) = (2 + 0.3t)x_{1,k} + (2.2 + 0.5t)x_{1,k}x_{2,k} + (0.8 - 0.1\cos(4t))x_{1,k}^2 \quad (\text{IV.108})$$

et

$$\begin{aligned} f_2(x_k) = & (2 + 0.3t)x_{1,k} + (0.7 + 0.08\sin(3t))x_{2,k}^2 + (2.2 + 0.5t)(x_{2,k} - x_{1,k}) \\ & + (0.8 - 0.1\cos(4t))(x_{1,k} - x_{2,k})^2 \end{aligned} \quad (\text{IV.109})$$

Le vecteur des trajectoires désirées est défini comme :

$$\begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} = \begin{bmatrix} \sin(2\pi t) \\ \cos(2\pi t) \end{bmatrix}; t \in [0,2] \quad (\text{IV.110})$$

Les paramètres de la commande sont choisis comme suit : $P_1 = 18$, $P_2 = 9$ et $u^* = 5$.

En appliquant la loi de commande proposée (IV.81) sur le système (IV.107), les résultats de simulations sont illustrés dans les figures IV.17, IV.18, IV.19 et IV.20. La norme-sup des erreurs de la première sortie et la deuxième sortie à travers les itérations sont présentées dans les figures IV.17 et IV.18, respectivement. Il est clair que la norme-sup de l'erreur tend vers zéro pour les deux sorties. Les figures IV.19 et IV.20 présentent les signaux de commande.

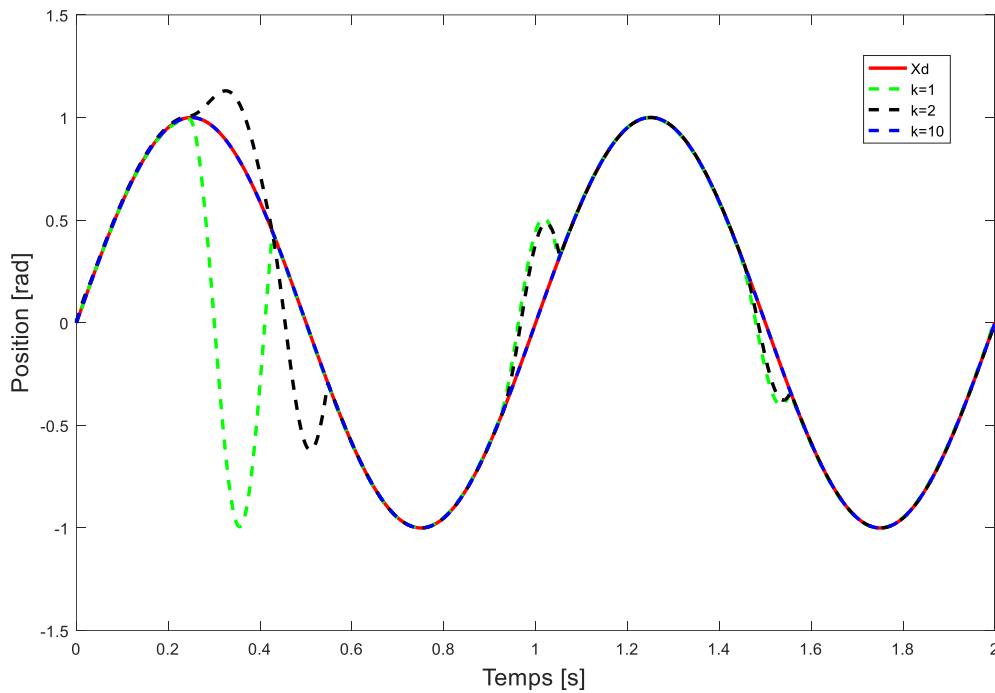


Figure IV.24 Trajectoires réelles et désirée à travers les itérations

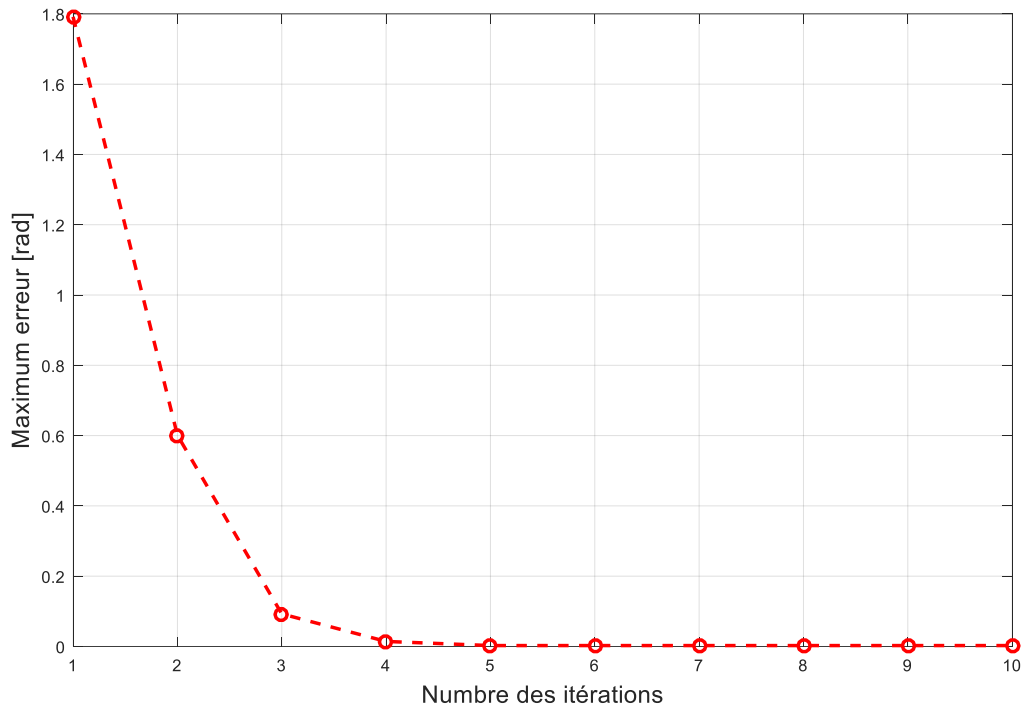


Figure IV.15 Norme Sup de l'erreur au cours des itérations.

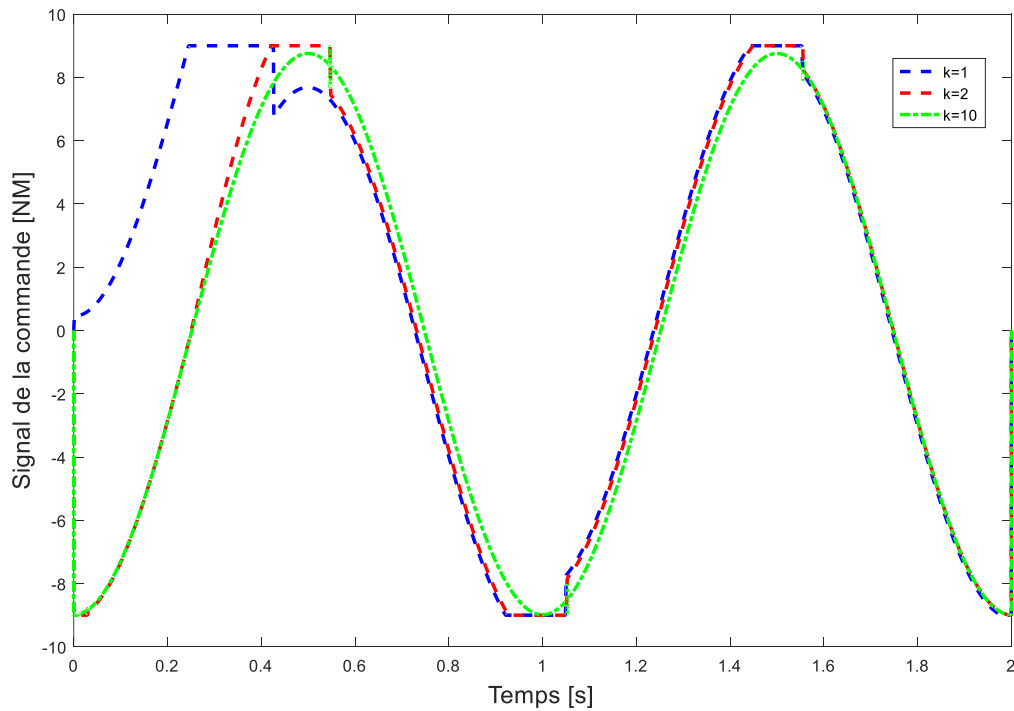


Figure IV.16 Signal de commande au cours des itérations.

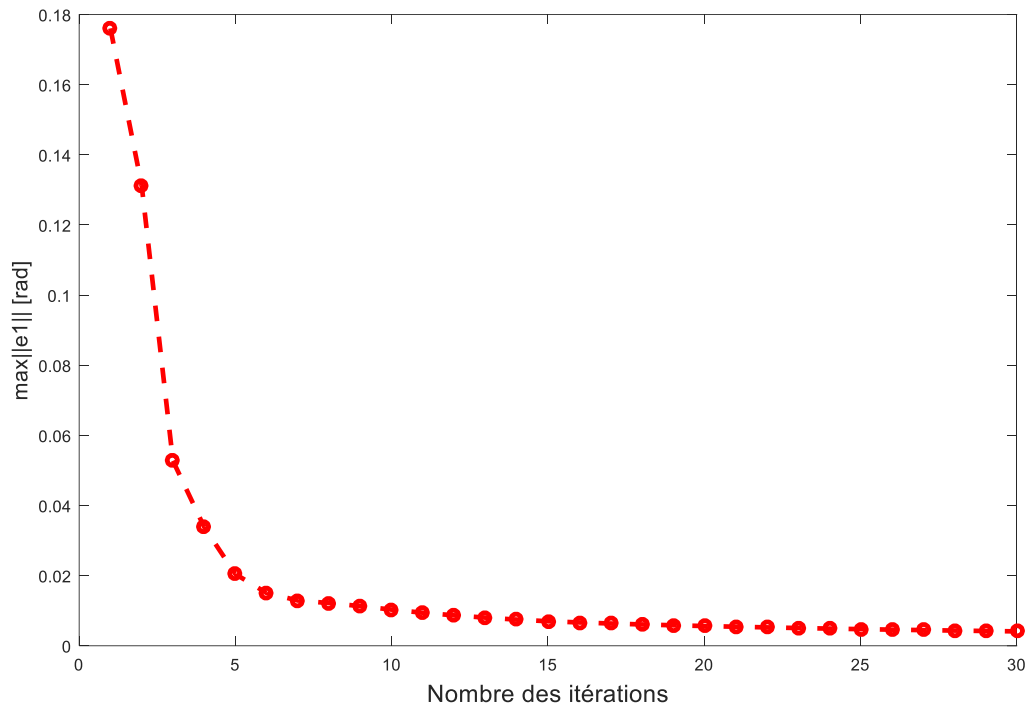


Figure IV.17 Norme sup de l'erreur de la première sortie au cours des itérations.

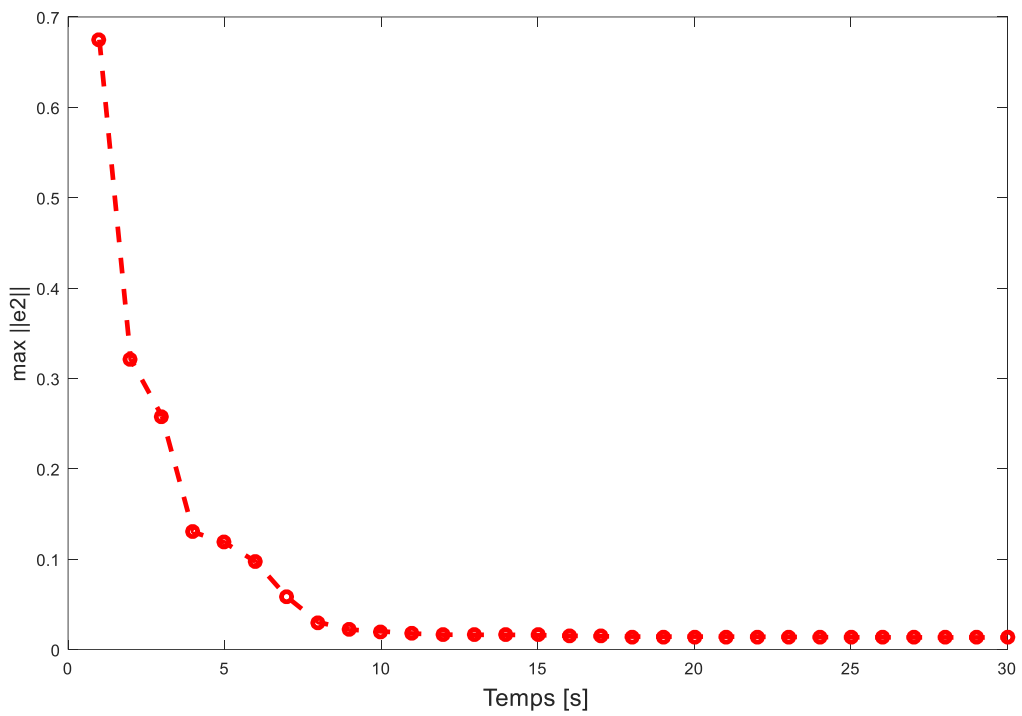


Figure IV.18 Norme sup de l'erreur de la deuxième sortie au cours des itérations.

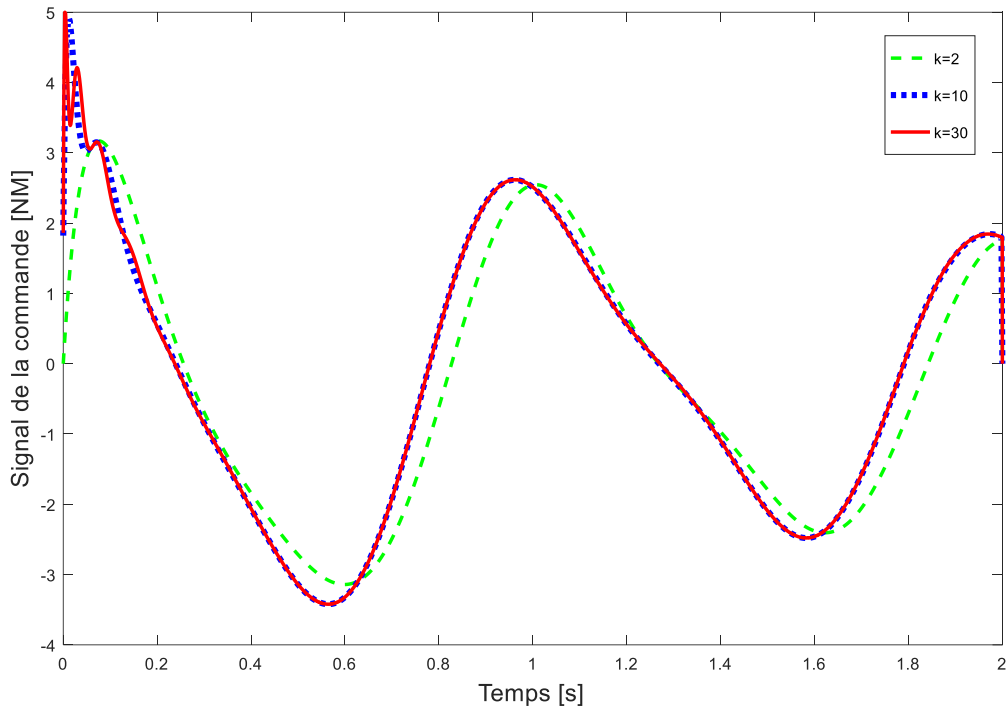


Figure IV.19 Premier signal de commande au cours des itérations.

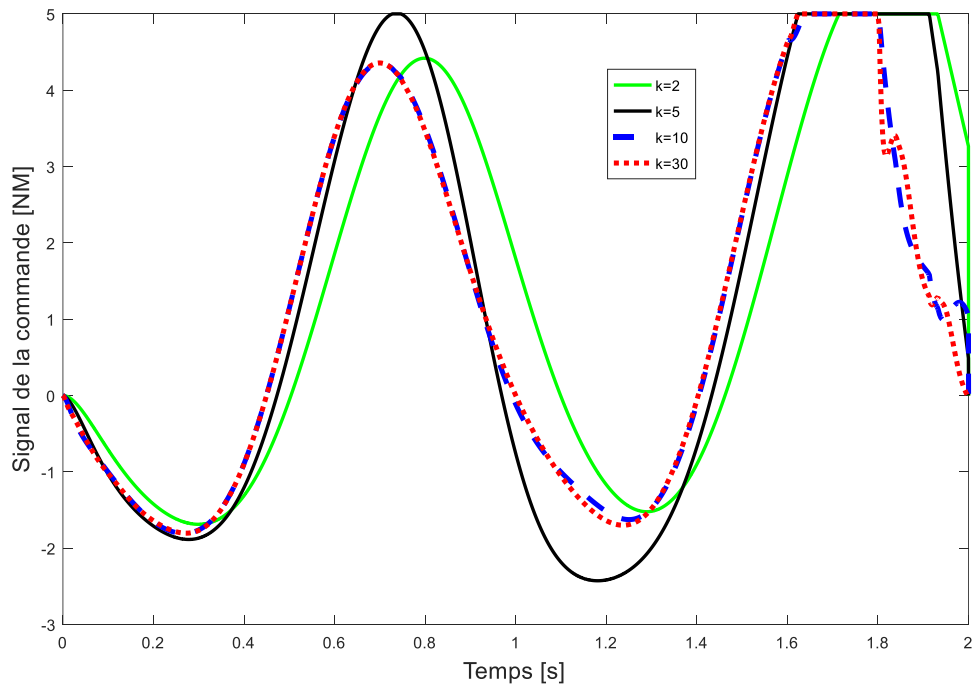


Figure IV.20 Deuxième signal de commande au cours des itérations.

IV.4 Conclusion

Dans ce chapitre, nous avons développé deux schémas de commandes par apprentissage itératif de type -P- pour étudier et surmonter le problème d'existence des non linéarités à l'entrée des systèmes répétitifs. Dans le premier schéma, nous avons présenté une loi de commande adaptative par apprentissage itératif de type -P- à base des réseaux de neurones à fonction de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des robots manipulateurs possédants des non-linéarités de type zone-morte à ses entrées. Dans la deuxième approche, une loi de commande par apprentissage itératif de Type -P- d'ordre élevée est proposée pour traiter le problème de poursuite des systèmes non linéaires multivariables MIMO en présence des saturations à leurs entrées. La méthode de Lyapunov et la méthode de λ -norme sont utilisées pour démontrer la stabilité des systèmes pour la première et la deuxième approche, respectivement. Pour montrer l'efficacité des approches proposées avec la présence des non linéarités à l'entrée, des résultats de simulation sont présentés

Chapitre. V.

Commande par apprentissage itératif à gains variables

V.1 Introduction

Parmi les difficultés rencontrées dans la commande par apprentissage itératif des systèmes répétitifs comme celles de la commande des systèmes standards, c'est le choix des gains de la commande proposée. Le problème d'optimisation de ces gains devient une autre direction dans le développement de la commande des systèmes non linéaire. A travers notre recherche sur la CAI, nous avons constaté que, les valeurs des gains de la commande sont fixes. Jusqu'à présent, il n'existe pas une technique bien précise pour trouver les gains optimaux et les valeurs sont choisies par la méthode essai-erreur dans les simulations. En fait, il y a quelques travaux qui ont traité ce problème. Par exemple, une condition sur les gains de commande a été présentée dans [MAD 08], mais cette condition reste insuffisante. Dans [WEI 11], une loi de commande avec une loi d'adaptation des gains sont présentées pour une classe des systèmes satisfaisant la condition de continuité globale de Lipschitz. Dans [YAN 15], les valeurs minimales de la commande des systèmes linéaires sont ajustées par un algorithme génétique combiné avec les réseaux de neurones, cette méthode est hors ligne et elle a nécessité une grande base de données.

D'autre part, parmi les méthodes stochastiques d'optimisation modernes utilisées dans ce sens, l'optimisation par essaim de particules (PSO). Cette technique a été appliquée dans de nombreux problèmes pour différents domaine d'applications [HAN 18], [GAO 18] et [ZOU 19]. L'algorithme a prouvé son efficacité pour résoudre ces problèmes. Cela nous a motivé d'utiliser l'algorithme de PSO pour trouver les valeurs optimales de la commande proposée.

Dans ce chapitre, deux approches de commandes adaptatives par apprentissage itératif à gains variables, sont proposées pour résoudre le problème de poursuite des systèmes non linéaires multivariables sans utiliser la condition globale de Lipschitz, d'où un approximateur de type

logique floue a été utilisé. Dans le premier schéma, les valeurs optimales des gains peuvent être déterminées par l'algorithme d'optimisation PSO (l'optimisation par essaim de particules) [BEN 19], alors que dans la seconde approche, les gains sont ajustés en ligne à travers les itérations via une loi d'adaptation proposée. La stabilité du système en boucle fermée est vérifiée en utilisant la méthode de Lyapunov. A la fin de ce chapitre, des résultats de simulation sont présentés pour montrer l'efficacité des méthodes proposées.

V.2 Position du problème

Nous considérons le système non linéaire suivant :

$$\dot{x}_k(t) = f(x_k(t), t) + B(x_k(t), t)u_k(t) \quad (\text{V.1})$$

où :

$x_k \in R^n, u_k \in R^n$ représentent le vecteur d'état et la commande, respectivement.

$f(*) = [f_1(*), \dots, f_n(*)] \in R^n$ est une fonction non linéaire inconnue, $B \in R^{n \times n}$ est une matrice de commande inconnue. $f(x_k(t))$ et $B(x_k(t))$ sont bornées par des fonctions inconnues :

$$\|B^{-1}(x_k(t))(\dot{x}_d(t) - f(x_k(t)))\| \leq \rho(x_k(t)) \quad (\text{V.2})$$

En utilisant la méthode d'approximation floue, la fonction $\rho(x_k(t))$ peut-être ajustée en ligne comme suit

$$\rho(x_k) = w_k^T(t)\varphi(x_k) + \varepsilon(x_k) \quad (\text{V.3})$$

où w_k est le vecteur des paramètres idéaux de la logique floue, ε est l'erreur d'approximation supposée bornée et très petite et $\varphi(x_k(t))$ est le vecteur des fonctions de base connu

$$\varphi_i(x_k) = \frac{\prod_{l=1}^n \mu_{l,j}(x_k)}{\sum_{k=1}^m \prod_{l=1}^n \mu_{l,j}(x_k)} \quad (\text{V.4})$$

d'où : m est le nombre des règles floues, n est le nombre des entrées pour chaque règle et $\mu_{l,j}(x_k(t))$ est la fonction d'appartenance. Dans ce travail, $\mu_{l,j}(x_k(t))$ est choisie de type fonction de base radiale.

Nous définissons \hat{w}_k comme l'estimation de w_k à travers les itérations, la loi d'adaptation proposée est donnée par

$$\hat{w}_k(t) = \hat{w}_{k-1}(t) + \gamma e_k^T(t) \operatorname{sgn}(e_k(t)) \varphi(x_k, t) \quad (\text{V.5})$$

Avec γ est un gain positif. $e_k(t) = x_d(t) - x_k(t)$ est l'erreur de poursuite.

L'erreur d'estimation est définie par

$$\tilde{w}_k(t) = w_k(t) - \hat{w}_k(t) \quad (\text{V.6})$$

La dérivée de l'erreur de poursuite

$$\dot{e}_k(t) = \dot{x}_d(t) - \dot{x}_k(t) \quad (\text{V.7})$$

A partir de l'équation (V.1), La dynamique des erreurs peut s'écrire

$$\dot{e}_k(t) = \dot{x}_d(t) - f(x_k(t), t) - B(x_k(t), t)u_k(t) \quad (\text{V.8})$$

Les hypothèses suivantes sont nécessaires.

Hypothèse V.1 : A chaque itération, les conditions initiales sont les mêmes, i.e., $x_k(0) = x_d(0)$.

Hypothèse V.2: La matrice d'entrée $B(x_k(t), t)$ est définie positive $\forall t \in [0, T]$, sans perte de généralité, nous supposons ici que $\dot{B}(x_k(t), t)$ est bornée.

Remarque V.1 : Dans la littérature, l'hypothèse V.2 est largement utilisée dans des nombreux travaux sur la CAI [Ji 16] et [ZHA 16].

Notre objectif est de proposer une loi de commande par apprentissage itératif à base de la logique floue $u_k(t)$ avec des gains variables qui sont ajustés en ligne, afin de forcer la sortie réelle $x_k(t)$ à suivre une trajectoire désirée $x_d(t)$, sur un intervalle de temps fixe $t \in [0, T]$, lorsque k tend vers l'infini.

V.3 Synthèse d'une loi de CAI Type -P-

Dans cette partie, nous proposons une loi de commande par apprentissage itératif à base de la logique floue, elle est donnée comme suit

$$u_k(t) = K_P e_k(t) + \hat{\rho}(x_k(t)) \operatorname{sgn}(e_k(t)) \quad (\text{V.9})$$

K_p est le gain proportionnel et $\hat{\rho}(x_k)$ est l'estimation de $\rho(x_k)$.

Théorème V.1 [BEN 19] : Considérons le système non linéaire MIMO incertains (V.1), vérifiant l'hypothèse V.1 et V.2, en appliquant la loi de commande (V.9) avec la loi d'adaptation à base de la logique floue (V.5), si

$$\lambda_{\min}(K_p) \geq \frac{1}{2} \lambda_{\min}(\dot{B}^{-1}(x_k, t)) \quad (\text{V.10})$$

Alors

$$\text{v) } e_k(t) \in L_{\infty e}[0, T] \text{ et } \hat{w}_k(t) \in L_{\infty e}[0, T], \forall k \in N.$$

$$\text{vi) } \lim_{k \rightarrow \infty} e_k(t) = 0, \forall t \in [0, T].$$

Preuve

La preuve est basée sur la théorie de Lyapunov comme nous avons fait dans les chapitres précédents.

Partie 1 : Soit la fonction de Lyapunov suivante

$$V_k(t) = \frac{1}{2} e_k^T B_k^{-1} e_k + \frac{1}{2\gamma} \int_0^t \tilde{w}_k^T \tilde{w}_k d\tau \quad (\text{V.11})$$

Définissons ΔV_k

$$\Delta V_k = V_k - V_{k-1} \quad (\text{V.12})$$

De (V.11), il vient

$$\Delta V_k(t) = \frac{1}{2} e_k^T B_k^{-1} e_k - \frac{1}{2} e_{k-1}^T B_{k-1}^{-1} e_{k-1} + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \quad (\text{V.13})$$

Cela implique que

$$\Delta V_k(t) \leq \frac{1}{2} e_k^T B_k^{-1} e_k + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \quad (\text{V.14})$$

En utilisant l'hypothèse V.1, (V.13) peut être réécrite comme suit

$$\Delta V_k(t) \leq \int_0^t \left(e_k^T B_k^{-1} \dot{e}_k + \frac{1}{2} e_k^T \dot{B}_k^{-1} e_k \right) d\tau + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \quad (\text{V.15})$$

En remplaçant (V.8) et (V.9) dans (V.15), on trouve

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(e_k^T B_k^{-1} (\dot{x}_d - f_k) - e_k^T K_p e_k + \hat{\rho}_k e_k^T \text{sgn}(e_k) + \frac{1}{2} e_k^T \dot{B}_k^{-1} e_k \right) d\tau \\ & + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \end{aligned} \quad (\text{V.16})$$

Sachant que $\tilde{\rho}_k = \rho_k - \hat{\rho}_k$, utilisant (V.2), (V.15) devient

$$\begin{aligned} \Delta V_k(t) \leq & \int_0^t \left(\tilde{\rho}_k e_k^T \text{sgn}(e_k) - e_k^T K_p e_k + \frac{1}{2} e_k^T \dot{B}_k^{-1} e_k \right) d\tau \\ & + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \end{aligned} \quad (\text{V.17})$$

A partir des équations (V.5) et (V.6), on trouve

$$\tilde{w}_k = \tilde{w}_{k-1} - \gamma e_k^T \text{sgn}(e_k) \varphi_k \quad (\text{V.18})$$

D'autre part, en utilisant l'équation (V.18), le dernier terme de l'équation (V.17) devient

$$\begin{aligned} & \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \\ & = \frac{1}{2\gamma} \int_0^t ((\tilde{w}_{k-1} - \gamma e_k^T \text{sgn}(e_k) \varphi_k)^T \tilde{w}_k \\ & \quad - (\tilde{w}_k + \gamma e_k^T \text{sgn}(e_k) \varphi_k)^T \tilde{w}_{k-1}) d\tau \end{aligned} \quad (\text{V.19})$$

Comme $\tilde{w}_{k-1}^T \tilde{w}_k = \tilde{w}_k^T \tilde{w}_{k-1}$, alors

$$\begin{aligned} & \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \\ & = \frac{1}{2\gamma} \int_0^t (-(\gamma e_k^T \text{sgn}(e_k) \varphi_k)^T \tilde{w}_k - (\gamma e_k^T \text{sgn}(e_k) \varphi_k)^T \tilde{w}_{k-1}) d\tau \end{aligned} \quad (\text{V.20})$$

En remplaçant (V.18) dans (V.20), il vient

$$\frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \leq \frac{1}{2\gamma} \int_0^t (-2\gamma e_k^T \text{sgn}(e_k) \varphi_k)^T \tilde{w}_k d\tau \quad (\text{V.21})$$

Sachant que $\varphi_k^T \tilde{w}_k = \tilde{w}_k^T \varphi_k = \tilde{\rho}_k$, donc

$$\frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \leq - \int_0^t \tilde{\rho}_k e_k^T \text{sgn}(e_k) d\tau \quad (\text{V.22})$$

En remplaçant (V.22) dans (V.17), on obtient

$$\Delta V_k \leq \int_0^t \left(-e_k^T K_p e_k + \frac{1}{2} e_k^T \dot{B}_k^{-1} e_k \right) d\tau. \quad (\text{V.23})$$

En appliquant le théorème de Rayleigh-Ritz sur (V.23), on trouve

$$\Delta V_k \leq \int_0^t \left(\left(-\lambda_{\min}(K_p) + \frac{1}{2} \lambda_{\max}(\dot{B}_k^{-1}) \right) \|e_k\|^2 \right) d\tau. \quad (\text{V.24})$$

Si on choisit $\lambda_{\min}(K_p) \geq \frac{1}{2} \lambda_{\max}(\dot{B}_k^{-1})$, alors

$$\Delta V_k \leq 0 \quad (\text{V.25})$$

Cela implique que V_k est décroissante à travers les itérations.

Partie 2 : Montrons que V_0 est bornée $\forall t \in [0, T]$.

De (V.11), avec $k=0$, il vient

$$V_0(t) = \frac{1}{2} e_0^T B_0^{-1} e_0 + \frac{1}{2\gamma} \int_0^t \tilde{w}_0^T \tilde{w}_0 d\tau. \quad (\text{V.26})$$

Par dérivation, on trouve

$$\dot{V}_0(t) = e_0^T B_0^{-1} \dot{e}_0 + \frac{1}{2} e_0^T \dot{B}_0^{-1} e_0 + \frac{1}{2\gamma} \tilde{w}_0^T \tilde{w}_0 \quad (\text{V.27})$$

En remplaçant (V.8) et (V.9) dans (V.27), on obtient

$$\dot{V}_0(t) = e_0^T B_0^{-1} (\dot{x}_d - f_0) - e_0^T K_p e_0 + \hat{w}_0 e_0^T \text{sgn}(e_0) + \frac{1}{2} e_0^T \dot{B}_0^{-1} e_0 + \frac{1}{2\gamma} \tilde{w}_0^T \tilde{w}_0 \quad (\text{V.28})$$

En utilisant (V.2), (V.28) devient

$$\dot{V}_0(t) = \tilde{\rho}_0 e_0^T \text{sgn}(e_0) - e_0^T K_p e_0 + \frac{1}{2} e_0^T \dot{B}_0^{-1} e_0 + \frac{1}{2\gamma} \tilde{w}_0^T \tilde{w}_0 \quad (\text{V.29})$$

À partir de (V.18) avec $k=0$ et sachant que $\tilde{\rho}_0 = \tilde{w}_0^T \varphi_0 = \varphi_0^T \tilde{w}_0$, alors

$$\dot{V}_0(t) = -e_0^T K_p e_0 + \frac{1}{2} e_0^T \dot{B}_0^{-1} e_0. \quad (\text{V.30})$$

En appliquant le théorème de Rayleigh-Ritz sur (V.30), on trouve

$$\dot{V}_0(t) \leq \left(-\lambda_{\min}(K_p) + \frac{1}{2} \lambda_{\max}(\dot{B}_0^{-1}) \right) \|e_0\|^2. \quad (\text{V.31})$$

Si on choisit $K_p \geq \frac{1}{2} \lambda_{\max}(\dot{B}_0^{-1})$, alors

$$\dot{V}_0(t) \leq 0 \quad (\text{V.32})$$

Cela implique que \dot{V}_0 est bornée, donc V_0 est bornée.

Partie 3 : Notons que $V_k(t)$ peut être représentée comme suit

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_j(t) \quad (\text{V.33})$$

En utilisant (V.23), on obtient

$$V_k(t) \leq V_0(t) - \sum_{j=1}^k \int_0^t \left(e_k^T \left(K_p - \frac{1}{2} \dot{B}_k^{-1} \right) e_k \right) d\tau. \quad (\text{V.34})$$

Cela implique

$$\sum_{j=1}^k \int_0^t \left(e_k^T \left(K_p - \frac{1}{2} \dot{B}_k^{-1} \right) e_k \right) d\tau \leq V_0(t) - V_k(t) \leq V_0(t) \quad (\text{V.35})$$

Comme $\left(K_p - \frac{1}{2} \dot{B}_k^{-1} \right)$, alors

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \forall t \in [0, T] \quad (\text{V.36})$$

V.4 Synthèse d'une loi de CAI à gains adaptatifs

Dans cette section, nous proposons une loi de commande par apprentissage itératif de type -P- dans laquelle le gain proportionnel associé est ajusté en ligne via une loi d'adaptation, d'où :

$$\begin{cases} u_k(t) = \hat{P}_k(t)e_k(t) + \hat{\rho}_k(x)\text{sgn}(e_k(t)) \\ \hat{P}_k(t) = \hat{P}_{k-1}(t) + \beta e_k^T(t)e_k(t) \end{cases} \quad (\text{V.37})$$

Où : la fonction $\hat{\rho}_k(x)$ est définie dans la partie précédente. $\hat{P}_{-1}(t) = 0$.

L'erreur d'estimation de la loi d'adaptation est présentée comme

$$\tilde{P}_k(t) = P(t) - \hat{P}_k(t) \quad (\text{V.38})$$

L'hypothèse suivante est utilisée dans le reste de ce travail.

Théorème V.2 : Considérons le système non linéaire MIMO incertain (V.1), vérifiant les hypothèses V.1 et V.2, en appliquant la loi de commande (V.37) avec la loi d'adaptation à base de la logique floue (V.5), alors on a

- i) $e_k(t) \in L_{\infty}[0, T]$ Et $\hat{w}_k(t) \in L_{\infty}[0, T]$ $\hat{P}_k(t) \in L_{\infty}[0, T]$
- ii) $\lim_{k \rightarrow \infty} e_k(t) = 0, \forall t \in [0, T]$

Preuve : La preuve est basée sur la méthode de Lyapunov.

Partie 1 : Soit la fonction de Lyapunov suivante

$$V_k(t) = \frac{1}{2} e_k^T B_k^{-1} e_k + \frac{1}{2\gamma} \int_0^t \tilde{w}_k^T \tilde{w}_k d\tau + \frac{1}{2\beta} \int_0^t \tilde{P}_k^2 d\tau. \quad (\text{V.39})$$

Nous utilisons les mêmes étapes de la section précédente, d'où, à partir de (V.37) et (V.38), il vient

$$\begin{aligned} \Delta V_k(t) &\leq \int_0^t (\tilde{\rho}_k e_k^T \text{sgn}(e_k) - \hat{P}_k e_k^T e_k) d\tau + \frac{1}{2\gamma} \int_0^t (\tilde{w}_k^T \tilde{w}_k - \tilde{w}_{k-1}^T \tilde{w}_{k-1}) d\tau \\ &\quad + \frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau \end{aligned} \quad (\text{V.40})$$

D'autre part, le dernier terme de (V.40) peut être réécrit comme

$$\frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau = \frac{1}{2\beta} \int_0^t (\tilde{P}_k + \tilde{P}_{k-1})(\tilde{P}_k - \tilde{P}_{k-1}) d\tau \quad (\text{V.41})$$

Sachant que $\tilde{P}_k = P - \hat{P}_k$, alors

$$\frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau = \frac{1}{2\beta} \int_0^t -(\hat{P}_k + \hat{P}_{k-1})(-(\tilde{P}_k - \tilde{P}_{k-1}) + 2\tilde{P}_k) d\tau \quad (\text{V.42})$$

Ce qui implique que

$$\frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau = \frac{1}{2\beta} \int_0^t \left(-(\hat{P}_k - \hat{P}_{k-1})^2 - 2\tilde{P}_k(\hat{P}_k - \hat{P}_{k-1}) \right) d\tau \quad (\text{V.43})$$

En remplaçant (V.37) dans (V.43), on trouve

$$\frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau = \frac{1}{2\beta} \int_0^t -\beta^2 (e_k^T e_k)^2 - 2\beta \tilde{P}_k e_k^T e_k d\tau \quad (\text{V.44})$$

En utilisant (V.38), on obtient

$$\frac{1}{2\beta} \int_0^t (\tilde{P}_k^2 - \tilde{P}_{k-1}^2) d\tau \leq \int_0^t \left(\hat{P}_k e_k^T e_k - \frac{\beta}{2} (e_k^T e_k)^2 \right) d\tau \quad (\text{V.45})$$

En remplaçant (V.22) et (V.45) dans (V.40), on trouve

$$\Delta V_k \leq -\frac{\beta}{2} \int_0^t (e_k^T e_k)^2 d\tau \quad (\text{V.46})$$

Donc

$$\Delta V_k \leq 0 \quad (\text{V.47})$$

Partie 2 : Montrons, maintenant, que V_0 est bornée $\forall t \in [0, T]$

A partir de (V.39) avec $k=0$, on a

$$V_0(t) = \frac{1}{2} e_0^T B_0^{-1} e_0 + \frac{1}{2\gamma} \int_0^t \tilde{w}_0^T \tilde{w}_0 d\tau + \frac{1}{2\beta} \int_0^t \tilde{P}_0^2 d\tau \quad (\text{V.48})$$

De la même manière utilisée dans la preuve du théorème V.1. On obtient

$$\dot{V}_0(t) = \tilde{\rho}_0 e_0^T \text{sgn}(e_0) - \hat{P}_0 e_0^T e_0 + \frac{1}{2\gamma} \tilde{w}_0^T \tilde{w}_0 + \frac{1}{2\beta} \tilde{P}_0^2 \quad (\text{V.49})$$

À partir de (V.40) avec $k=0$, alors

$$\frac{1}{2\beta} \tilde{P}_0^2 = \frac{1}{2\beta} (P_0 - \hat{P}_0)^2 \quad (\text{V.50})$$

L'équation (V.50) peut être réécrite

$$\frac{1}{2\beta} \tilde{P}_0^2 = \frac{1}{2\beta} (-2(P_0 - \hat{P}_0)\hat{P}_0 - \hat{P}_0^2 + P_0^2) \quad (\text{V.51})$$

Ce qui nous donne

$$\frac{1}{2\beta} \tilde{\alpha}_0^2 \leq \hat{P}_0 e_0^T e_0 + \frac{1}{2\beta} P_0^2 \quad (\text{V.52})$$

En remplaçant (V.28) et (V.52) dans (49), on trouve

$$\dot{V}_0(t) \leq \frac{1}{2\beta_2} P^2 \quad (\text{V.53})$$

Il est clair que \dot{V}_0 est bornée, donc V_0 est bornée.

Partie 3 : Notons que $V_k(t)$ peut être représentée comme suit

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_k(t) \quad (\text{V.54})$$

En utilisant (46), il vient

$$V_k \leq V_0 - \frac{\beta}{2} \sum_{j=1}^k \int_0^t \|e_k\|^2 d\tau \quad (\text{V.55})$$

Cela implique que

$$\sum_{j=1}^k \int_0^t \|e_k\|^2 d\tau \leq \frac{\beta}{2} V_0 \quad (\text{V.56})$$

Comme $\frac{\beta}{2} V_0$ est bornée, alors

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \forall t \in [0, T] \quad (\text{V.57})$$

V.5 Résultats de simulation

Considérons un robot manipulateur à deux degrés de liberté. Son modèle dynamique est défini sous la forme [TAY 04] :

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} C_{11}\dot{q}_1 + C_{12}\dot{q}_2 + G_1 - d_1 \\ C_{21}\dot{q}_1 + C_{22}\dot{q}_2 + G_2 - d_2 \end{bmatrix} + \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \right\} \quad (\text{V.58})$$

Où

$$M_{11} = m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_2)) + I_1 + I_2,$$

$$M_{12} = M_{21} = m_2 (l_{c2}^2 + l_1 l_{c2} \cos(q_2)) + I_2,$$

$$M_{22} = m_2 l_{c2}^2 + I_2, C_{11} = h\dot{q}_2, C_{12} = h\dot{q}_1 + h\dot{q}_2, C_{21} = -h\dot{q}_1,$$

$$C_{22} = 0, h = -m_2 l_1 l_{c2} \sin(q_2),$$

$$G_1 = (m_1 l_{c1} + m_2 l_1) g \cos(q_1) + m_2 l_{c2} g \cos(q_1 + q_2),$$

$$G_2 = m_2 l_{c2} g \cos(q_1 + q_2).$$

Les paramètres suivants sont donnés par

$$m_1 = m_2 = 1[\text{Kg}], l_{c1} = l_{c2} = 0.25[\text{m}], \quad l_1 = l_2 = 0.5[\text{m}], \quad I_1 = I_2 = 0.1[\text{Kg.m}^3], \quad g = 9.81 \left[\frac{\text{m}}{\text{s}^2} \right].$$

Le vecteur de perturbations est donné par

$$\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} 13t - \frac{1}{4}t^2 \\ 5t - \frac{1}{4}t^2 \end{bmatrix} \quad (\text{V.59})$$

Le vecteur des trajectoires désirées est donné par

$$\begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} = \begin{bmatrix} \sin(2\pi t) \\ \cos(2\pi t) \end{bmatrix}; t \in [0,2] \quad (\text{V.60})$$

V.5.1 Cas 1 : Utilisation de l'algorithme de PSO

En appliquant la loi de commande proposée (V.9) avec la loi d'adaptation (V.5), deux ensembles flous sont utilisés. On choisit la fonction d'appartenance comme $\mu_{l,j}(x_k) + \exp[-a_{l,j}(x_j - b_{l,j})^2]$, avec $a_{j,l} = 0.5b_{j,l} = [1.5, 1.5, -1.5, -1.5]$, K_p et γ sont ajustées via l'algorithme de PSO comme $X = [K_p, \gamma] \times n \times 2$, tel que, n=nombre de population. La valeur minimale pour les deux paramètres est fixée à 0. La valeur maximale pour les deux paramètres est fixée à 200. La taille de population=10, le facteur minimal et le facteur maximal des poids d'inertie sont choisis $\omega_{max} = 1$ $\omega_{min} = C = 0.3$. La fonction de cout est la suivante $C = \min(\text{sqrt}(e_1^2) + \text{sqrt}(e_2^2))$.

Par la suite, et en utilisant MATLAB, la valeur minimale de la fonction de cout et les valeurs optimales sont obtenues par l'algorithme de PSO comme $C_{optimal} = 0.0073$, $K_{p,optimal} = 145.3666$ et $\gamma_{optimal} = 111.1914$. Avec ces paramètres, les résultats de simulation sont présentés dans les figures V.1, V.2, V.3, V.4 et V.5. Les figures V.1 et V.2 présentent les trajectoires réelles et désirées à l'itération 20 pour la première et la deuxième articulation, respectivement. Il est clair que les trajectoires réelles suivent parfaitement les trajectoires désirées pour les deux articulations. Les figures V.3 et V.4 présentes les erreurs de poursuite à

la 1^{ère}, 2^{ème}, 5^{ème} et 20^{ème} itérations pour la première et la deuxième articulation, respectivement. La figure 5 présente La norme-Sup des erreurs à travers les itérations pour les deux articulations, on peut voir que les erreurs diminuent à travers les itérations. Une comparaison de La norme-Sup des erreurs pour différentes valeurs des gains à l'itération 20^{ème} est montrée dans le tableau V.1.

V.5.2 Cas 2 : Utilisation de la loi d'adaptation des gains

En appliquant la loi de commande proposée (V.37) avec la loi d'adaptation (V.5). Les paramètres de la logique floue sont choisis comme dans le premier cas, $\gamma = 15, \beta = 1.5$ et $t \in [0,2]$. Les résultats de simulation sont donnés dans les figures V.6, V.7, V.8, V.9 et V.10.

Les figures V.6 et V.7 présentent les trajectoires réelles et désirées à la 2^{ème}, 5^{ème} et la 20^{ème} itérations pour la première et la deuxième articulation, respectivement. On peut voir que les trajectoires réelles suivent mieux les trajectoires désirées à travers les itérations. La norme sup-des erreurs à travers les itérations pour les deux articulations sont montrées dans les figures V.8 et V.9, tel que, le maximum des erreurs de la première et la deuxième articulation après 20 itérations sont inférieurs à 0.02120 et 0.01152, respectivement. Donc, le maximum de l'erreur est réduit plus de 68 fois et 52 fois pour la première et la deuxième articulation, respectivement. L'évolution de la norme-Sup du gain proportionnel P est montrée dans la figure V.10. Il est clair que cette norme atteint la valeur 9.6828 après 6 itérations. A partir de ces résultats, nous pouvons conclure que les deux méthodes proposées fonctionnent bien.

Par comparaison des performances obtenues à partir des deux méthodes, nous remarquons qu'après 20 itérations le maximum des erreurs ($\max(|e_1(t)| + |e_2(t)|)$) est égal à 0,0073 et 0,032 pour la méthode de PSO et la méthode adaptative, respectivement. Par conséquent, on peut conclure que la méthode de PSO est meilleure en termes de rapidité de convergence que la deuxième approche.

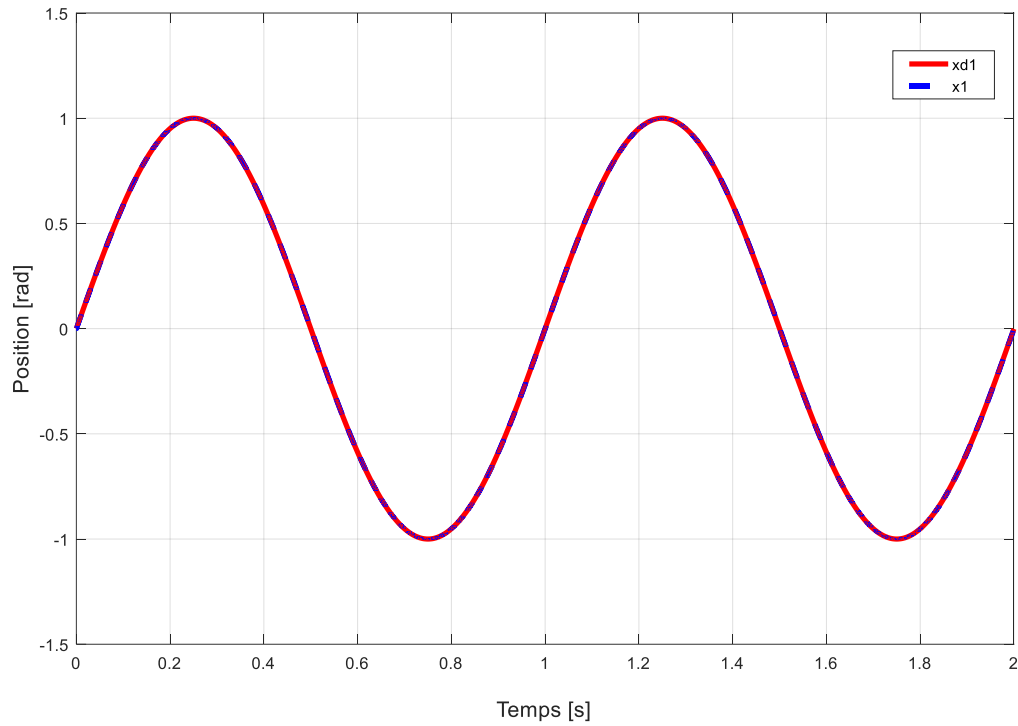


Figure V.1 Trajectoires réelle et désirée de la première articulation pour la 20^{ème} itérations.

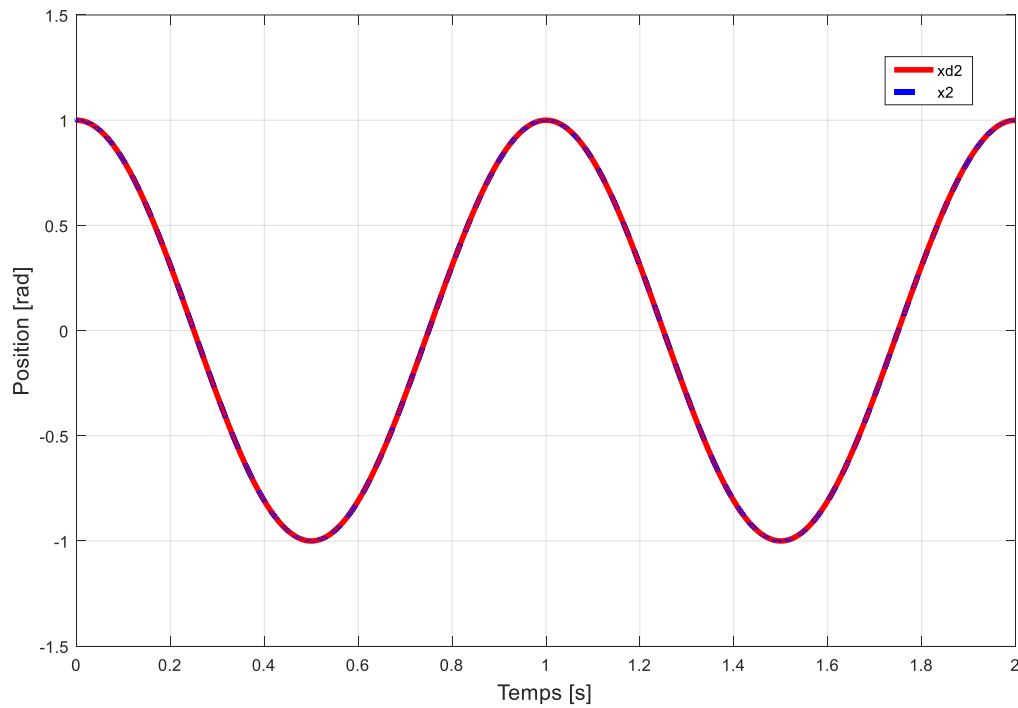


Figure V.2 Trajectoires réelle et désirée de la deuxième articulation pour la 20^{ème} itérations.

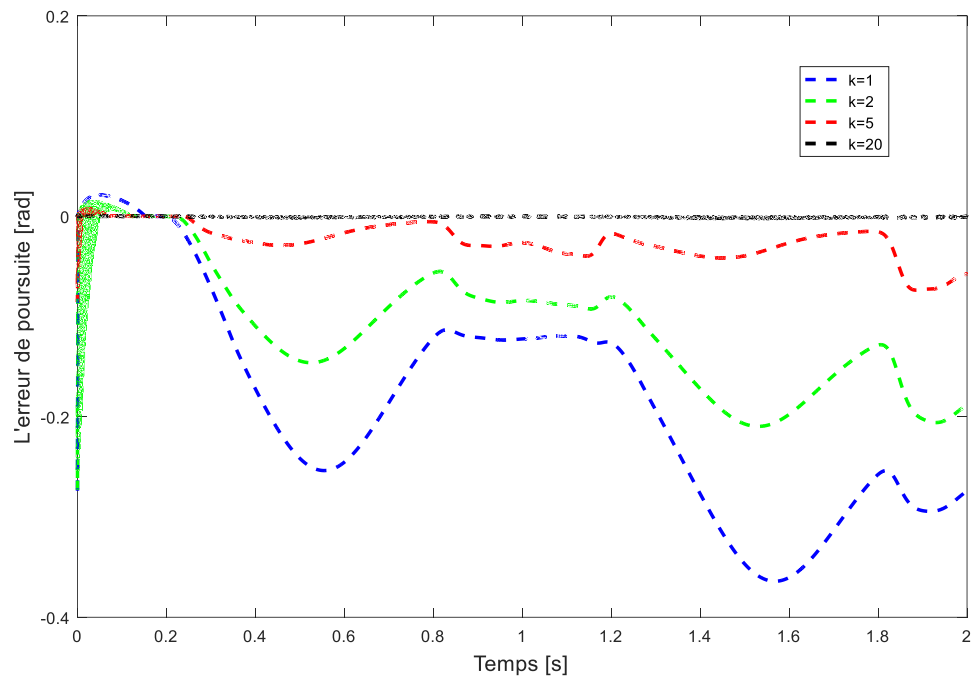


Figure V.3 Erreurs de poursuite de la première articulation.

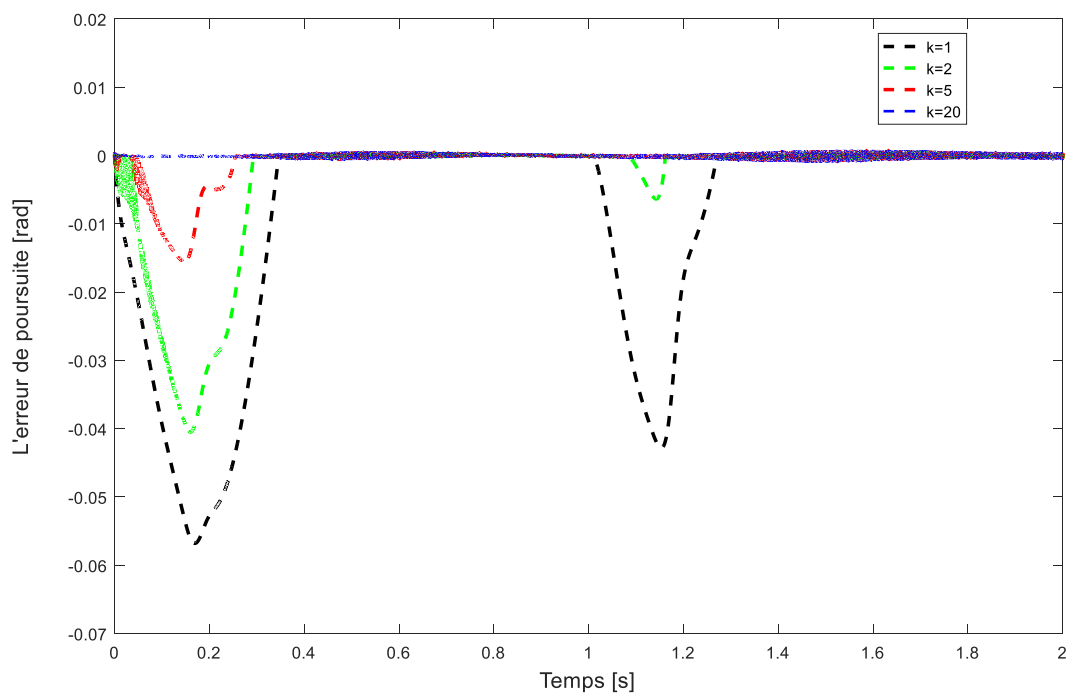


Figure V.4 Erreurs de poursuite de la deuxième articulation.

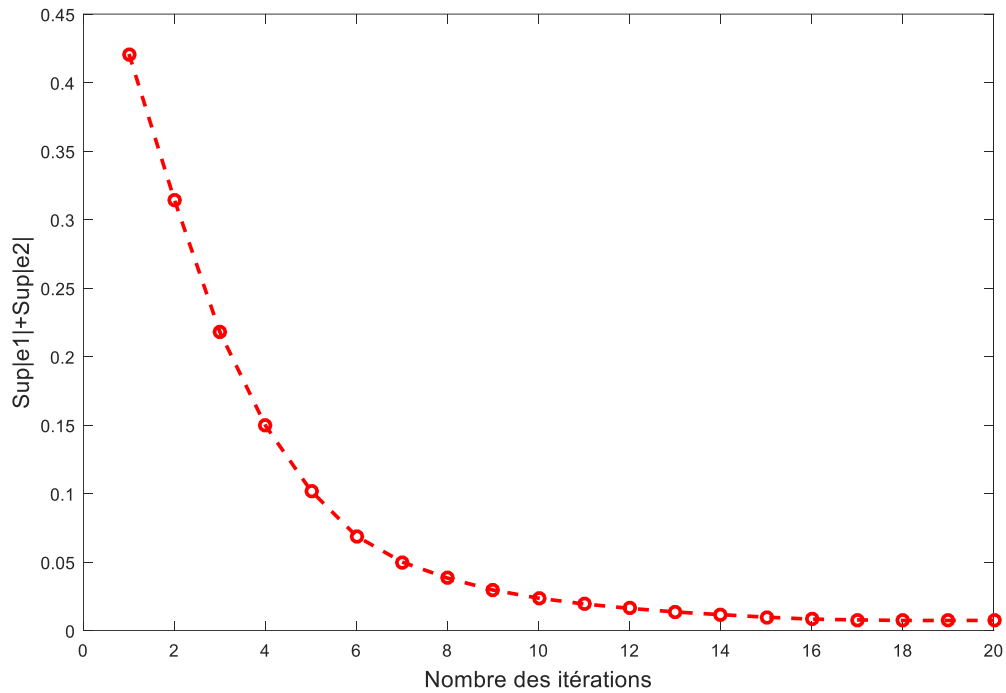


Figure V.5 Norme sup des erreurs de la première et la deuxième articulation au cours des itérations.

Table V.1 Valeurs de sup ($\|e_1\| + \|e_2\|$) pour des différents gains

Gains (K_p, γ)	5, 2	30,15	100,90	170,150	200,200	PSOILFC
Max($ e_1 + e_2 $)	1.286	0.119	0.009	0.0078	0.0079	0.0073

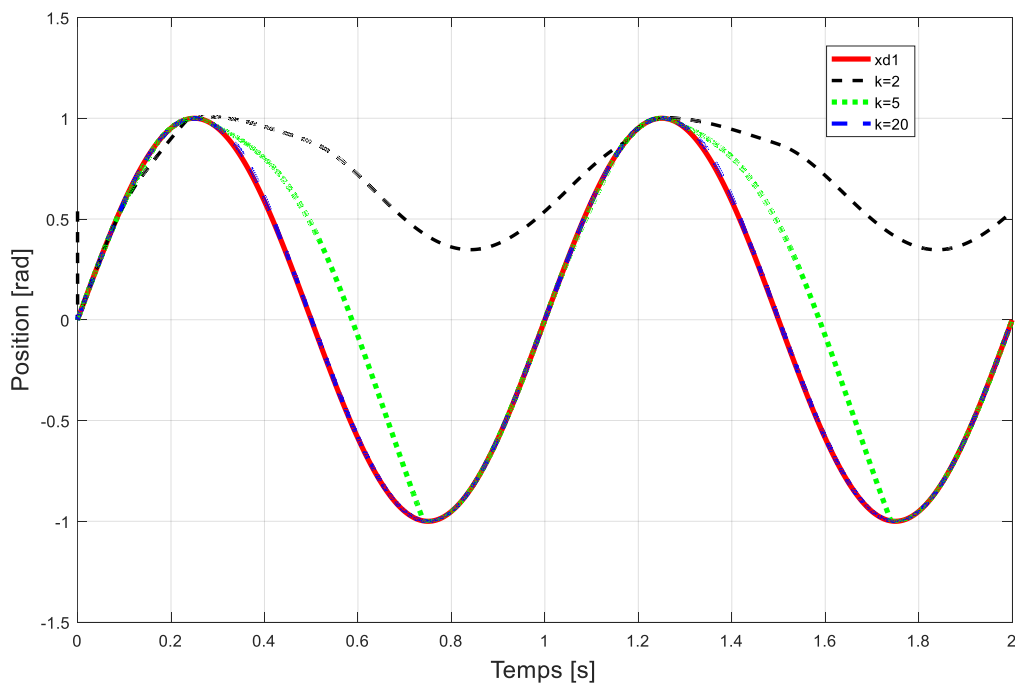


Figure V.6 Trajectoires réelles et désirée de la première articulation.

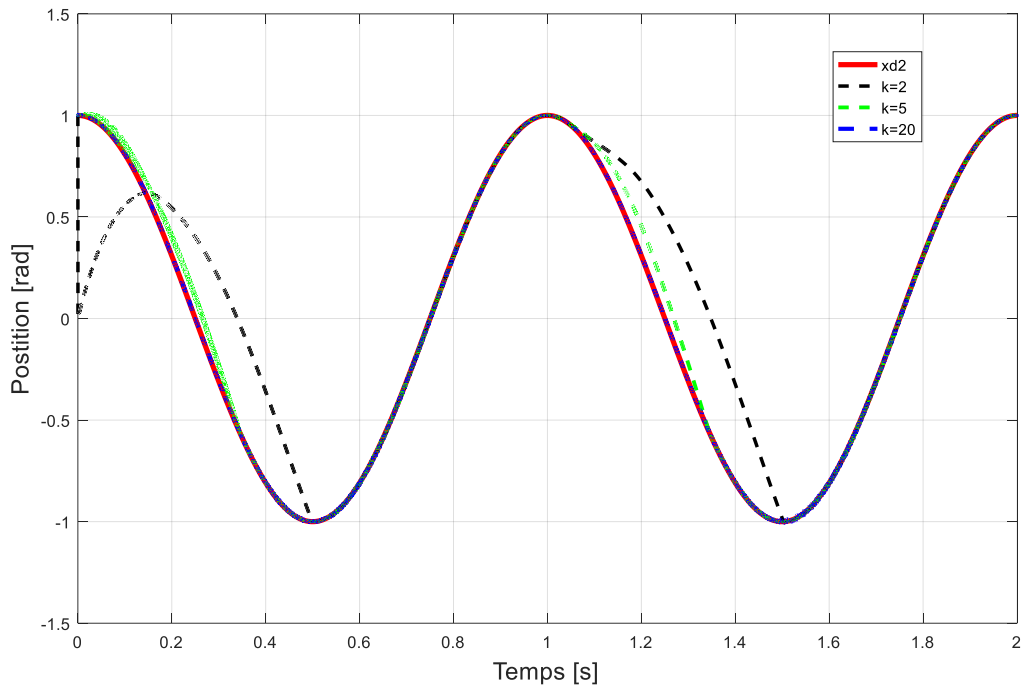


Figure V.7 Trajectoires réelles et désirée de la deuxième articulation.

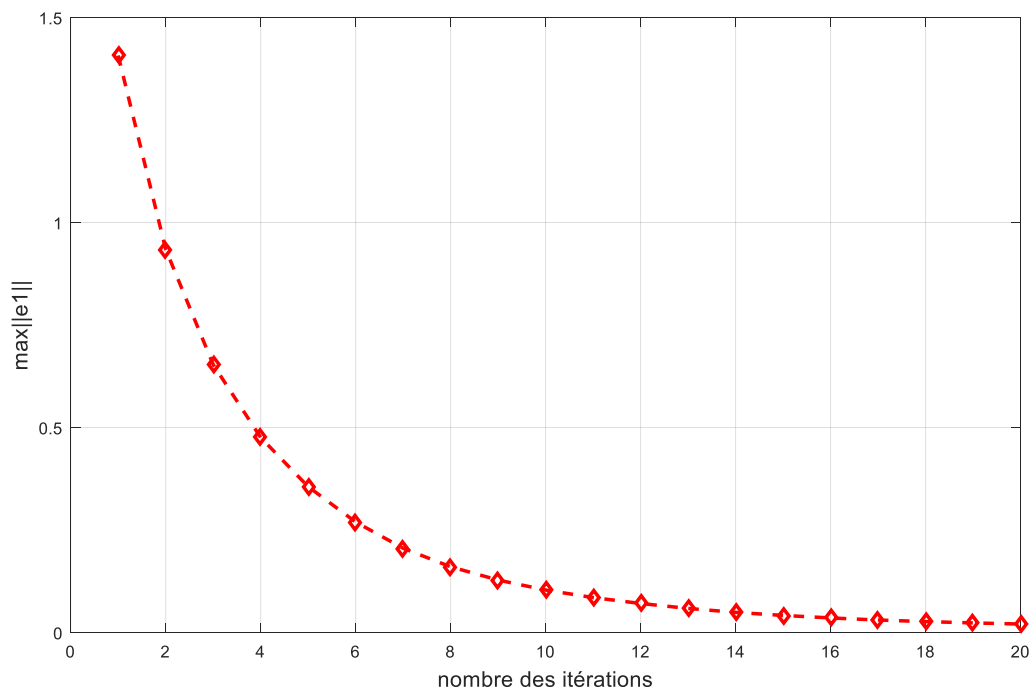


Figure V.8 Norme sup de l'erreur de la première articulation au cours des itérations.

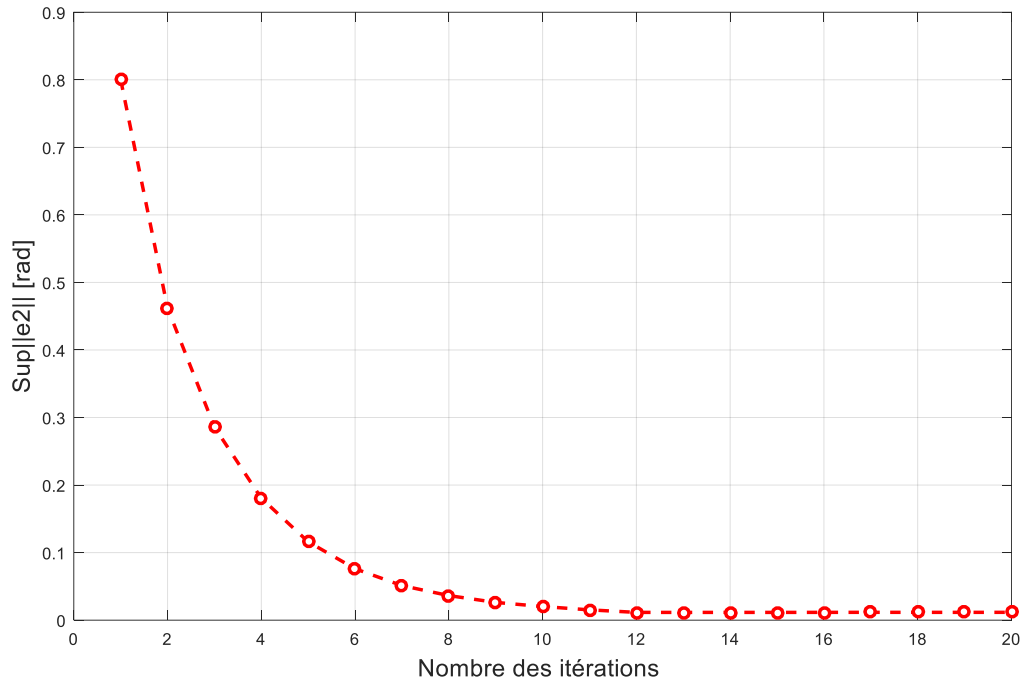


Figure V.9 Norme sup de l'erreur de la deuxième articulation au cours des itérations.

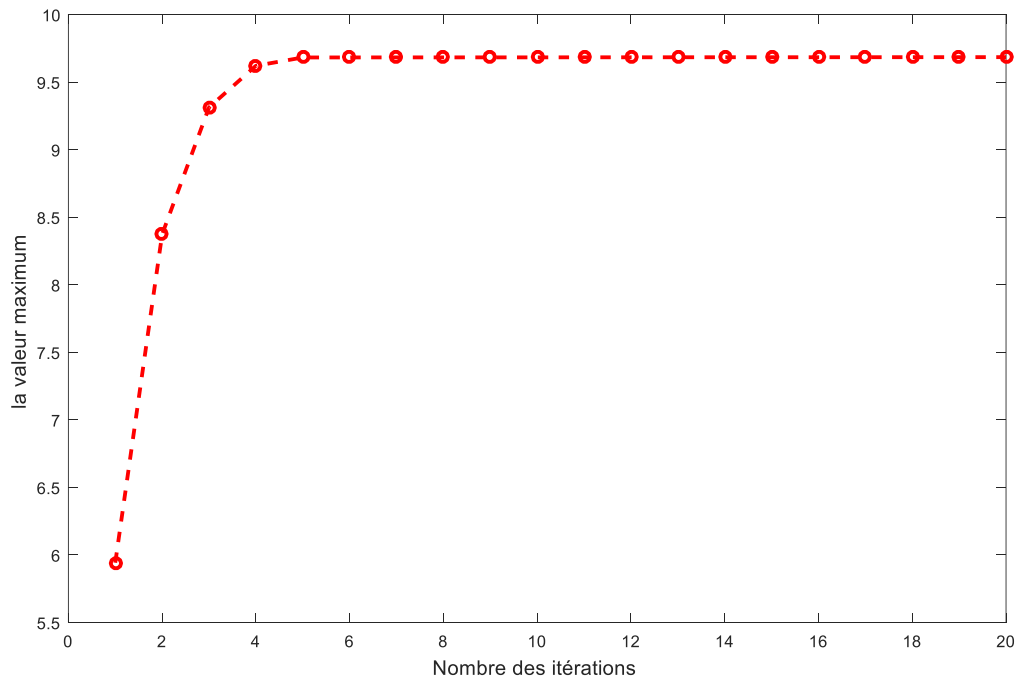


Figure V.10 Norme sup de la valeur de \hat{P}_k au cours des itérations.

V.6 Conclusion

Dans ce chapitre, nous avons traité le problème de choix des gains de la commande par apprentissage itératif (CAI). Pour ce faire, deux approches de CAI des systèmes MIMO non-linéaires ont été présentées, dans lesquelles, un approximateur universel à base de la logique floue est utilisé pour estimer les fonctions non linéaires inconnues du système. Dans le premier schéma, un algorithme d'optimisation de type PSO (l'optimisation par essaim de particules) est utilisé pour trouver les valeurs optimales de la loi de commande proposée. Dans le deuxième schéma, les valeurs de la loi de commande sont ajustées en ligne à travers les itérations via une loi d'adaptation proposée. Par ailleurs, nous avons montré la stabilité du système pour les deux lois de commande par l'utilisation de la théorie de Lyapunov. A la fin, des résultats de simulations sont effectués sur un robot manipulateur à deux degrés de liberté. D'après ces résultats de simulation, nous constatons que les deux méthodes proposées sont efficaces et fonctionnent bien et nous avons remarqué que la méthode de PSO est meilleure en termes de rapidité de convergence que la deuxième méthode.

Chapitre. VI.

Commande par apprentissage itératif des systèmes non linéaires en présence des perturbations non répétitives

VI.1 Introduction

La plupart des classes des systèmes non linéaires que l'on peut trouver dans l'industrie, sont soumises à des perturbations externes répétitives et non répétitives. La présence de ces perturbations peut détruire le comportement du système de commande [MER 05]. A cet effet, afin d'améliorer les performances de la commande il est très important de développer des nouvelles lois de commande basées sur l'utilisation des estimateurs de perturbations.

Dans la littérature, nous avons constaté que la majorité des travaux sur la commande des systèmes non linéaires répétitifs considèrent que les perturbations sont variables en fonction du temps, mais, elles sont invariables vis-à-vis au domaine des itérations [WEI 14] et [WEI 17]. Il existe d'autres recherches qui ont supposé que les perturbations soient variables en itérations, mais elles sont supposées bornées par une constante connue [BOU 10], [DIN 14] et [HE 18]. Dans [SUN 18], un estimateur de perturbations a été présenté pour une classe des systèmes linéaire, dans lequel les dérivées des perturbations sont considérées nulles.

Dans ce chapitre, nous proposons une loi de commande par apprentissage itératif de Type -P- à base des réseaux de neurones à fonction de base radiales (RBF) [BEN 192] afin de résoudre le problème de poursuite d'une classe des systèmes non linéaires multivariables soumise à des perturbations externes inconnues variables en fonction du temps et des itérations en même temps. Dans laquelle, un estimateur des perturbations est proposé pour surmonter cet effet. La stabilité asymptotique du système en boucle fermée est démontrée par la méthode de Lyapunov.

A la fin de ce chapitre, les résultats de simulation sont présentés pour montrer l'efficacité de la méthode proposée.

VI.2 Position du problème

Nous considérons le système non linéaire suivant :

$$\dot{x}_k = f(x_k(t), t) + B(x_k(t), t)u_k(t) + d_k(t) \quad (\text{VII.1})$$

Où :

$x_k \in R^n, u_k \in R^n$ représentent respectivement, le vecteur d'état et la commande.

$f(*) = [f_1(*), \dots, f_n(*)] \in R^n$ est une vecteur des fonctions non linéaire inconnues, $B \in R^{n \times n}$ est une matrice de commande et $d_k(t)$ est le vecteur des perturbations externes. $f(x_k(t))$ est bornée par une fonction inconnue, donc

$$\|(\dot{x}_d - f(x_k))\| \leq \delta(x_k) \quad (\text{VII.2})$$

En utilisant la technique d'approximation des réseaux de neurones RBF, la fonction inconnue $\delta(x_k)$ peut être estimée par

$$\hat{\delta}(x_k) = \hat{\theta}_k^T \varphi(x_k) + \varepsilon(x_k) \quad (\text{VII.3})$$

Où ε est l'erreur d'approximation, elle est supposée très petite et bornée. $\varphi(x_k, t)$ est le vecteur des fonctions à base radiales, dans ce chapitre, la forme de la fonction radiale est choisie comme suit

$$\varphi_i(x_k) = \exp\left(-\frac{\|x_k - C_i\|^2}{\sigma_i^2}\right) \quad (\text{VII.4})$$

où $C_i \in R^l$ et σ_i sont respectivement, le centre et le rayon d'influence de neurone.

$\hat{\theta}_k(t) \in R^n$ est le vecteur d'estimation de $\theta_k(t)$ à travers les itérations. La loi d'adaptation proposée est comme suit :

$$\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + \beta_1 e_k^T(t) \text{sgn}(e_k(t)) \varphi(x_k, t) \quad (\text{VII.5})$$

avec $\hat{\theta}_{-1}(t) = 0$ et β_1 est une constante positive.

L'erreur d'estimation est donnée par

$$\tilde{\theta}_k(t) = \theta_k - \hat{\theta}_k(t) \quad (\text{VII.6})$$

L'erreur de la poursuite à la $k^{\text{ième}}$ itération est définie par :

$$e_k(t) = x_d(t) - x_k(t) \quad (\text{VII.7})$$

Par dérivation, on obtient

$$\dot{e}_k(t) = \dot{x}_d(t) - \dot{x}_k(t) \quad (\text{VII.8})$$

En remplaçant (VII.1) dans (VII.8), il vient

$$\dot{e}_k(t) = \dot{x}_d(t) - f(x_k(t), t) - B(x_k(t), t)u_k(t) - d_k(t) \quad (\text{VII.9})$$

Objectif : L'objectif principal est de développer une loi de commande par apprentissage itératif combinée avec les réseaux de neurones $u_k(t)$ afin de forcer la sortie réelle $x_k(t)$, d'un système non linéaire multivariable qui est soumis aux perturbations externes non répétitives, à suivre une trajectoire désirée $x_d(t)$ sur un intervalle de temps fixe $t \in [0, T]$, lorsque k tend vers l'infini.

L'hypothèse suivante est nécessaire.

Hypothèse VII.1 : A chaque itération, les conditions initiales sont les mêmes, i.e., $x_k(0) = x_d(0)$.

VI.3 Synthèse de la loi de commande

Afin de commander le système (VII.1), nous proposons la loi de CAI de Type -P- suivante :

$$u_k(t) = B^{-1}(x_k, t) [K_p e_k(t) + \hat{\delta}_k(x) \text{sgn}(e_k(t)) - \hat{d}_k(t)] \quad (\text{VII.10})$$

Où K_p est le gain proportionnel, $\hat{\delta}_k(x)$ est défini dans (VII.3) est $\hat{d}_k(t)$ l'estimateur des perturbations. Il est donné par

$$\hat{d}_k(t) = \hat{d}_{k-1}(t) - \beta_2 e_k^T(t) \quad (\text{VII.11})$$

avec β_2 est un gain positif.

Théorème [BEN 192] : Etant donnée le système non linéaire MIMO (VII.1) satisfaisant l'hypothèses VII.1. En appliquant la loi de commande (VII.10) avec l'estimateur des perturbations (VII.11), alors

- i) $e_k(t) \in L_{\infty e}[0, T], \hat{\theta}_k(x) \in L_{\infty e}[0, T]$ et $\hat{d}_k(x) \in L_{\infty e}[0, T]$
 ii) $\lim_{k \rightarrow \infty} e_k(t) = 0, \forall t \in [0, T]$.

Preuve

La preuve est basée sur la théorie de Lyapunov comme dans les sections précédentes.

Les notations suivantes sont utilisées $B^{-1}(x_k, t) = B_k^{-1}, d_k(t) = d_k, \theta_k(t) = \theta_k, \delta(x_k) = \delta_k, f(x_k) = f_k, \varphi(x_k) = \varphi_k$ et $e_k(t) = e_k$

Partie 1 : La fonction de Lyapunov est choisie comme suit

$$V_k(t) = \frac{1}{2} e_k^T e_k + \frac{1}{2\beta_1} \int_0^t \tilde{\theta}_k^T \tilde{\theta}_k d\tau + \frac{1}{2\beta_2} \int_0^t \tilde{d}_k^T \tilde{d}_k d\tau \quad (\text{VII.12})$$

On note

$$\Delta V_k = V_k - V_{k-1} \quad (\text{VII.13})$$

De (VII.12), on obtient

$$\begin{aligned} \Delta V_k(t) &= \frac{1}{2} e_k^T e_k - \frac{1}{2} e_{k-1}^T e_{k-1} + \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ &\quad + \frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \end{aligned} \quad (\text{VII.14})$$

Cela implique que

$$\begin{aligned} \Delta V_k(t) &\leq \frac{1}{2} e_k^T e_k + \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ &\quad + \frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \end{aligned} \quad (\text{VII.15})$$

En utilisant l'hypothèse 1, (VII.15) peut être réécrite sous forme

$$\begin{aligned} \Delta V_k(t) &\leq \int_0^t e_k^T \dot{e}_k d\tau + \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ &\quad + \frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \end{aligned} \quad (\text{VII.16})$$

En remplaçant (VII.9) et (VII.10) dans (VII.15), il vient

$$\begin{aligned} \Delta V_k(t) &\leq \int_0^t (e_k^T (\dot{x}_d - f_k) - e_k^T K_p e_k - \hat{\delta}_k e_k^T \text{sgn}(e_k) + e_k^T \hat{d}_k - e_k^T d_k) d\tau \\ &\quad + \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau + \frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \end{aligned} \quad (\text{VII.17})$$

En utilisant (VII.2) et sachant que $\tilde{\delta}_k = \delta_k - \hat{\delta}_k$ et $\tilde{d}_k = d_k - \hat{d}_k$, on obtient

$$\begin{aligned} \Delta V_k(t) &\leq \int_0^t (\tilde{\delta}_k e_k^T \operatorname{sgn}(e_k) - e_k^T K_p e_k - \tilde{d}_k e_k^T) d\tau \\ &+ \frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau + \frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \end{aligned} \quad (\text{VII.18})$$

De (VII.6), on a

$$\tilde{\theta}_k = \tilde{\theta}_{k-1} - \beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k \quad (\text{VII.19})$$

De (VII.11), il vient

$$\tilde{d}_k = \tilde{d}_{k-1} + \beta_2 e_k \quad (\text{VII.20})$$

Maintenant, nous examinons le deuxième terme de (VII.18). En utilisant (VII.19), on trouve

$$\begin{aligned} &\frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ &= \frac{1}{2\beta_1} \int_0^t \left((\tilde{\theta}_{k-1} - \beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k)^T \tilde{\theta}_k \right. \\ &\quad \left. - (\tilde{\theta}_k + \beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k)^T \tilde{\theta}_{k-1} \right) d\tau \end{aligned} \quad (\text{VII.21})$$

Sachant que, $\tilde{\theta}_{k-1}^T \tilde{\theta}_k = \tilde{\theta}_k^T \tilde{\theta}_{k-1}$, alors

$$\begin{aligned} &\frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \\ &= \frac{1}{2\beta_1} \int_0^t \left(-(\beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k)^T \tilde{\theta}_k \right. \\ &\quad \left. - (\beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k)^T \tilde{\theta}_{k-1} \right) d\tau \end{aligned} \quad (\text{VII.22})$$

En remplaçant (VII.19) dans (VII.22), on obtient

$$\frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq \frac{1}{2\beta_1} \int_0^t (-2\beta_1 e_k^T \operatorname{sgn}(e_k) \varphi_k)^T \tilde{\theta}_k d\tau \quad (\text{VII.23})$$

Comme $\varphi_k^T \tilde{\theta}_k = \tilde{\theta}_k^T \varphi_k = \tilde{\delta}_k$, donc

$$\frac{1}{2\beta_1} \int_0^t (\tilde{\theta}_k^T \tilde{\theta}_k - \tilde{\theta}_{k-1}^T \tilde{\theta}_{k-1}) d\tau \leq - \int_0^t \tilde{\delta}_k e_k^T \operatorname{sgn}(e_k) d\tau \quad (\text{VII.24})$$

En utilisant (VII.20), le dernier terme de (VII.18) devient

$$\begin{aligned} &\frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \\ &= \frac{1}{2\beta_2} \int_0^t \left((\tilde{d}_{k-1} + \beta_2 e_k)^T \tilde{d}_k - (\tilde{d}_k - \beta_2 e_k)^T \tilde{d}_{k-1} \right) d\tau \end{aligned} \quad (\text{VII.25})$$

Sachant que, $\tilde{d}_{k-1}^T \tilde{d}_k = \tilde{d}_k^T \tilde{d}_{k-1}$, alors

$$\frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau = \frac{1}{2\gamma} \int_0^t ((\beta_2 e_k)^T \tilde{d}_k + (\beta_2 e_k)^T \tilde{d}_{k-1}) d\tau \quad (\text{VII.26})$$

En remplaçant (VII.20) dans (VII.26), on trouve

$$\frac{1}{2\beta_2} \int_0^t (\tilde{d}_k^T \tilde{d}_k - \tilde{d}_{k-1}^T \tilde{d}_{k-1}) d\tau \leq \int_0^t \tilde{d}_k e_k^T d\tau \quad (\text{VII.27})$$

Maintenant, en substituant (VII.24) et (VII.27) dans (VII.18), Il vient

$$\Delta V_k \leq \int_0^t -e_k^T K_p e_k d\tau \quad (\text{VII.28})$$

En utilisant le théorème de Rayleigh-Ritz, (VII.28) devient

$$\Delta V_k \leq \int_0^t -\lambda_{\min}(K_p) \|e_k\|^2 d\tau \quad (\text{VII.29})$$

Il est clair que

$$\Delta V_k \leq 0 \quad (\text{VII.30})$$

Donc V_k est décroissante.

Partie 2 : Maintenant, nous démontrons que $V_0(t)$ est bornée $\forall t \in [0, T]$

De (VII.11), avec $k=0$, il vient

$$V_0(t) = \frac{1}{2} e_0^T e_0 + \frac{1}{2\beta_1} \int_0^t \tilde{\theta}_0^T \tilde{\theta}_0 d\tau + \frac{1}{2\beta_2} \int_0^t \tilde{d}_0^T \tilde{d}_0 d\tau \quad (\text{VII.31})$$

En dérivant les deux côtés de (VII.31), on obtient

$$\dot{V}_0(t) = e_0^T \dot{e}_0 + \frac{1}{2\beta_1} \tilde{\theta}_0^T \dot{\tilde{\theta}}_0 + \frac{1}{2\beta_2} \tilde{d}_0^T \dot{\tilde{d}}_0 \quad (\text{VII.32})$$

En remplaçant (VII.9) et (VII.10) dans (VII.27), on trouve

$$\begin{aligned} \dot{V}_0(t) = & e_0^T (\dot{x}_d - f_0) - e_0^T K_p e_0 - \hat{\delta}_0 e_0^T \text{sgn}(e_0) + e_0^T \hat{d}_0 - e_0^T d_0 + \frac{1}{2\beta_1} \tilde{\theta}_0^T \dot{\tilde{\theta}}_0 \\ & + \frac{1}{2\beta_2} \tilde{d}_0^T \dot{\tilde{d}}_0 \end{aligned} \quad (\text{VII.33})$$

En utilisant (VII.2), (VII.28) devient

$$\dot{V}_0(t) \leq \tilde{\delta}_0 e_0^T \text{sgn}(e_0) - e_0^T K_p e_0 - e_0^T \tilde{d}_0 + \frac{1}{2\beta_1} \tilde{\theta}_0^T \dot{\tilde{\theta}}_0 + \frac{1}{2\beta_2} \tilde{d}_0^T \dot{\tilde{d}}_0 \quad (\text{VII.34})$$

De (VII.19) avec $k=0$, on a

$$\tilde{\theta}_0 = -\beta_1 e_0^T \text{sgn}(e_0) \varphi_0 \quad (\text{VII.35})$$

De (VII.20) avec $k=0$, il vient

$$\tilde{d}_0 = \beta_2 e_0 \quad (\text{VII.36})$$

Il est clair que $\frac{1}{2\beta_1} \tilde{\theta}_0^T \dot{\tilde{\theta}}_0 \leq \frac{1}{\beta_1} \tilde{\theta}_0^T \dot{\tilde{\theta}}_0$ et $\frac{1}{2\beta_2} \tilde{d}_0^T \dot{\tilde{d}}_0 \leq \frac{1}{\beta_2} \tilde{d}_0^T \dot{\tilde{d}}_0$

Sachant que $\tilde{\delta}_0 = \tilde{\theta}_0^T \varphi_0 = \varphi_0^T \tilde{\theta}_0$, alors

$$\dot{V}_0(t) = -e_0^T K_p e_0 \quad (\text{VII.37})$$

En utilisant le théorème de Rayleigh-Ritz, (VII.37) devient

$$\dot{V}_0(t) \leq -\lambda_{\min}(K_p) \|e_0\|^2 \quad (\text{VII.38})$$

D'où

$$\dot{V}_0(t) \leq 0 \quad (\text{VII.39})$$

Il est clair que \dot{V}_0 est bornée, ce qui implique que V_0 est bornée.

Partie 3 : La fonction $V_k(t)$ peut être réécrite comme suit

$$V_k(t) = V_0(t) + \sum_{j=1}^k \Delta V_k(t) \quad (\text{VII.40})$$

En utilisant (VII.29), on obtient

$$V_k(t) \leq V_0(t) - \sum_{j=1}^k \int_0^t -\lambda_{\min}(K_p) \|e_k\|^2 d\tau \quad (\text{VII.41})$$

Ce qui nous donne

$$\sum_{j=1}^k -\lambda_{\min}(K_p) \int_0^t \|e_k\|^2 d\tau \leq V_0(t) - V_k(t) \leq V_0(t) \quad (\text{VII.42})$$

Comme $V_0(t)$ est bornée, alors

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \forall t \in [0, T] \quad (\text{VII.43})$$

VI.4 Résultats de Simulation

Nous considérons le système non linéaire suivant [LI 16] :

$$f(x_k) = \begin{bmatrix} x_{1k} + \sin(x_{2k}) \\ 2 + \sin(x_{1k}) + x_{2k} \end{bmatrix} \quad (\text{VII.44})$$

$$B(x_k) = \begin{bmatrix} 2 + \sin(x_{1k}) + 0.1 \sin(2\pi t) & 0 \\ 0 & 1 + \sin(x_{2k}) + 0.1 \sin(4\pi t) \end{bmatrix} \quad (\text{VII.45})$$

Le vecteur des perturbations est choisi comme suit

$$d_k(t) = \begin{bmatrix} \text{rand}(k) \sin(t) \\ \text{rand}(k) \sin(t) \end{bmatrix} \quad (\text{VII.46})$$

Le vecteur des trajectoires désirées est choisi comme suit

$$\begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} = \begin{bmatrix} \sin(2\pi t) \\ \cos(2\pi t) \end{bmatrix} \quad (\text{VII.47})$$

Les paramètres de simulation sont donnés par : $K_p = 12$, $\beta_1 = 4$ et $\beta_2 = 10$ et les paramètres de RBF sont choisis comme suit : $C_1 = 0.2$, $C_2 = 0.7$, $C_3 = -0.2$, $C_4 = -0.7$ et $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 0.15$.

En appliquant la loi de commande par apprentissage itératif proposée (VII.10) et l'estimateur des perturbations (VII.11), les résultats de simulations sont présentés dans les Figures VII.1-VII.6. Les trajectoires réelles et désirées de la 1^{ère}, la 2^{ème} et la 10^{ème} itérations pour la première et la deuxième sortie sont montrées dans les figures VII.1 et VII.2, respectivement. Nous pouvons observer que les trajectoires réelles suivent exactement les trajectoires désirées à travers les itérations, pour les deux sorties. Les Figures VII.3 et VII.4 présentent les erreurs de poursuite de la 1^{ère}, la 2^{ème} et la 10^{ème} itérations pour la première et la deuxième sortie, respectivement. Il est clair que, les erreurs diminuent à travers les itérations. La norme-Sup des erreurs pour la première et la deuxième sortie sont montrées dans les Figures VII.5 et VII.6, respectivement. Nous constatons qu'après la 10^{ème} itérations, les valeurs de la norme-Sup de la première et la deuxième sortie sont inférieures à 0,0065 et 0,0011, respectivement. Ça veut dire qu'après 10 itérations, la première erreur est réduite plus que 98 fois alors que la deuxième est réduite plus que 378 fois. Par conséquent, nous pouvons conclure que la commande proposée améliore les performances du système de commande à travers les itérations, ce qui prouve l'efficacité de cette technique de commande.

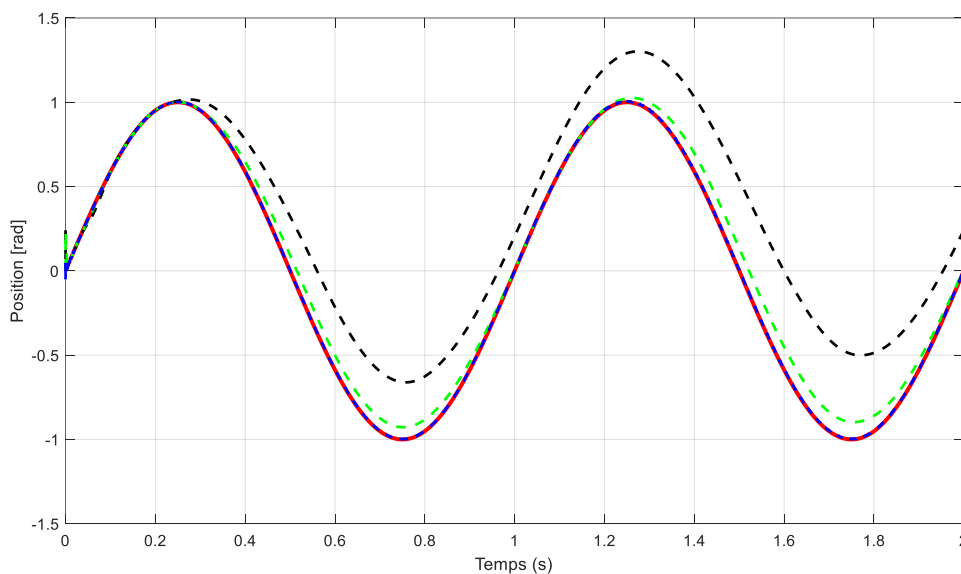


Figure VI.1 Trajectoires réelles et désirée à travers les itérations pour la première sortie.

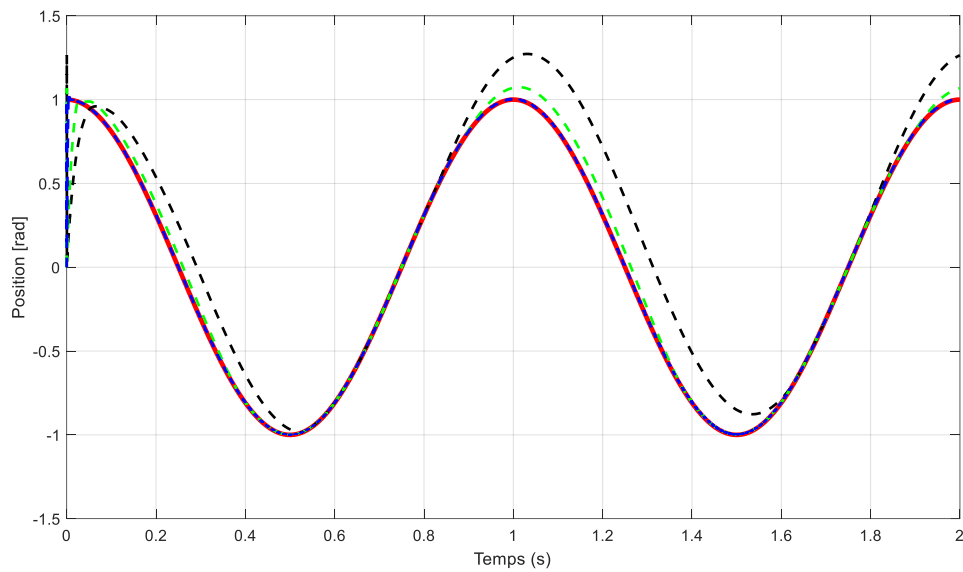


Figure VI.2 Trajectoires réelles et désirée à travers les itérations pour la deuxième sortie.

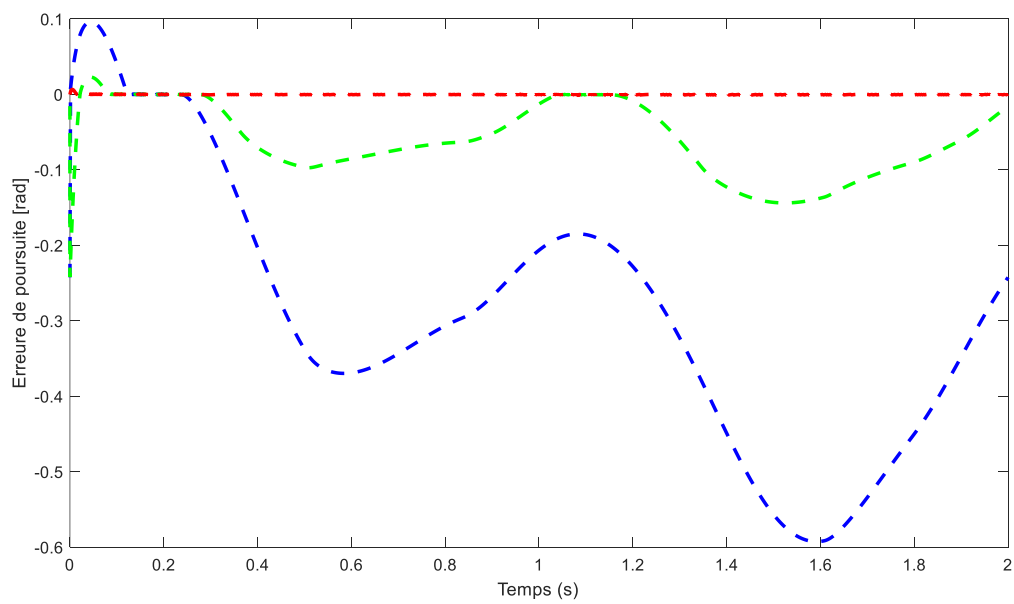


Figure VI.3 Erreurs de poursuite à travers les itérations pour la première sortie

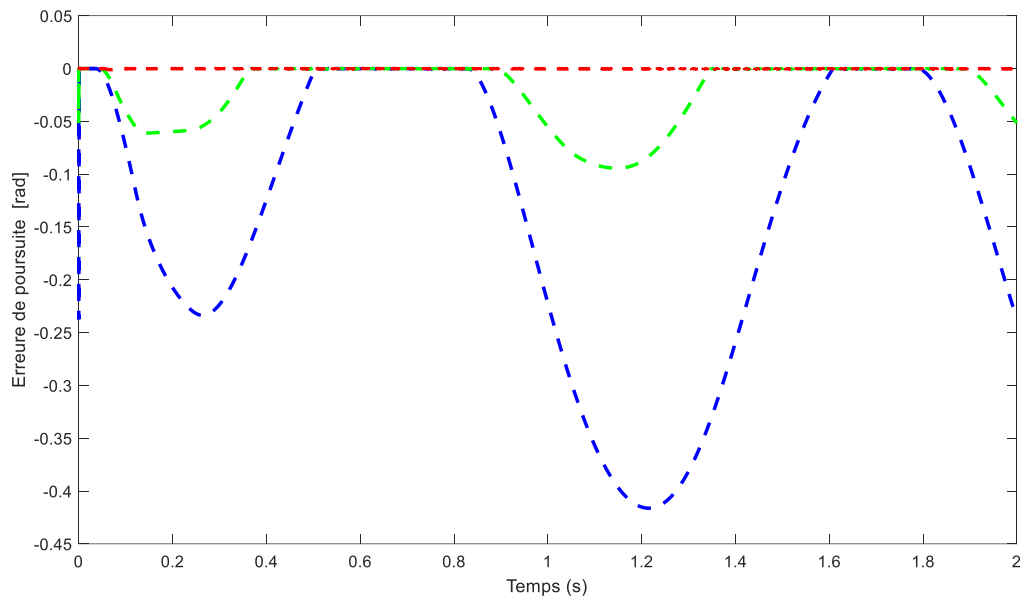


Figure VI.4 Erreurs de poursuite à travers les itérations pour la deuxième sortie

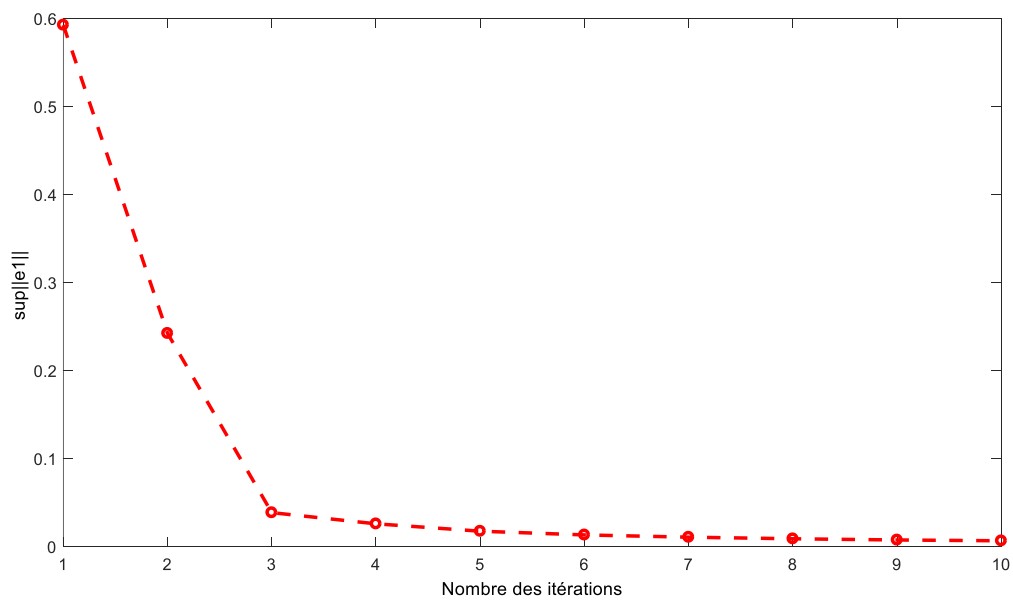


Figure VI.5 Norme Sup de l'erreur de la première sortie au cours des itérations.

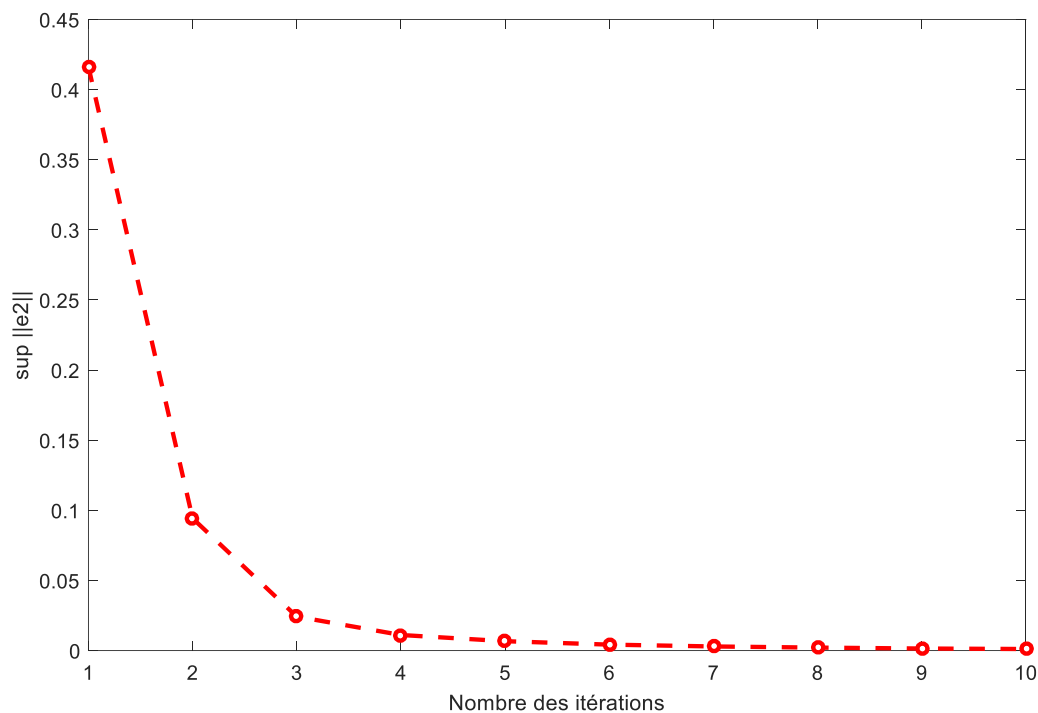


Figure VI.6 Norme Sup de l'erreur de la deuxième sortie au cours des itérations

VI.5 Conclusion

Dans ce chapitre, nous avons présenté une loi de commande par apprentissage itératif de type - P- à base des réseaux de neurones à fonctions de base radiales (RBF) pour une classe des systèmes non linéaires répétitif qui sont soumis aux perturbations externes inconnues. De plus, ces perturbations sont supposées à la fois variables en fonction du temps et en itérations. Pour ce faire, nous avons ajouté un estimateur de perturbations avec le schéma de commande itérative. En utilisant la méthode de Lyapunov, nous avons démontré la stabilité du système en boucle fermée. A la fin de ce chapitre, des simulations sont effectués sur un système non linéaire pour prouver l'efficacité de la méthode proposée.

Conclusion générale

Ce travail s'intéresse aux développements des lois de commande par apprentissage itératif des systèmes non linéaires multivariables. La classe des systèmes considérés dans ce manuscrit sont des systèmes qui fonctionnent d'une manière répétitive dans un intervalle de temps fixe. Ces systèmes sont largement utilisés dans l'industrie, l'exemple le plus simple sont les robots manipulateurs qui sont programmés pour exécuter des tâches répétitives. En effet, lorsqu'un système exécute une tâche répétitive, alors, en utilisant un régulateur conventionnel, la même erreur se répète d'un cycle à l'autre. L'idée de base de la commande par apprentissage itératif est de trouver un mécanisme itératif adéquat permettant d'apprendre les erreurs des cycles précédents et exécuter progressivement mieux le nouveau cycle. Par conséquent, le système se corrige au cours des itérations et les performances s'améliorent.

Tout d'abord, avant d'entamer les contributions, nous avons étudié la commande par apprentissage itératif, dans sa forme générale en donnant ses différents types. Puis, nous avons présenté les deux méthodes utilisées pour l'analyse de la stabilité du système en boucle fermée dans cette thèse. Ces méthodes sont la méthode de la norme- λ et la méthode de Lyapunov.

A travers notre recherche, nous avons constaté que, la commande par apprentissage itératif comme la plupart des commandes des systèmes non linéaires, rencontre des problèmes et des contraintes lors de l'analyse de stabilité et l'implémentation pratique. Parmi ces contraintes, on trouve la nécessité, pour les schémas classiques de la CAI, de satisfaire la condition de continuité globale de Lipschitz. Dans ce sens, nous avons surmonté ce problème en proposant un schéma de CAI sans utiliser cette hypothèse. Un autre problème rencontré est la nécessité de connaître le signe de la matrice de commande. Dans cette thèse, on trouve la synthèse d'une loi de CAI sans supposer la connaissance de ce signe. De plus, dans la pratique, l'existence des non-linéarités de type zone-morte et saturation sur les entrées des systèmes à commander peuvent déstabiliser et détériorer les performances des systèmes. Dans le but de donner des solutions à ces problèmes, nous avons développé deux lois de commande CAI en tenant compte ces deux contraintes. La première est appliquée sur un robot manipulateur avec la contrainte de la zone-morte à l'entrée, alors que la deuxième est appliquée sur un système non linéaire avec la contrainte de saturation en entrée. Il y a, également, d'autres contributions dans cette thèse,

telles que, la proposition des schémas de commande à gain variable et la synthèse de loi de commande avec un estimateur des perturbations. En fait, il est connu que l'utilisation de l'action dérivée dans les applications pratiques peut générer des bruits, et à cause de la nature itérative du système, ces bruits s'accumulent à travers les itérations ce qui pose un problème. Afin de surmonter ce problème, nous avons proposé des commandes de type P, sans utiliser l'action dérivée. Pour toutes les commandes proposées dans cette thèse, la stabilité de système en boucle fermée est démontrée en utilisant une des deux méthodes, la méthode de la norme- λ ou la méthode de Lyapunov.

Le manuscrit présenté est divisé en sept chapitres. Dans le premier chapitre, nous avons donné une introduction sur les motivations et objectifs de la thèse, ainsi qu'une brève histoire sur la commande par apprentissage itératif.

Le deuxième chapitre a présenté le principe de base de la commande par apprentissage itératif, les différentes classifications et les méthodes utilisées pour l'analyse de la stabilité. Dans le troisième chapitre, nous avons abordé la commande par apprentissage itératif à base des réseaux de neurones à fonctions de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des systèmes non linéaires multivariables répétitifs sans la connaissance du signe de la matrice de commande.

Le quatrième chapitre présente la synthèse de deux approches de commandes par apprentissage itératif. Dans le premier schéma, nous avons présenté une loi de commande adaptative par apprentissage itératif de type -P- à base des réseaux de neurones à fonction de base radiales (RBF) pour résoudre le problème de poursuite des trajectoires des robot manipulateurs possédants des non-linéarités de type zone-morte à leurs entrées et soumis à des perturbations externes non répétitives. Dans la deuxième approche, une loi de commande par apprentissage itératif de Type -P- d'ordre élevée est proposée pour traiter le problème de poursuite des systèmes non linéaires multivariables MIMO en présence des saturations à leurs entrées.

Dans le cinquième chapitre, une loi de commande par apprentissage itératif à gains variables est synthétisée. Le choix des gains est fait par la technique de l'optimisation PSO.

Le sixième chapitre est consacré à la synthèse d'un schéma de commande par apprentissage itératif avec un estimateur de perturbations est proposé pour une classe des systèmes non linéaires soumis à des perturbations externes inconnues.

A l'issue de ce travail, nous proposons les perspectives suivantes :

- Développement des lois de commande par apprentissage itératif pour les systèmes stochastiques.
- Développement des lois de commande par apprentissage itératif d'ordre fractionnaire.
- Développement des lois de commande par apprentissage itératif pour des systèmes d'ordre fractionnaire.
- Développer une loi de commande par apprentissage itératif à itération fini.

Bibliographie

- [AHN 07] Ahn, H. S., Chen, Y., & Moore, K. L. «Iterative learning control: Brief survey and categorization.» *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37, n° 6 (2007): 1099-1121.
- [ALA 08] Alavandar, S., & Nigam, M. J. «Fuzzy PD+ I control of a six DOF robot manipulator.» *Industrial Robot: An International Journal*, 2008: 35(2), 125-132.
- [ARI 84] Arimoto, S., Kawamura S. and Miyazaki F. «Bettering operation of robots by learning.» *Journal of Robotic Systems*, 1984: 1(2) :123-140.
- [BEN 19] Bensidhoum, T., Bouakrif, F. & Zasadzinski, M. «Iterative learning fuzzy control with optimal gains for a class of nonlinear systems. , 2019. p.» *In : 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE*, 2019: p.210-215.
- [BEN 191] Bensidhoum, T., Bouakrif, F. & Zasadzinski, M. «Iterative learning radial basis function neural networks control for unknown multi input multi output nonlinear systems with unknown control direction.» *Transactions of the Institute of Measurement and Control*, 2019: 41(12),3452-3467.
- [BEN 192] Bensidhoum, T., Bouakrif, F., & Zasadzinski, M. «Iterative learning neural networks control with disturbances estimation for nonlinear systems.» *In 2019 4th World Conference on Complex Systems (WCCS)*, 2019: (pp. 1-5). IEEE.
- [BEN 20] Bensidhoum, T., & Bouakrif, F. «Adaptive P-type iterative learning radial basis function control for robot manipulators with unknown varying disturbances and unknown input dead zone.» *International Journal of Robust and Nonlinear Control*, 2020: 30(10),4075-4094.
- [BER 93] Berghuis, H., & Nijmeijer, H. «A passivity approach to controller-observer design for robots.» *IEEE Transactions on robotics and automation*, 1993: 9(6), 740-754.
- [BIE 89] Bien, Z., & Huh, K. M. «Higher-order iterative learning control algorithm.» *In IEE Proceedings D (Control Theory and Applications). IET Digital Library*, 1989: 136(3),105-112.
- [BOU 08] Bouakrif, Farah. «Commande par apprentissage itératif des robots manipulateurs.» *Thèse de doctorat ENP, Alger*, 2008.
- [BOU 10] Bouakrif, F. «Reference model iterative learning control for linear systems.» *In 18th Mediterranean Conference on Control and Automation, MED'10*, 2010, June: (pp. 1637-1642). IEEE.

-
- [BOU 101] Bouakrif, F. «Reference model iterative learning control for nonlinear systems with repeatable and non-repeatable uncertainties.» *The International Journal of Advanced Manufacturing Technology*, 2010: 51(9-12),1159-1169.
- [BOU 11] Bouakrif, F. «D-type iterative learning control without resetting condition for robot manipulators.» *Robotica*, 2011: 29(7), 975.
- [BOU 111] Bouakrif, F. «Iterative learning control for strictly unknown nonlinear systems subject to external disturbances.» *International Journal of Control, Automation and Systems*, 2011: 9(4),642-648.
- [BOU 13] Bouakrif, F., Boukhetala, D., & Boudjema, F. «Velocity observer-based iterative learning control for robot manipulators.» *International Journal of Systems Science*, 2013: 44(2),214-222.
- [BOU 14] Bouakrif, F. «Comparison between three types of iterative learning control for nonlinear systems: On-line, off-line and on-line taking into account the previous error. In 2014 International Conference on Electrical Sciences and Technologi.» *In 2014 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM)*, 2014: 1-6. IEEE.
- [BOU 16] Bouakrif, F. «Iterative learning control for affine and non-affine.» *In: Vaidyanathan S and Volos C (eds) Advances and Applications in Nonlinear Control Systems Switzerland: Springer International Publishing*, 635 (2016): 555–574.
- [BOU 161] Bouakrif, F., & Zasadzinski, M. «Trajectory tracking control for perturbed robot manipulators using iterative learning method.» *The International Journal of Advanced Manufacturing Technology*, , 2016: 87(5-8), 2013-2022.
- [BOU 18] Bouakrif, F., & Zasadzinski, M. «High order iterative learning control to solve the trajectory tracking problem for robot manipulators using Lyapunov theory. .» *Transactions of the Institute of Measurement and Control*, 2018: 40(15), 4105-4114.
- [BOU 20] Bouakrif, F., Bensidhoum, T., & Zasadzinski, M. «Trajectory tracking controller for nonlinear systems with disturbances using iterative learning algorithm without resetting condition.» *In : 2020 7th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE, Vol. 1., 2020, June.: pp. 517-522.*
- [BRI 06] Bristow, D. A., Tharayil, M., & Alleyne, A. G. «A survey of iterative learning control.» *IEEE control systems magazine*, 2006: 26(3),96-114.
- [CHE 06] Chen, Y., Moore, K. L., Yu, J., & Zhang, T. «Iterative learning control and repetitive control in hard disk drive industry-a tutorial.» *In Proceedings of the 45th IEEE Conference on Decision and Control*. 2006. 2338-2351.

-
- [CHE 11] Chen, W., Li, J., & Li, J. «Practical adaptive iterative learning control framework based on robust adaptive approach.» *Asian Journal of Control*, 2011: 13(1)85-93.
- [CHI 02] Chien, C. J., & Fu, L. C. «An iterative learning control of nonlinear systems using neural network design.» *Asian Journal of Control*, 2002: 4(1),21-29.
- [CHI 04] Chien, C. J., & Yao, C. Y. «An output-based adaptive iterative learning controller for high relative degree uncertain linear systems.» *Automatica*, 2004: 40(1),145-153.
- [CHI 08] Chien, C. J. «A combined adaptive law for fuzzy iterative learning control of nonlinear systems with varying control tasks.» *IEEE Transactions on Fuzzy Systems*, 2008: 16(1),40-51.
- [CHI 18] Chi, R., Huang, B., Hou, Z., & Jin, S. «Data-driven high-order terminal iterative learning control with a faster convergence speed.» *International Journal of Robust and Nonlinear Control*, 2018: 28(1),103-119.
- [DER 00] De Roover, D., & Bosgra, O. H. « Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system.» *international Journal of Control*, 2000: 73(10), 968-979.
- [DIN 14] Ding, J., & Yang, H. «Adaptive iterative learning control for a class of uncertain nonlinear systems with second-order sliding mode technique.» *Circuits, Systems, and Signal Processing*, 2014: 33(6),1783-1797.
- [GAO 18] Gao, S. Z., Wu, X. F., Luan, L. L., Wang, J. S., & Wang, G. C. «PSO optimal control of model-free adaptive control for PVC polymerization process .» *International Journal of Automation and Computing*, 2018: 15(4), 482-491.
- [GE 04] Ge, S. S., Hong, F., & Lee, T. H. (2004). Adaptive neural control of nonlinear time-delay system with unknown virtual control coefficients. *IEEE Transactions on Systems, Man, and Cybernetics-PartB: Cybernetics*, 34(1), 499–516.
- [GU 19] Gu, P., & Tian, S. «D-type iterative learning control for one-sided Lipschitz nonlinear systems.» *International Journal of Robust and Nonlinear Control*, 2019: 29(9), 2546-2560.
- [HAK 08] Hakvoort, W. B. J., Aarts, R. G., van Dijk, J., & Jonker, J. B. «Lifted system iterative learning control applied to an industrial robot.» *Control Engineering Practice*, 2008: 16(4), 377-391.
- [HAN 18] Hannan, M. A., Akhtar, M., Begum, R. A., Basri, H., Hussain, A., & Scavino, E. «Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm.» *Waste management*, 2018: 71, 31-41.

-
- [HE 18] He, W., Meng, T., He, X., & Ge, S. S. «Unified iterative learning control for flexible structures with input constraints.» *Automatica*, 2018: 96, 326-336.
- [HOU 11] Hou, Z., Wang, Y., Yin, C., & Tang, T. «Terminal iterative learning control based station stop control of a train.» *International Journal of Control*, 2011: 84(7), 1263-1274.
- [JIA 04] Jian.X. Xu, Y. Tan, and T.H. Lee,. «Iterative learning control design based on composite energy function with input saturation.» *Automatica*, 2004: 40(8), 1371-1377.
- [JI 16] Ji, H., Hou, Z., Fan, L., & Lewis, F. L. «Adaptive iterative learning reliable control for a class of non-linearly parameterised systems with unknown state delays and input saturation.» *IET Control Theory & Applications*, 2016: 10(17),2160-2174.
- [JIN 16] Jin, X. «Adaptive iterative learning control for high-order nonlinear multi-agent systems consensus tracking.» *Systems & Control Letters* 89 (2016): 16-23.
- [JIN 17] Jin, X. «Adaptive finite-time fault-tolerant tracking control for a class of MIMO nonlinear systems with output constraints.» *International Journal of Robust and Nonlinear Control*,, 2017: 27(5),722-741.
- [KAN 05] Kang, M. K., Lee, J. S., & Han, K. L. «Kinematic path-tracking of mobile robot using iterative learning control.» *Journal of Robotic Systems*, 2005: 22(2), 111-121.
- [KEL 05] Kelly R, Santibàñez V, Loría A. *Control for Robot Manipulators in Joint Space*. London, UK: Springer-Verlag, 2005.
- [KIM 96] Kim, D. I., & Kim, S. «An iterative learning control method with application for CNC machine tools.» *IEEE Transactions on Industry Applications*, 1996: 32(1), 66-72.
- [KIM 17] Kim, M., Kuc, T. Y., Kim, H., & Lee, J. S. «Adaptive iterative learning controller with input learning technique for a class of uncertain MIMO nonlinear systems.» *International Journal of Control, Automation and Systems*, 2017: 15(1),315-328.
- [LEA 06] Leang, K. K., & Devasia, S. «Leang, K. K., & Devasia, S. (2006). Design of hysteresis[-compensating iterative learning control for piezo-positioners: Application to atomic force microscopes.» *Mechatronics*, 2006: 16(3-4), 141-158.
- [LI 13] Li, J., & Li, J. «Adaptive iterative learning control for consensus of multi-agent systems.» *IET Control Theory & Applications*, 2013: 7(1),136-142.
- [LI 131] Li, X. D., Chow, T. W., & Cheng, L. L. «Adaptive iterative learning control of non-linear MIMO continuous systems with iteration-varying initial error and

-
- reference trajectory.» *International Journal of Systems Science*, 2013: 44(4), 786-794.
- [LI 16] Li, X., Huang, D., Chu, B., & Xu, J. X. «Robust iterative learning control for systems with norm-bounded uncertainties.» *International Journal of Robust and Nonlinear Control*, 2016: 26(4),697-718.
- [LI 17] Li, X., & Shen, D. «Two novel iterative learning control schemes for systems with randomly varying trial lengths.» *Systems & Control Letters* 107 (2017): 9-16.
- [MAD 08] Madady, A. «PID type iterative learning control with optimal gains. .» *International Journal of Control, Automation, and Systems*, 2008: 6(2), 194-203.
- [MCC 43] McCulloch, W. S., & Pitts, W. «A Logical Calculus of the Ideas Immanent in Nervous Activity.» *Bulletin of Mathematical Biophysics*, 1943: 5, 115-133.
- [MER 05] Merry, R., van de Molengraft, R., and Steinbuch, M.,. «The influence of disturbances in iterative learning control.» *In Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on*, 2005: 974-979.
- [MEZ 02] Mezghani, M., Roux, G., Cabassud, M., Le Lann, M. V., Dahhou, B., & Casamatta, G. «Application of iterative learning control to an exothermic semibatch chemical reactor.» *IEEE Transactions on Control Systems Technology*, 2002: 10(6), 822-834.
- [NOR 02] Norrlof, M. «An adaptive iterative learning control algorithm with experiments on an industrial robot.» *IEEE Transactions on robotics and automation*, 2002: 18(2), 245-251.
- [PAS 13] Paszke, W., Rogers, E., Gałkowski, K., & Cai, Z. «Robust finite frequency range iterative learning control design and experimental verification. 21(10), 1310-1320.» *Control Engineering Practice*,, 2013: 21(10),1310-1320.
- [QU 02] Qu, Z. «An iterative learning algorithm for boundary control of a stretched moving string.» *Automatica*, 2002: 38(5),821-827.
- [HOR 93] Horowitz, R. «Learning control of robot manipulators.» *Trans. ASME J. Dyn. Syst. Meas. Control*, 1993: 115(2B),402-411.
- [SEB 19] Sebastian, G. Feedback-based iterative learning control for constrained systems with application to robotic manipulators. Thèse de doctorat, université de Melbourne, Australie. 2019.
- [SEB 191] Sebastian, G., Tan, Y., & Oetomo, D. «Convergence analysis of feedback-based iterative learning control with input saturation.» *Automatica*, 2019: 101, 44-52.

-
- [SUN 18] Sun, J., and Li, S., «Disturbance observer based iterative learning control method for a class of systems subject to mismatched disturbances.» *Transactions of the Institute of Measurement and Control*, 2018: 39(11),1749-1760,.
- [TAN 08] Tan, Y., & Xu, J. X. «A note on iterative learning control for nonlinear systems with input uncertainties.» *IFAC Proceedings Volumes*, , 2008: 41(2), 12474-12479.
- [TAN 12] Tan, Y., Dai, H. H., Huang, D., & Xu, J. X. «Unified iterative learning control schemes for nonlinear dynamic systems with nonlinear input uncertainties.» *Automatica*, 2012: 48(12), 3173-3182.
- [TAY 04] Tayebi, A. «Adaptive iterative learning control for robot manipulators.» *Automatica*, 2004: 40(7), 1195-1203.
- [TAY 06] Tayebi, A, et S islem. «Adaptive iterative learning control for robot manipulators: Experimental results.» *Control Engineering Practice*, 2006: 14(7), 843-851.
- [TAY 08] Tayebi, A., Abdul, S., Zaremba, M. B., & Ye, Y. «Tayebi, A., Abdul, S., Zaremba, M. B., & Ye, Y. Robust iterative learning control design: application to a robot manipulator.» *IEEE/ASME Transactions on mechatronics*, 2008: 13(5), 608-613.
- [UCH 78] Uchiyama, M. «Formation of high-speed motion pattern of a mechanical arm by trial.» *Transactions of the Society of Instrument and Control Engineers*, 1978: 14(6),706-712.
- [WAN 09] Wang, Y., Gao, F., & Doyle III, F. J. «Survey on iterative learning control, repetitive control, and run-to-run control.» *Journal of Process Control* 19, n° 10 (2009): 1589-1600.
- [WAN 13] Wang, Y. C., Chien, C. J., & Chuang, C. N. «Adaptive iterative learning control of robotic systems using backstepping design.» *Transactions of the Canadian Society for Mechanical Engineering*, 2013: 37(3), 591-601.
- [WAN 18] Wang, C.,Zheng,M.,Wang,Z.,Peng,C.,&Tomizuka,M. «Robust iterative learning control for vibration suppression of industrial robot manipulators, 140(1).» *Journal of Dynamic Systems, Measurement, and Control*, 2018: 140(1).
- [WEI 11] Wei, J., & Hu, Y. «Open-closed-loop PD-type iterative learning control with variable gains.» *In International Conference on Computer Science, Environment, Ecoinformatics, and Education*. Berlin, Heidelberg.: Springer,, 2011. (pp. 592-597).
- [WEI 14] Wei, J., Hu, Y., & Sun, M. «Adaptive iterative learning control for a class of nonlinear time-varying systems with unknown delays and input dead-zone.» *IEEE/CAA Journal of Automatica Sinica*,, 2014: 1(3), 302-314.

-
- [WEI 17] Wei, J., Zhang, Y., Sun, M., & Geng, B. «Adaptive iterative learning control of a class of nonlinear time-delay systems with unknown backlash-like hysteresis input and control direction.» *ISA transactions* 70 (2017): 79-92.
- [XU 98] Xu, J. X., & Qu, Z. «Robust iterative learning control for a class of nonlinear systems.» *Automatica*, 1998: 34(8), 983-988.
- [YAN 01] Yan, X. G., Chen, I. M., & Lam, J. «D-type learning control for nonlinear time-varying systems with unknown initial states and inputs.» *Transactions of the Institute of Measurement and Control* 23, n° 2 (2001): 69-82.
- [YAN 15] Yanxin, Z., Anqi, W., Tingxu, Z., & Zhiqing, H. «PD iterative learning control based on neural network and genetic parameter optimization. .» *In The 27th Chinese Control and Decision Conference* , 2015: 5192-5195. IEEE.
- [YAN 19] Yang, N., & Li, J. «Distributed iterative learning coordination control for leader–follower uncertain non-linear multi-agent systems with input saturation.» *IET Control Theory & Applications*, 2019: 13(14), 2252-2260.
- [YOU 18] You, B. «Normalized learning rule for iterative learning control.» *International Journal of Control, Automation and Systems*, 2018: 16(3),1379-1389.
- [YUL 13] Yu, L., Zhang, M., & Fei, S. «Non-linear adaptive sliding mode switching control with average dwell-time.» *International Journal of Systems Science*2013, 2013: 44(3), 471-478.
- [YUL 17] Yu, L., Huang, J., & Fei, S. «Sliding mode switching control of manipulators based on disturbance observer.» *Circuits, Systems, and Signal Processing*, 2017: 36(6), 2574-2585.
- [YUL 18] Yu, L., Li, C., & Fei, S. «Any-wall touch control system with switching filter based on 3-D sensor .» *IEEE Sensors Journal*, 2018: 18(11), 4697-4703.
- [YUM 13] Yu, M., Ye, X., & Qi, D. «Robust adaptive repetitive learning control for a class of time-varying nonlinear systems with unknown control direction.» *Journal of Control Theory and Applications*, 2013: 11(3),336-342.
- [YUQ 16] Yu, Q., Hou, Z., & Chi, R. «Adaptive iterative learning control for nonlinear uncertain systems with both state and input constraints.» *Journal of the Franklin Institute*, 2016: 353(15), 3920-3943.
- [ZAD 65] Zadeh, L. «Information and control.» *Fuzzy Sets*, 1965: 8,338-353.
- [ZHA 18] Zhang, J., & Meng, D. «Iterative Learning Control for Continuous-Time Systems with Locally Lipschitz Nonlinearity and Input Saturation.» *In 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*, 2018: 908-913. IEEE.

- [ZHA 16] Zhang, R., Hou, Z., Ji, H., & Yin, C. «Adaptive iterative learning control for a class of non-linearly parameterised systems with input saturations.» *International Journal of Systems Science*, 2016: 47(5),1084-1094.
- [ZHA 07] Zhang, T. P., & Ge, S. S. «Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs.» *Automatica*, 2007: 43(6), 1021-1033.
- [ZOU 19] Zou, J., Chang, Q., Ou, X., Arinez, J., & Xiao, G. «Resilient adaptive control based on renewal particle swarm optimization to improve production system energy efficiency.» *Journal of Manufacturing Systems*, 2019: 50, 135-145.

ملخص

على مدى العقود الثلاثة الماضية, عرف قانون التحكم بالتعليم المرحلي قدرا كبيرا من الاهتمام. في الواقع, يؤدي استخدام قوانين التحكم التقليدية إلى الحصول على نفس الخطأ في كل دورة. ومن هنا تأتي الحاجة إلى إيجاد آلية قادرة على استخدام المعلومات من الدورات السابقة لتنفيذ الدورة الجديدة بشكل أفضل. وهذا ما يميز قانون التحكم بالتعليم المرحلي.

العمل المقدم في هذه المذكرة يتعلق بتطبيق قوانين جديدة للتحكم بالتعليم المرحلي على الأنظمة غير الخطية المتكررة. خلال بحثنا ودراستنا المعمقة, وجدنا أن هذا النوع من التحكم يعاني من العديد من المشاكل عند تحليل الاستقرار والتطبيق العملي. من بين هذه الصعوبات, الحاجة إلى تلبية شرط الاستمرارية العامة ليشيز, لهذا تم اقتراح قانون دون استخدام هذه الفرضية. هناك مشكلة أخرى تمت معالجتها وهي شرط المعرفة المسبقة لإشارة مصفوفة التحكم. في هذه الأطروحة, تم اقتراح قانون التحكم بالتعليم المرحلي دون معرفة هذه الإشارة.

بالإضافة إلى ذلك, من الناحية العملية, يمكن أن يؤدي وجود عوامل غير خطية مثل المنطقة الميتة والتشبع عند المدخلات إلى زعزعة استقرار وتدهور أداء الأنظمة. من أجل توفير حلول لهذه المشاكل, قمنا بتطوير قانونين للتحكم بالتعليم المرحلي مع مراعاة هذين المعيقين, تم تطبيق القانون الأول على روبوت مع قيد المنطقة الميتة للإدخال, بينما تم تطبيق القانون الثاني على نظام غير خطي مع قيد تشبع الإدخال. بالإضافة إلى ذلك, تم اقتراح مساهمات أخرى, مثل قانون التحكم في اختيار قيم المتغير وقانون التحكم مع تقدير الاضطرابات المتكررة وغير المتكررة.

الكلمات المفتاحية : قانون التحكم بالتعليم المرحلي, الأنظمة غير الخطية المتكررة, ذراع آلي, المنطقة الميتة, التشبع, الشبكة العصبية من نوع RBF, طريقة ليابونوف, مقياس λ , تقدير الاضطرابات.

Abstract

Over the past three decades, the control for repetitive systems has generated a great deal of interest. In industrial applications where the system involved executes the same operation many times over a fixed finite time interval, iterative learning control (ILC) is the best choice for such systems. The basic idea of this method is to use information from previous operations to improve system performance from iteration to iteration.

This thesis focuses on the design and development of new ILC schemes for repetitive nonlinear systems. Indeed, during the stability analysis and the practical implementation of this type of control, we found that it suffers from many problems. The need to satisfy the global Lipschitz continuity condition is one of these difficulties. In this sense, we have synthesized an ILC scheme without using this hypothesis. Another problem encountered is the necessity to know the sign of the control directions. In this thesis, a new ILC without any prior knowledge of the control directions has been proposed. In addition, in practice, the existence of non-linearities such as dead zone and saturation at the inputs can destabilize and deteriorate the performance of the system. To provide solutions to these problems, we have developed two ILC control laws considering these two constraints. The first scheme is applied to robot manipulators with the input dead zone constraint, while the second is applied to nonlinear systems with the input saturation constraint. Furthermore, other contributions have been proposed, such as, ILC with variable gains and ILC scheme for nonlinear systems with external repetitive and non-repetitive disturbances.

Keywords: iterative learning control, adaptive, repetitive nonlinear systems, robot manipulator, dead zone constraint, saturation constraint, RBF neural networks, Lyapunov theory, λ -norm, non-repetitive disturbances.

Résumé

Au cours des trois dernières décennies, la commande des systèmes répétitifs a suscité énormément d'intérêt. En fait, l'utilisation d'un régulateur classique pour de tels systèmes fournira la même erreur à chaque cycle. D'où, la nécessité de trouver un mécanisme capable d'utiliser les informations des cycles précédents pour exécuter progressivement mieux le nouveau cycle. Ceci caractérise la commande par apprentissage itératif (CAI).

Cette thèse porte sur la conception et le développement de nouveaux schémas de CAI des systèmes non linéaires répétitifs. En effet, lors de l'analyse de stabilité et l'implémentation pratique de ce type de commande, nous avons constaté qu'elle souffre de nombreux problèmes. Parmi ces difficultés, la nécessité de satisfaire la condition de continuité globale de Lypschitz. Dans ce sens, nous avons pu synthétiser un schéma de CAI sans utiliser cette hypothèse. Un autre problème rencontré est la connaissance du signe de la matrice de commande. Dans cette thèse, une loi de CAI sans la connaissance de ce signe a été proposée. De plus, dans la pratique, l'existence des non-linéarités de type zone-morte et saturation en entrées peuvent déstabiliser et détériorer les performances des systèmes. Dans le but de donner des solutions à ces problèmes, nous avons développé deux lois de commande CAI en tenant compte ces deux contraintes. La première est appliquée sur un robot manipulateur avec la contrainte de la zone-morte en entrée, alors que la deuxième est appliquée sur un système non linéaire avec la contrainte de saturation en entrée. En outre, d'autres contributions ont été proposées, telles que, loi de commande à gain variable et loi de CAI avec estimateur de perturbations répétitives et non répétitives.

Mot clés : Commande par apprentissage itératif, systèmes non linéaires répétitifs, robot manipulateur, zone-morte, saturation, réseaux de neurones RBF, théorie de Lyapunov, la norme λ , perturbations non répétitives.

